TLE985xQX

Arm® Cortex®-M0 Microcontroller with LIN and H-Bridge NFET Driver for Automotive Applications
AD-Step

User Manual

About this document

This User Manual is addressed to embedded hardware and software developers. It provides the reader with detailed descriptions about the behavior of the TLE985xQX functional units and their interaction.

The manual describes the functionality of the superset device of the TLE985xQX Embedded Power IC familiy. For the available functionality (features) of a specific TLE985xQX derivative (derivative device), please refer to the respective Data Sheet. For simplicity, the various device types are referenced by the collective term TLE985xQX throughout this manual.

Table of Contents

1 Overview

- 32-bit Arm® Cortex®*-M0 Core
  - up to 40 MHz clock frequency
  - one clock per machine cycle architecture
  - single cycle multiplier
- On-chip memory
  - up to 96 KB Flash (product variant dependent, including EEPROM)
  - 4 KB EEPROM (emulated in Flash)
  - 512 bytes 100 Time Programmable Memory (100TP)
  - 4 KB RAM
  - Boot ROM for startup firmware and Flash routines
- Math Co-Processor Unit with Divider Unit for signed and unsigned 32-bit division operations
- On-chip OSC and PLL for clock generation
  - PLL loss-of-lock detection
- MOSFET Driver including charge pump for H-Bridge motor applications
- Current Sense Amplifier
- High-Side Switch with cyclic sense option and PWM functionality, e.g. for supplying LEDs or switch panels (min. 150 mA)
- 4 High Voltage Monitor Input pins for wake-up and with cyclic sense with analog measurement option
- 10 General-purpose I/O Ports (GPIO)
- 5 Analog input Ports
- 10-Bit A/D Converter with 5 analog inputs + VBAT_SENSE + VS + 4 high voltage monitoring inputs
- 8-Bit A/D Converter with 9 inputs for voltage and temperature supervision
- Measurement unit with 12 channels together with the onboard 10-Bit A/D converter and data post processing
- 16-Bit timers - GPT12, Timer 2 and Timer 21
- Capture/compare unit for PWM signal generation (CCU6)
- 2 full duplex serial interfaces (UART1, UART2), UART1 with LIN support
- 2 synchronous serial channels (SSC1, SSC2)
- On-chip debug support via 2-wire SWD
- 1 LIN 2.2 transceiver
- Single power supply V_S = 5.5 V to 28 V
- Extended supply voltage range V_S = 3 V to 28 V
- Low-dropout voltage regulators (LDO)
- 5 V voltage supply VDDEXT for external loads (e.g. Hall-sensor)
- Core logic supply at 1.5 V
- Programmable window watchdog (WDT1) with independent on-chip clock source
- Power saving modes:
  - Micro Controller Unit slow-down mode
  - Sleep Mode with cyclic sense option

  - Cyclic wake-up during Sleep Mode
  - Stop Mode with cyclic sense option
- Power-on and undervoltage/brownout reset generator
- Overtemperature protection incl. shutdown
- Short circuit protection for all voltage regulators and actuators (High Side Switch)
- Loss of clock detection with fail safe mode for power switches
- Temperature Range T_j = -40°C to +150°
- Package VQFN-48-31 with LTI feature
- Green package (RoHS compliant)
- AEC Qualified

2 Block Diagram

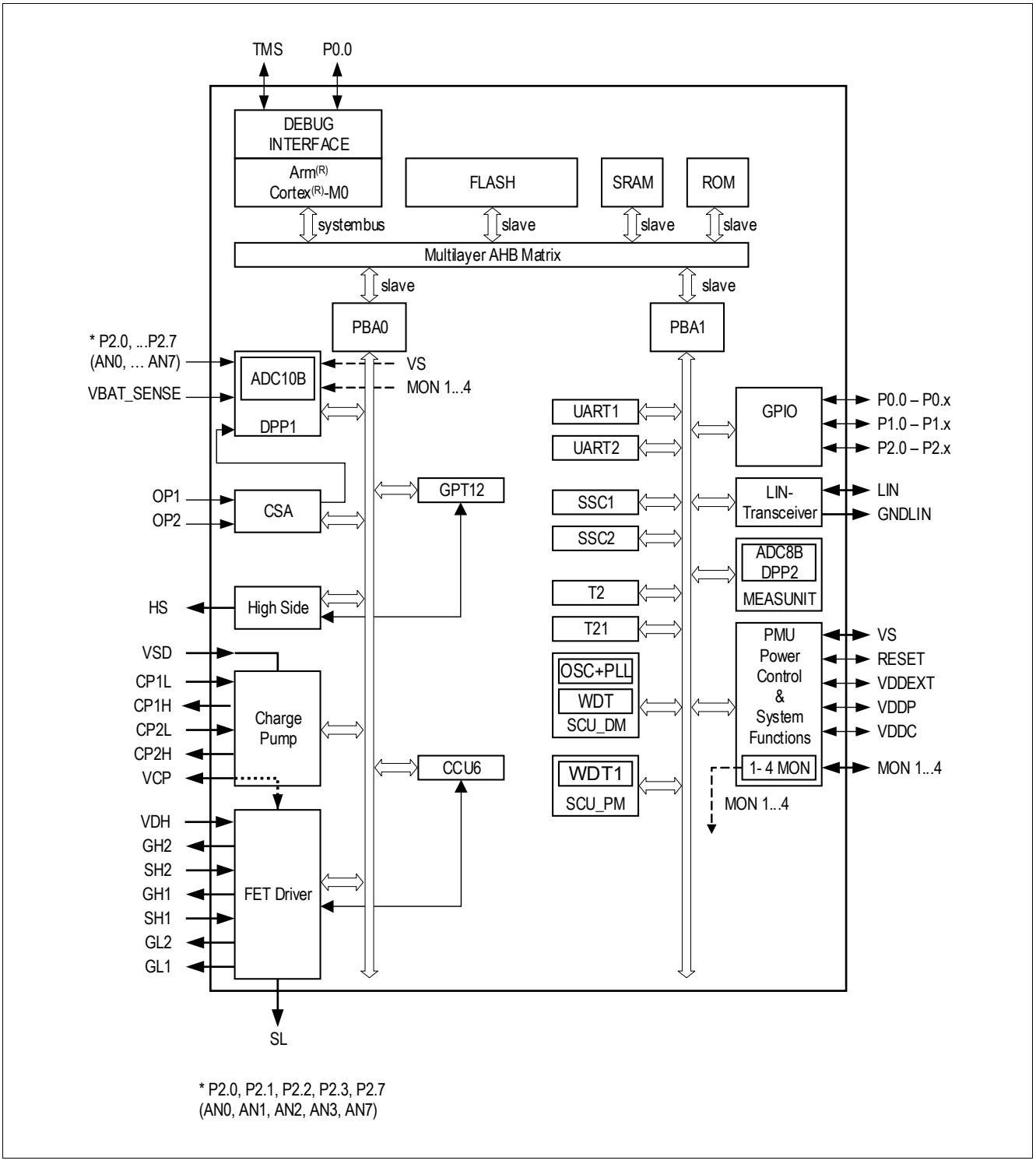


Figure 1 Block Diagram TLE985xQX

3 General Device Information

3.1 Pin Configurations

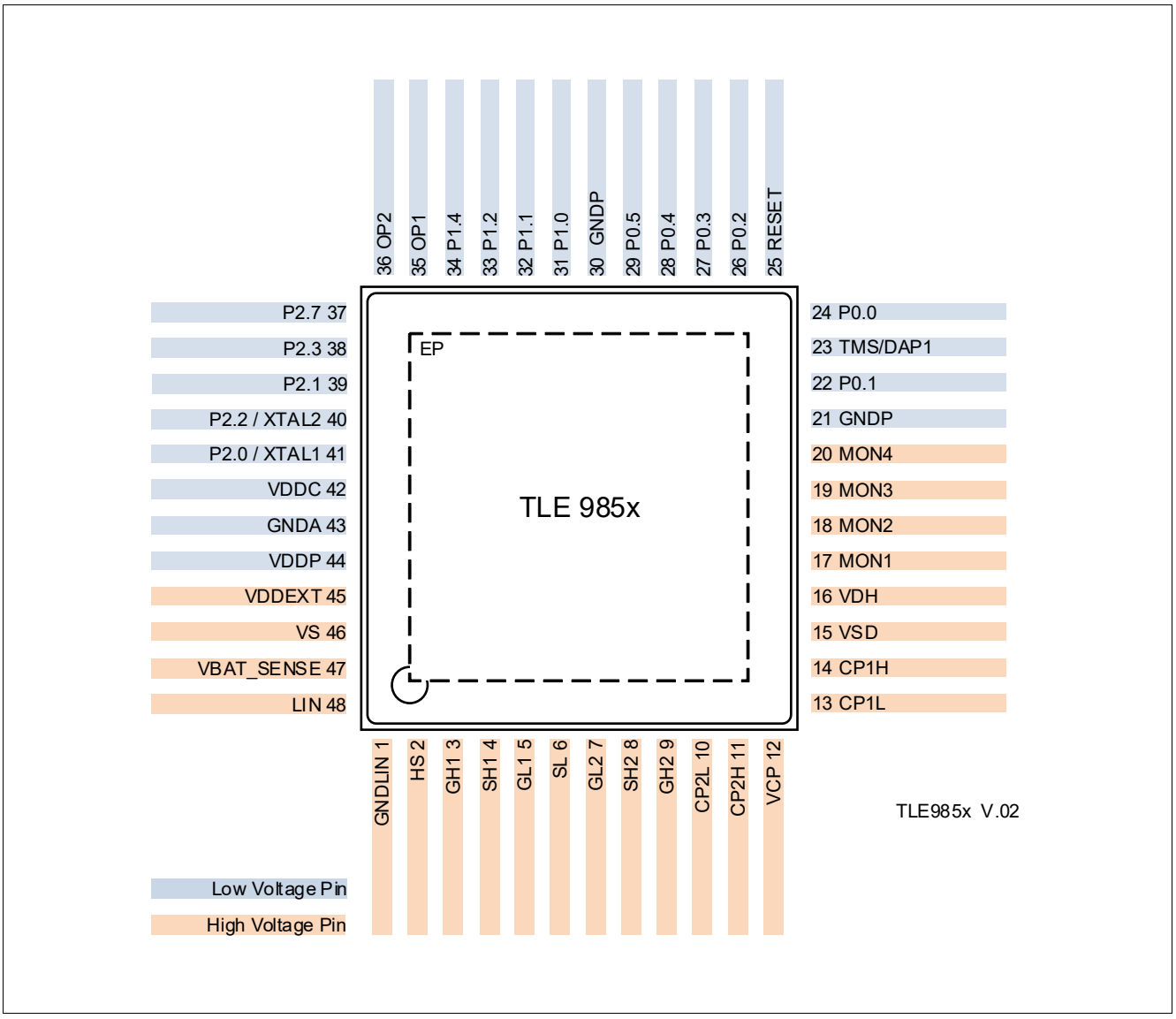


Figure 2 Pin Configuration VQFN-48, TLE985xQX

3.2 Pin Definitions and Functions

After reset, all pins are configured as input (except supply and LIN pins) with one of the following settings:

- Pull-up enabled only (PU)
- Pull-down enabled only (PD)
- Input with both pull-up and pull-down disabled (I)
- Output with output stage deactivated = high impedance state (Hi-Z)

The functions and default states of the TLE985xQX external pins are provided in the following table.

Type: indicates the pin type.

- I/O: Input or output
- I: Input only
- O: Output only
- P: Power supply

Not all alternate functions listed, see **Chapter 15**.

Table 1 Pin Definitions and Functions

Symbol	Pin Number	Type	Reset State	Function
P0				**Port 0** Port 0 is a 6-Bit bidirectional general purpose I/O port. Alternate functions can be assigned and are listed in the Port description. Main function is listed below.
P0.0	24	I/O	I/PU	SWD_CLK Serial Wire Debug Clock GPIO General Purpose IO Alternate function mapping see **Table 223**
P0.1	22	I/O	I/PU	GPIO General Purpose IO Alternate function mapping see **Table 223**
P0.2	26	I/O	I/PD	GPIO General Purpose IO Alternate function mapping see **Table 223**
P0.3	27	I/O	I/PU	GPIO General Purpose IO Alternate function mapping see **Table 223**
P0.4	28	I/O	I/PU	GPIO General Purpose IO Alternate function mapping see **Table 223**
P0.5	29	I/O	I/PU	GPIO General Purpose IO Alternate function mapping see **Table 223**
P1				**Port 1** Port 1 is a 4-Bit bidirectional general purpose I/O port. Alternate functions can be assigned and are listed in the Port description. Main function is listed below.
P1.0	31	I/O	I	GPIO General Purpose IO Alternate function mapping see **Table 224**
P1.1	32	I/O	I	GPIO General Purpose IO Alternate function mapping see **Table 224**
P1.2	33	I/O	I	GPIO General Purpose IO Alternate function mapping see **Table 224**

Table 1 Pin Definitions and Functions (cont'd)

Symbol	Pin Number	Type	Reset State	Function	
P1.4	34	I/O	I	GPIO	General Purpose IO Alternate function mapping see **Table 224**
P2				**Port 2** Port 2 is a 5-Bit general purpose input-only port. Alternate functions can be assigned and are listed in the Port description. Main function is listed below.	
P2.0	41	I	I	AN0 XTAL1[1)]	ADC1 analog input channel 6 External oscillator input Alternate function mapping see **Table 225**
P2.1	39	I	I	AN1	ADC1 analog input channel 7 Alternate function mapping see **Table 225**
P2.2	40	I O	I Hi-Z	AN2 XTAL2[1)]	ADC1 analog input channel 8 External oscillator output Alternate function mapping see **Table 225**
P2.3	38	I	I	AN3	ADC1 analog input channel 9 Alternate function mapping see **Table 225**
P2.7	37	I	I	AN7	ADC1 analog input channel 12 Alternate function mapping see **Table 225**
Power Supply					
VS	46	P	–	Battery supply input	
VDDP	44	P	–	I/O port supply (5.0 V). Do not connect external loads. For buffer and bypass capacitors.	
VDDC	42	P	–	Core supply (1.5 V during Active Mode, 0.9 V during Stop Mode). Do not connect external loads. For buffer/bypass capacitor.	
VDDEXT	45	P	–	External voltage supply output (5.0 V, 40 mA)	
GNDP	21, 30	P	–	Core supply ground	
GNDA	43	P	–	Analog supply ground	
GNDLIN	1	P	–	LIN ground	
Monitor Inputs					
MON1	17	I	I	High Voltage Monitor Input 1	
MON2	18	I	I	High Voltage Monitor Input 2	
MON3	19	I	I	High Voltage Monitor Input 3	
MON4	20	I	I	High Voltage Monitor Input 4	
High Side Switch Outputs					
HS	2	O	Hi-Z	High Side Switch output	
LIN Interface					
LIN	48	I/O	PU	LIN bus interface input/output	
Charge Pump					
CP1H	14	P	–	Charge Pump Capacity 1 High, connect external C	

Table 1 Pin Definitions and Functions (cont'd)

Symbol	Pin Number	Type	Reset State	Function
CP1L	13	P	–	Charge Pump Capacity 1 Low, connect external C
CP2H	11	P	–	Charge Pump Capacity 2 High, connect external C
CP2L	10	P	–	Charge Pump Capacity 2 Low, connect external C
VCP	12	P	–	Charge Pump Capacity
VSD	15	P	–	Battery supply input for Charge Pump
MOSFET Driver				
VDH	16	P	–	Voltage Drain High Side MOSFET Driver
GH1	3	P	–	Gate High Side FET 1
GH2	9	P	–	Gate High Side FET 2
SH1	4	P	–	Source High Side FET 1
SH2	8	P	–	Source High Side FET 2
GL1	5	P	–	Gate Low Side FET 1
GL2	7	P	–	Gate Low Side FET 2
SL	6	P	–	Source Low Side FETs
Others				
TMS	23	I	I/PD	TMS test mode select input DAP1
RESET	25	I/O	I/O/PU	Bidirectional reset input/output, not available during Sleep Mode
VBAT_SENSE	47	I	I	Battery supply voltage sense input
OP1	35	I	–	Negative current sense amplifier input
OP2	36	I	–	Positive current sense amplifier input
EP	–	–	–	Exposed Pad, connect to GND

1) configurable by user

4 Modes of Operations

This highly integrated circuit contains analog and digital functional blocks. For system and interface control an embedded 32-Bit Arm® Cortex®-M0 microcontroller is included. For internal and external power supply purposes, on-chip low drop-out regulators are existent. An internal oscillator (no external components necessary) provides a cost effective and suitable clock in particular for LIN slave nodes. As communication interface, a LIN transceiver and several High Voltage Monitor Inputs with adjustable threshold and filters are available. Furthermore one High-Side Switch (e.g. for driving LEDs or powering of switches), a driver for 4 n-channel MOSFETs including a two-stage charge pump and several general purpose input/outputs (GPIO) with pulse-width modulation (PWM) capabilities are available.

The Micro Controller Unit supervision and system protection including reset feature is controlled by a programmable window watchdog. A cyclic wake-up circuit, supply voltage supervision and integrated temperature sensors are available on-chip.

All relevant modules offer power saving modes in order to support terminal 30 connected automotive applications. A wake-up from the power saving mode is possible via a LIN bus message, via the monitoring inputs or repetitive with a programmable time period (cyclic wake-up).

The integrated circuit is available in a VQFN-48-31 package with 0.5 mm pitch and is designed to withstand the challenging conditions of automotive applications.

The TLE985xQX has several operational modes mainly to support low power consumption requirements. The low power modes and state transitions are depicted in **Figure 3** below.

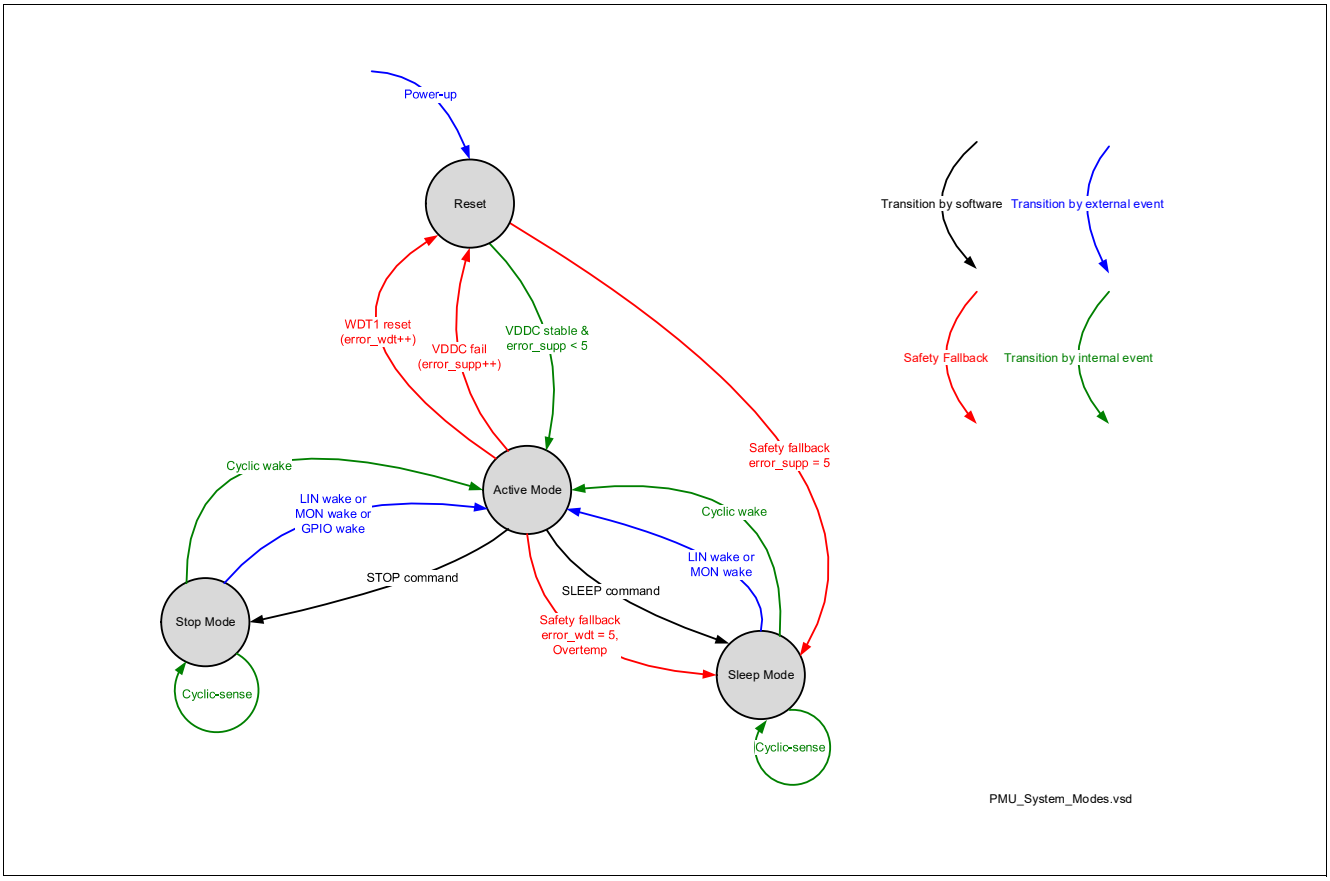


Figure 3 Power Control State Diagram

Modes of Operations

Reset Mode

The Reset Mode is a transition mode e.g. during power-up of the device after a power-on reset. In this mode the on-chip power supplies are enabled and all other modules are initialized. Once the core supply VDDC is stable, the Active Mode is entered . In case the watchdog timer WDT1 fails for more than four times, a fail-safe transition to the Sleep Mode is done.

Active Mode

In Active Mode all modules are activated and the TLE985xQX is fully operational.

Stop Mode

The Stop Mode is one out of two major low power modes. The transition to the low power modes is done by setting the respective Bits in the mode control register. In Stop Mode the embedded microcontroller is still powered allowing faster wake-up reaction times, but not clocked. A wake-up from this mode is possible by LIN bus activity, the High Voltage Monitor Input pins or the respective 5V GPIOs.

Sleep Mode

The Sleep Mode is a major low-power mode. The transition to the low-power modes is done by setting the respective Bits in the Micro Controller Unit mode control register. The sleep time is configurable. In Sleep Mode the embedded microcontroller power supply is deactivated, allowing the lowest system power consumption, but the wake-up time is longer compared to the Stop Mode. In this mode a 64 bit wide buffer for data storage is available. A wake-up from this mode is possible by LIN bus activity or the High Voltage Monitor Input pins and cyclic wake. A wake-up from Sleep Mode behaves similar to a power-on reset. While changing into Sleep Mode, no incoming wake-requests are lost (i.e. no dead-time). It is possible to enter sleep-mode even with LIN dominant.

Cyclic Wake-up Mode

The cyclic wake-up mode is a special operating mode of the Sleep Mode and the Stop Mode. The transition to the cyclic wake-up mode is done by first setting the respective Bits in the mode control register followed by the SLEEP or STOP command. Additional to the cyclic wake-up behavior (wake-up after a programmable time period), the wake-up sources of the normal Stop Mode and Sleep Mode are available.

Cyclic Sense Mode

The cyclic sense mode is a special operating mode of the Sleep Mode and the Stop Mode. The transition to the cyclic sense mode is done by first setting the respective Bits in the mode control register followed by the STOP or SLEEP command. In cyclic sense mode the High Side Switch can be switched on periodically for biasing some switches for example. The wake-up condition is configurable, when the sense result of defined monitor inputs at a window of interest changed compared to the previous wake-up period or reached a defined state respectively. In this case the Active Mode is entered immediately.

The following table shows the possible power mode configurations of each major module or function respectively.

Table 2 Power Mode Configurations

Module/function	Active Mode	Sleep Mode	Stop Mode	Comment
VPRE, VDDP, VDDC	ON	OFF	ON	–
VDDEXT	ON/OFF	OFF	cyclic ON/OFF	–
HS	ON/OFF	cyclic ON/OFF	cyclic ON/OFF	cyclic sense

Table 2 Power Mode Configurations (cont'd)

Module/function	Active Mode	Sleep Mode	Stop Mode	Comment
Bridge Driver	ON/OFF	OFF[1)]	OFF[1)]	–
LIN TRx	ON/OFF	wake-up only / OFF	wake-up only/ OFF	–
MONx (wake-up)	n.a.	disabled/static/ cyclic	disabled/static/ cyclic	cyclic: combined with HS=on
MONx (measurement)	ON/OFF	OFF	OFF	available on all channels
VS sense	ON/OFF brownout detection	brownout detection	brownout detection	brownout det. done in PCU
VBAT_SENSE	ON/OFF	OFF	OFF	–
GPIO 5V	ON	OFF	ON	–
WDT1	ON	OFF	OFF	–
CYCLIC WAKE	n.a.	cyclic wake-up/ cyclic sense/OFF	cyclic wake-up/ cyclic sense/OFF	cyclic sense with HS; wake-up needs MC for enter Sleep Mode again
Measurement	ON[2)]	OFF	OFF	–
Micro Controller Unit	ON/slow-down/STOP	OFF	OFF	–
CLOCK GEN (MC)	ON	OFF	OFF	–
LP_CLK (f_{LP_CLK})	ON	OFF	OFF	WDT1
LP_CLK2 (f_{LP_CLK2})	ON	ON	ON	for cyclic wake-up

1) Bridge Driver "Hold Mode" is available in sleep mode and stop mode.

2) May not be switched off due to safety reasons

Wake-up Source Prioritization

All wake-up sources have the same priority. In order to handle the asynchronous nature of the wake-up sources, the first wake-up signal will initiate the wake-up sequence. Nevertheless all wake-up sources are latched in order to provide all wake-up events to the application software. The software can clear the wake-up source flags. It is ensured, that no wake-up event is lost.

As default wake-up sources, MON inputs and cyclic wake are activated after power-on reset, LIN is disabled as wake-up source by default.

Wake-up Levels and Transitions

The wake-up can be triggered by rising, falling or both signal edges for each monitor input individually.

5 Device Register Types

The following register types are used within this device. List see in **Table 3**.

Table 3 Register Types

Type	can be modified by			Description	Error response
	Hard ware	Firmware	Soft ware		on write access
r	yes	no	no	read-only flag	err
rc	yes	no	yes	set when HW = 1, clear on read, with handshake	err
rh	yes	no	no	read-only flag which is modified by hardware	err
rw	no	no	yes	bit can be read or written	ok
rwh	yes	no	yes	bit can be written by hardware and software; hardware has priority	ok
rwh1	yes	no	yes	bit can be written by hardware and software; hardware has priority.	ok
rwhir	yes	no	yes	bit can be written by hardware and software; hardware has only priority to clear the bit.	ok
rwhrs	yes	no	yes	reset when HWr = 1; set/reset by FW, set when HWs = 1	ok
rwhxr	yes	no	yes	set by HW, set by FW; clear by external reg	ok
rwhxre	yes	no	yes	set by HW (edge-triggered), set by FW; clear by external reg	ok
rwpt	no	yes	no	protected bit; read operation is always possible	err
rwpw	no	yes	no	password protected	ok
rwt	no	no	yes	read/write toggle	ok
rwv	no	no	yes	virtual rw	ok
w	no	no	yes	clear on write '1', no action on write '0'; read always '0'; (no FF)	ok

6 Power Management Unit (PMU)

6.1 Features

- System modes control (startup, sleep, stop and active)
- Power management (cyclic wake, cyclic sense)
- Control of system voltage regulators with diagnosis (overload, short, overvoltage)
- Fail safe mode detection and operation in case of system errors
- Wake-up sources configuration and management (LIN, MON, GPIOs)
- System error logging

6.2 Introduction

The purpose of the power management unit is to ensure the fail safe behavior of the system IC. Therefore the power management unit controls all system modes including the corresponding transitions. The power management unit is responsible for generating all needed voltage supplies for the embedded MCU (VDDC, VDDP) and the external supply (VDDEXT). Additionally, the PMU provides well defined sequences for the system mode transitions and generates hierarchical reset priorities. The reset priorities control the reset behavior of all system functionalities especially the reset behavior of the embedded MCU. All these functions are controlled by finite state machines. The system master functionality of the PMU requires the generation of an independent logic supply and system clock. Therefore the PMU has a module internal logic supply and system clock which works independently of the MCU clock.

6.2.1 Block Diagram

The following figure shows the structure of the Power Management Unit. **Table 4** describes the submodules more detailed.

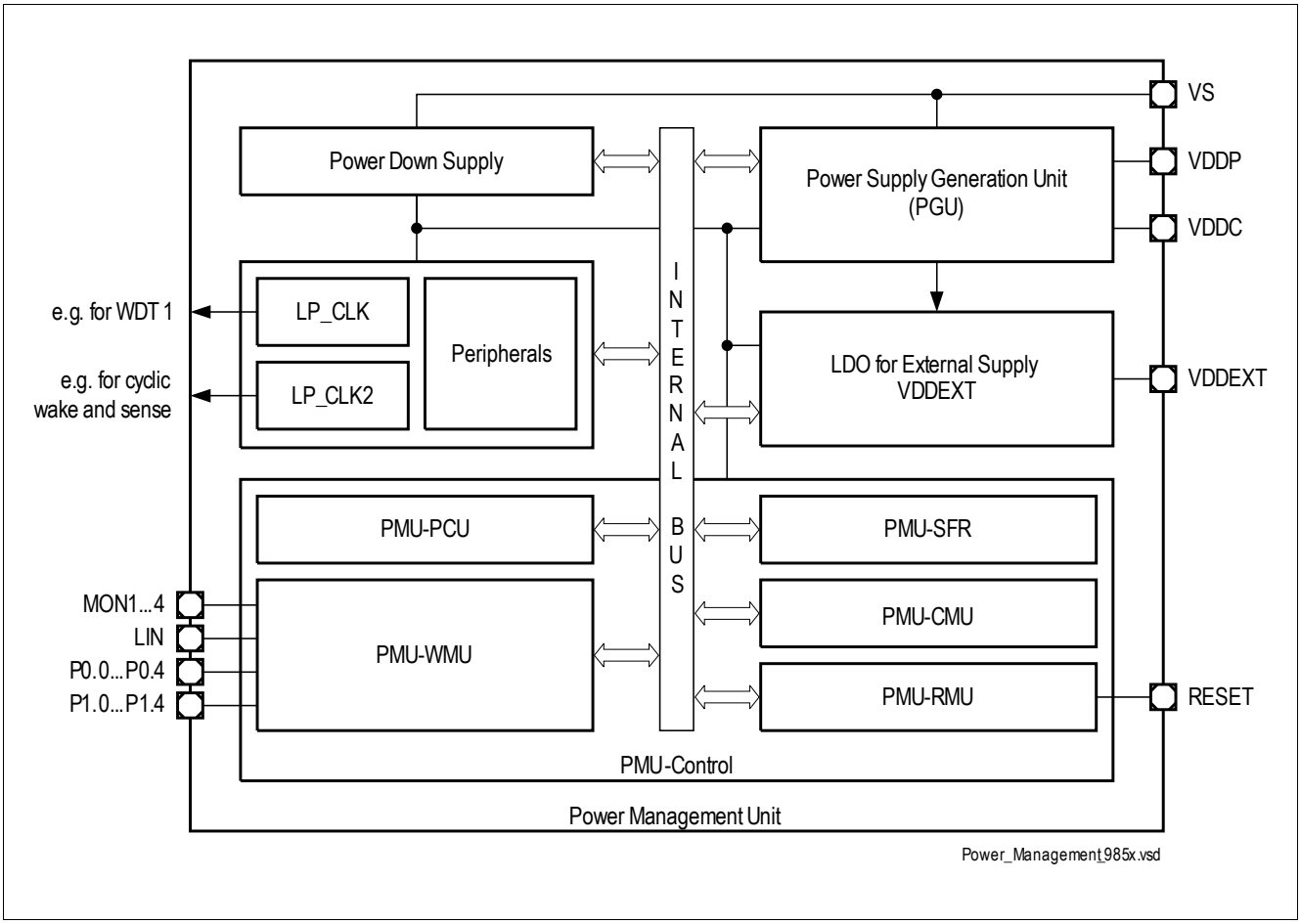


Figure 4 Power Management Unit Block Diagram

Table 4 Description of PMU Submodules

Mod. Name	Modules	Functions
Power Down Supply	Independent Supply Voltage Generation for PMU	This supply is dedicated to the PMU to ensure an independent operation from generated power supplies (VDDP, VDDC).
LP_CLK (= f_{LP_CLK})	- Clock Source for all PMU submodules - Backup Clock Source for System (can be selected as fsys clock source through SCU_APCLK.SYSCLKSEL) - Clock Source for WDT1	This ultra low power oscillator generates the clock for the PMU. This clock is also used as backup clock for the system in case of PLL Clock failure and as independent clock source for WDT1.
LP_CLK2 (= f_{LP_CLK2})	Clock Source for PMU	This ultra low power oscillator generates the clock for the PMU in Stop Mode and in the cyclic modes.

Table 4 Description of PMU Submodules (cont'd)

Mod. Name	Modules	Functions
Peripherals	Peripheral Blocks of PMU	These blocks include the analog peripherals to ensure a stable and fail safe PMU startup and operation (bandgap, bias).
Power Supply Generation Unit (PGU)	Voltage regulators for VDDP and VDDC	This block includes the voltage regulators for the pad supply (VDDP) and the core supply (VDDC).
VDDEXT	Voltage regulator for VDDEXT to supply external modules (e.g. Sensors)	This voltage regulator is a dedicated supply for external modules.
PMU-SFR	All PMU relevant Extended Special Function Registers	This module contains all PMU relevant registers, which are needed to control and monitor the PMU.
PMU-PCU	Power Control Unit of the PMU	This block is responsible for controlling all power related actions within the PGU Module.It also contains all regulator related diagnosis like under- and overvoltage detection, overcurrent and short circuit diagnoses.
PMU-WMU	Wake-up Management Unit of the PMU	This block is responsible for controlling all Wake-up related actions within the PMU Module.
PMU-CMU	Cyclic Management Unit of the PMU	This block is responsible for controlling all actions within cyclic mode.
PMU-RMU	Reset Management Unit of the PMU	This block generates resets triggered by the PMU like undervoltage or short circuit reset, and passes all resets to the relevant modules and their register. A reset status register with every reset source is available.

6.2.2 PMU Modes Overview

The following state diagram shows the available modes of the device.

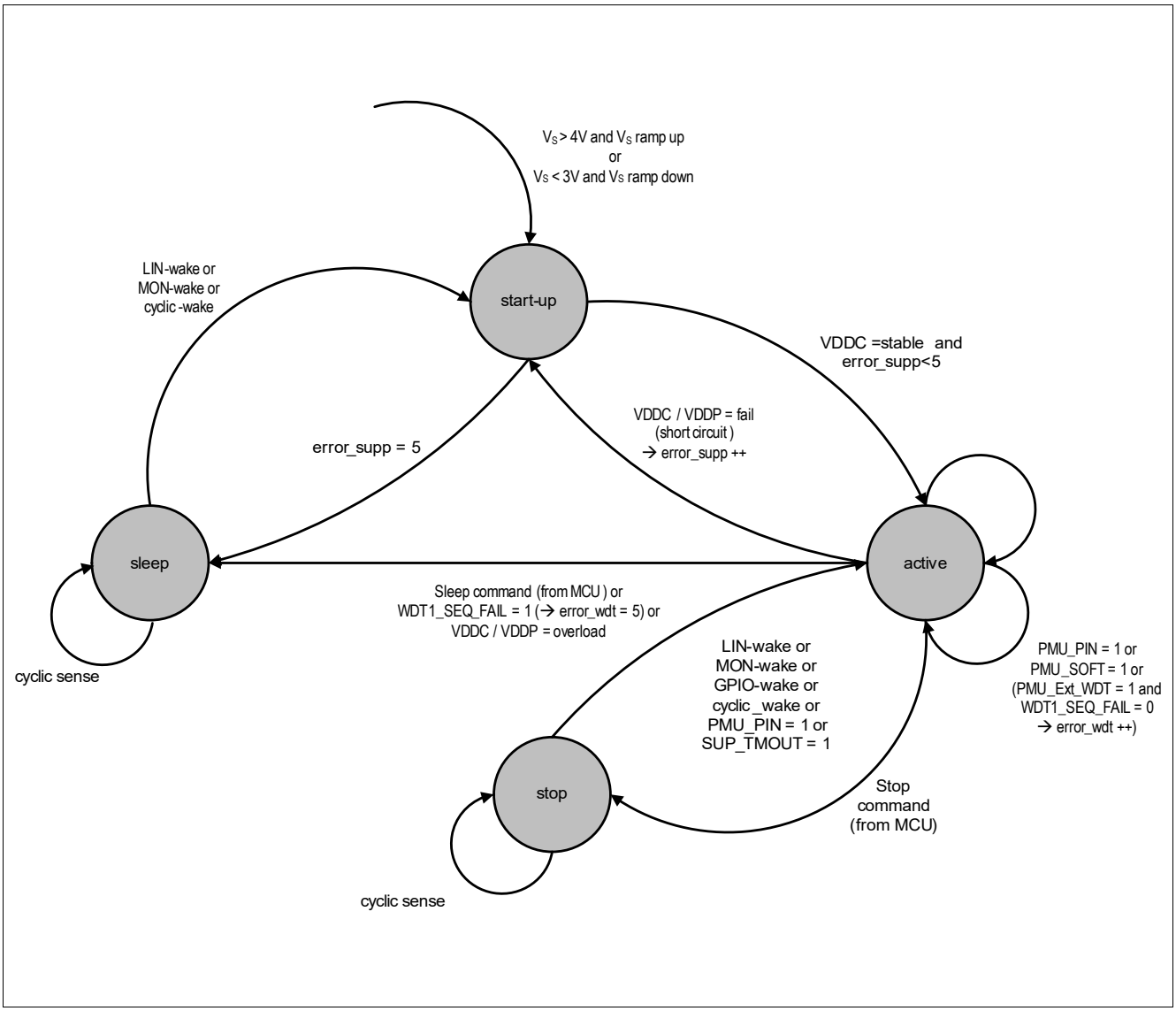


Figure 5 Power Management Unit System Modes

Active Mode

In Active Mode the Power Management Unit releases the reset of the embedded MCU and the application software takes control of the system. Now the PMU is responsible for supplying and supervising the embedded system. The supervision functionality of the PMU monitors the output voltage/current of the generated supplies and the status information of the system watchdog (WDT1).

Under normal operating conditions (exceptions see Chapter **Power Control Unit - Fail Safe Scenarios**) the power save modes are set by the user software only. The PMU gets the respective command and after a certain delay the corresponding ready signal will follow. The user software has to write the command to the power mode control register (PMCON0) of the SCU. As a consequence the SCU sends the MCU in data retention mode and accepts this with the respective ready signal.

Sleep Mode

The Sleep Mode is the power saving mode where the lowest power consumption is achieved. In this mode the PMU resets all system functionalities and switches off all voltage supplies (VDDP, VDDC, VDDEXT) which are generated in the PMU. The only submodules of the PMU which stay active are the ones responsible for controlling the wake-up procedure of the system. **Figure 6** shows the Sleep Mode entry procedure.

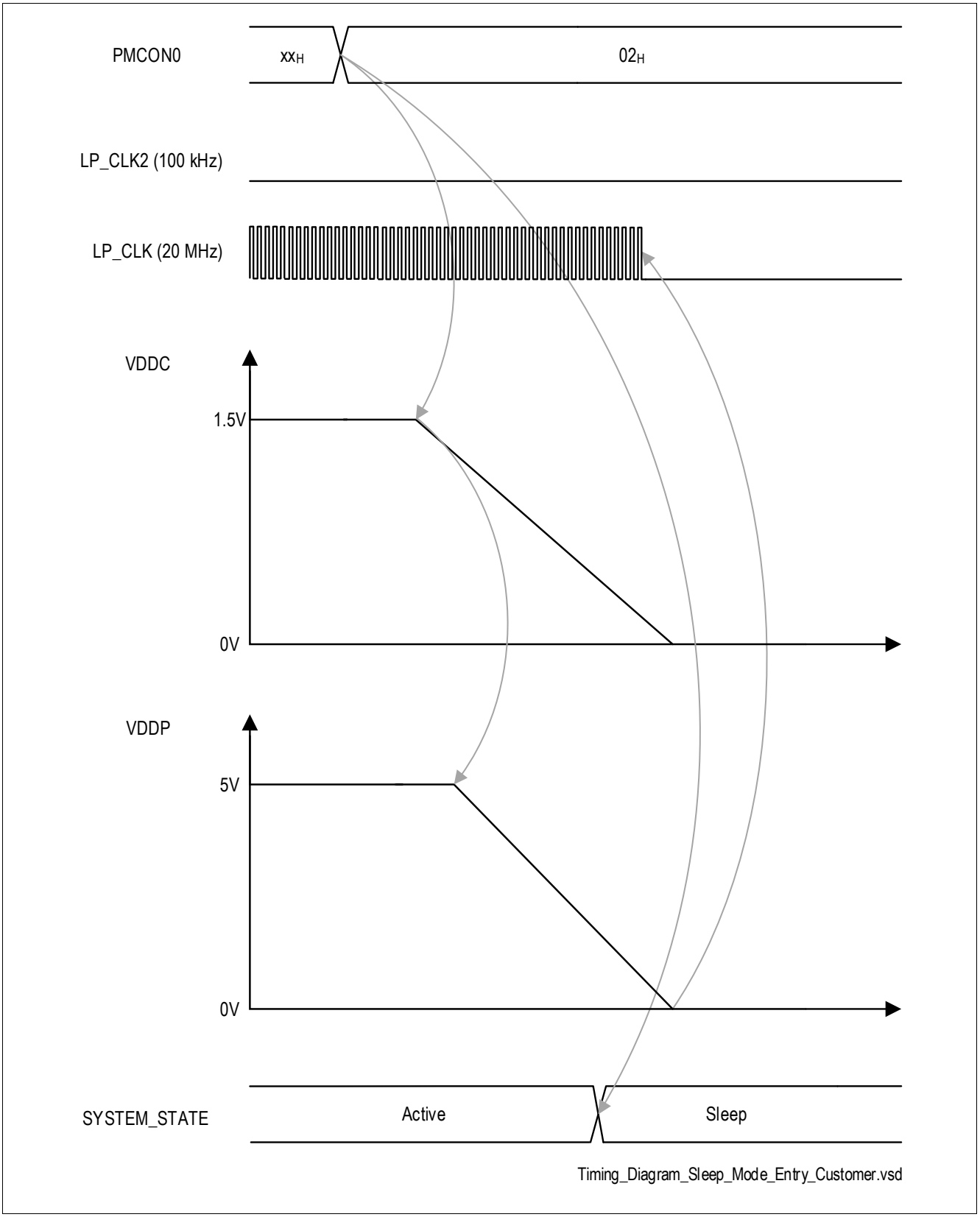


Figure 6 Sleep Mode Entry Timing

The Sleep Mode is terminated by a LIN dominant pulse or a corresponding (rising edge / falling edge) activity at the MON input. These events are triggered outside of the PMU. The PMU itself processes the wake-up information in an independent FSM which starts the PMU internal system clock to process the following

startup sequences in a synchronous way. A successful startup sequence enters the startup Mode automatically. The wake-up procedure described is the default setup of the PMU.

The Sleep Mode can be terminated by synchronous wake-up events too. If this is desired, the PMU must be configured by setting the corresponding SFRs. A synchronous wake-up can be configured using the Cyclic Sense. If these synchronous wake-up events are configured then the power consumption of the PMU increases in Sleep Mode. The increased current consumption is caused by an oscillator which generates the needed time base (typ. 100 kHz). **Figure 7** illustrates the wake via LIN

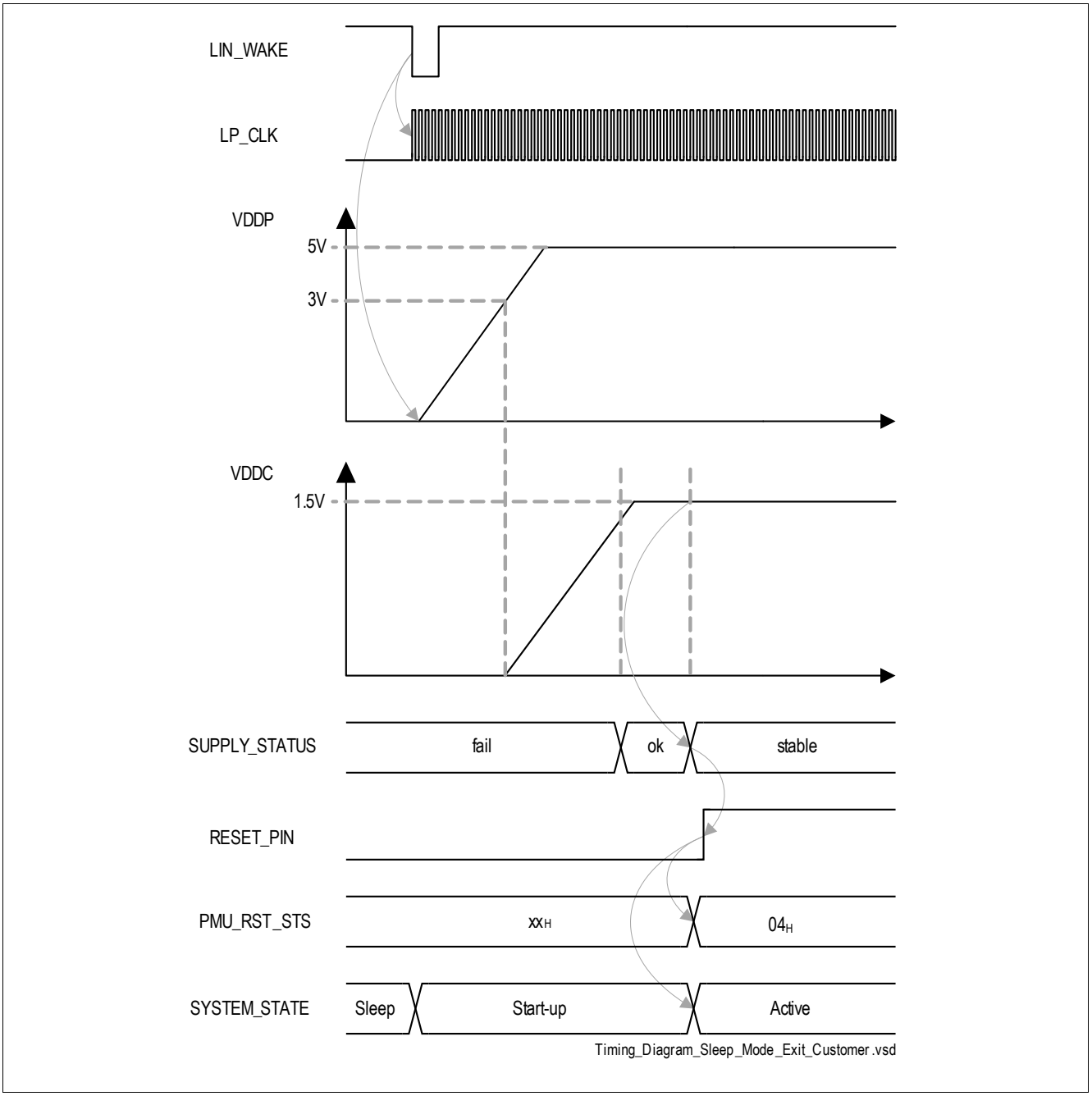


Figure 7 Sleep Mode LIN Wake-up Timing

The wake-up procedure from Sleep Mode via MONx pins (instead of LIN) follows the same sequence as shown in the figure above.

Stop Mode

The objective of the Stop Mode is to provide a data retention feature for the embedded MCU and the special function registers (XSFRs). In the Stop Mode the core supply voltage VDDC goes from 1.5 V to 0.9 V with the objective to reduce leakage current as much as possible. During the Stop Mode the dynamic behavior (load jumps) of the PMU internally generated voltage supplies are very limited. The corresponding limitation is given by the external buffer capacitor at the VDDC/VDDP pin. In case of a 330 nF buffer capacitor at VDDC the allowable load jump is 300µA/ms. The figure below shows the Stop Mode entry sequence.

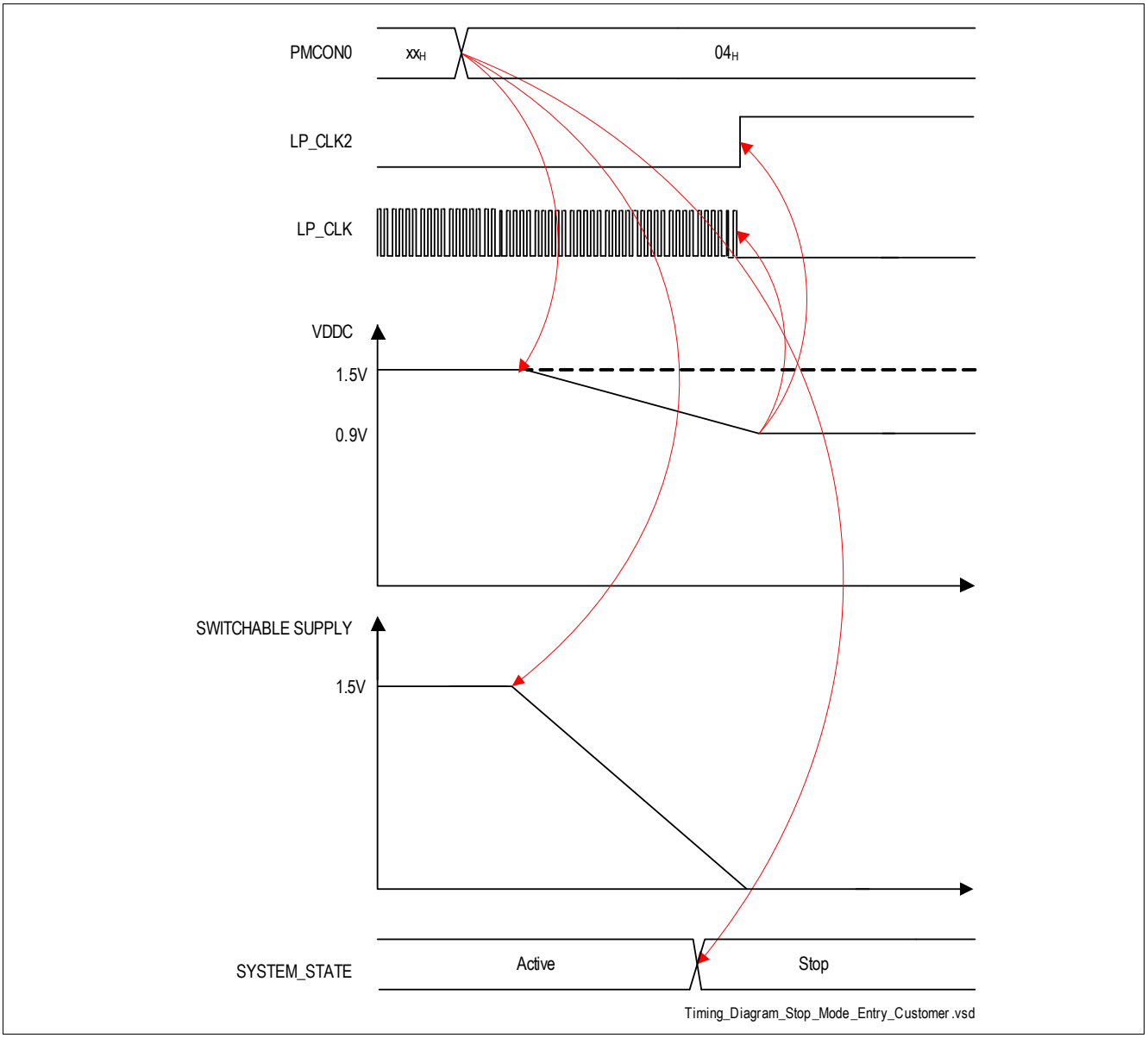


Figure 8 Stop Mode Entry Timing

The wake-up features to terminate the Stop Mode are equivalent to those which are used for Sleep-exit. The asynchronous wake-up works using a LIN message or an event (rising edge/falling edge) at one of the MON inputs. In addition to the asynchronous wake-up over high-voltage inputs (MONs) the Stop Mode terminates by an event at one of the GPIO pins. The wake-up configuration of every MON and GPIO input is stored in the corresponding XSFR. The configuration for the high-voltage inputs (MONs) are used for Stop-exit and Sleep-exit (same XSFR). Generally the synchronous wake-up features are equivalent to the Sleep Mode exit. The Stop Mode terminates by using one of the synchronous wake-up features. The synchronous wake-up features are

separated in Cyclic Sense and wake-up after time-out (Cyclic Wake). Both of these wake-up procedures work similarly to the Sleep-exit. In Cyclic Sense mode, both the MONx inputs as well as the GPIOs can be evaluated and a transition will cause a termination of the Stop Mode. The sensing period for MONx inputs and GPIOs is generated with the same time base (typ. 100 kHz). The sensing period is set in the **PMU_SLEEP**. To bias the external load of the GPIOs, the supply voltage VDDEXT may switch on for the sensing time. Only during this sensing time the PMU evaluates the corresponding GPIO. In case of a valid wake-up signal the PMU goes to Active Mode and the application software takes control over the system. If no valid wake-up information is available, then the external supply VDDEXT switches off until the configured sensing period starts again.

6.3 Power Supply Generation (PGU)

As shown in the diagram below the Power Supply Generation consists of the following modules:

Submodules of PSG are:

- **Power Down Supply:** independent analog supply voltage generation for Power Control Unit logic, for VDDP Regulator and for VDDC Regulator.
- **VPRE:** analog supply voltage pre-regulator. Purpose of this regulator is the power dissipation reduction for the following regulator stages.
- **VDDP:** 5V digital voltage regulator used for internal modules and all GPIOs.
- **VDDC:** 1.5V digital voltage regulator used for internal microcontroller modules and core logic.
- **PCU:** Power Control Unit responsible for supervising and controlling 5V regulator and 1.5V regulator.

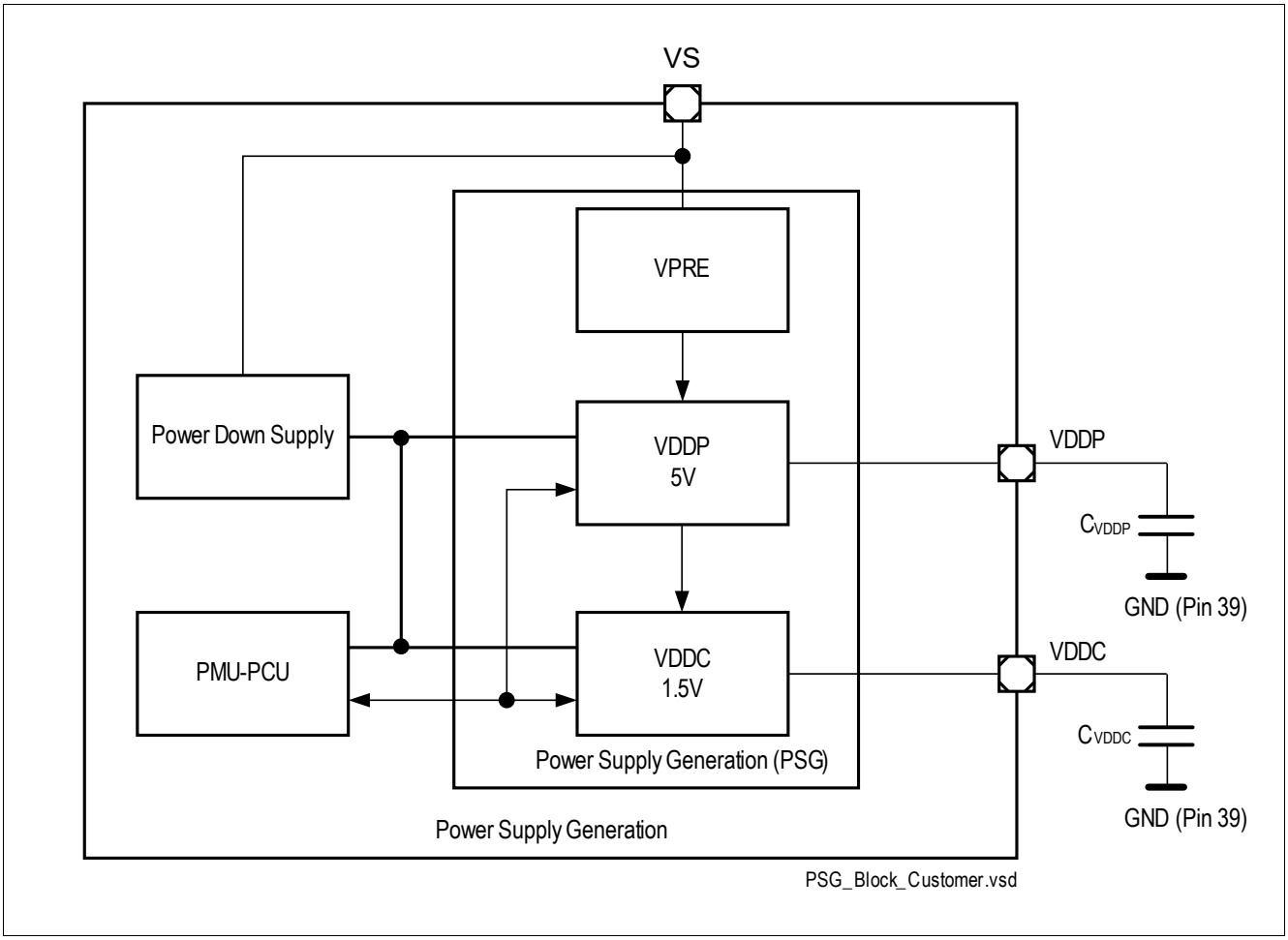


Figure 9 Power Supply Generation Block Diagram

6.3.1 Voltage Regulator 5.0V (VDDP)

This module represents the 5 V voltage regulator, which provides the pad supply for the parallel port pins and other 5 V analog functions (e.g. LIN Transceiver).

Power Management Unit (PMU)

Features

- 5 V low-drop voltage regulator
- Overcurrent Monitoring and Shutdown with MCU signalling (Interrupt)
- Overvoltage monitoring with MCU signalling (Interrupt)
- Undervoltage monitoring with MCU signalling (Interrupt)
- Undervoltage monitoring with Reset (UnderVoltage Reset, V_{DDPUV})
- Overtemperature Shutdown with MCU signalling (Interrupt)
- Pre-Regulator for VDDC Regulator
- GPIO Supply
- Pull Down Current Source at the output for Sleep Mode only

The output capacitor C_{VDDP} is mandatory to ensure a proper regulator functionality.

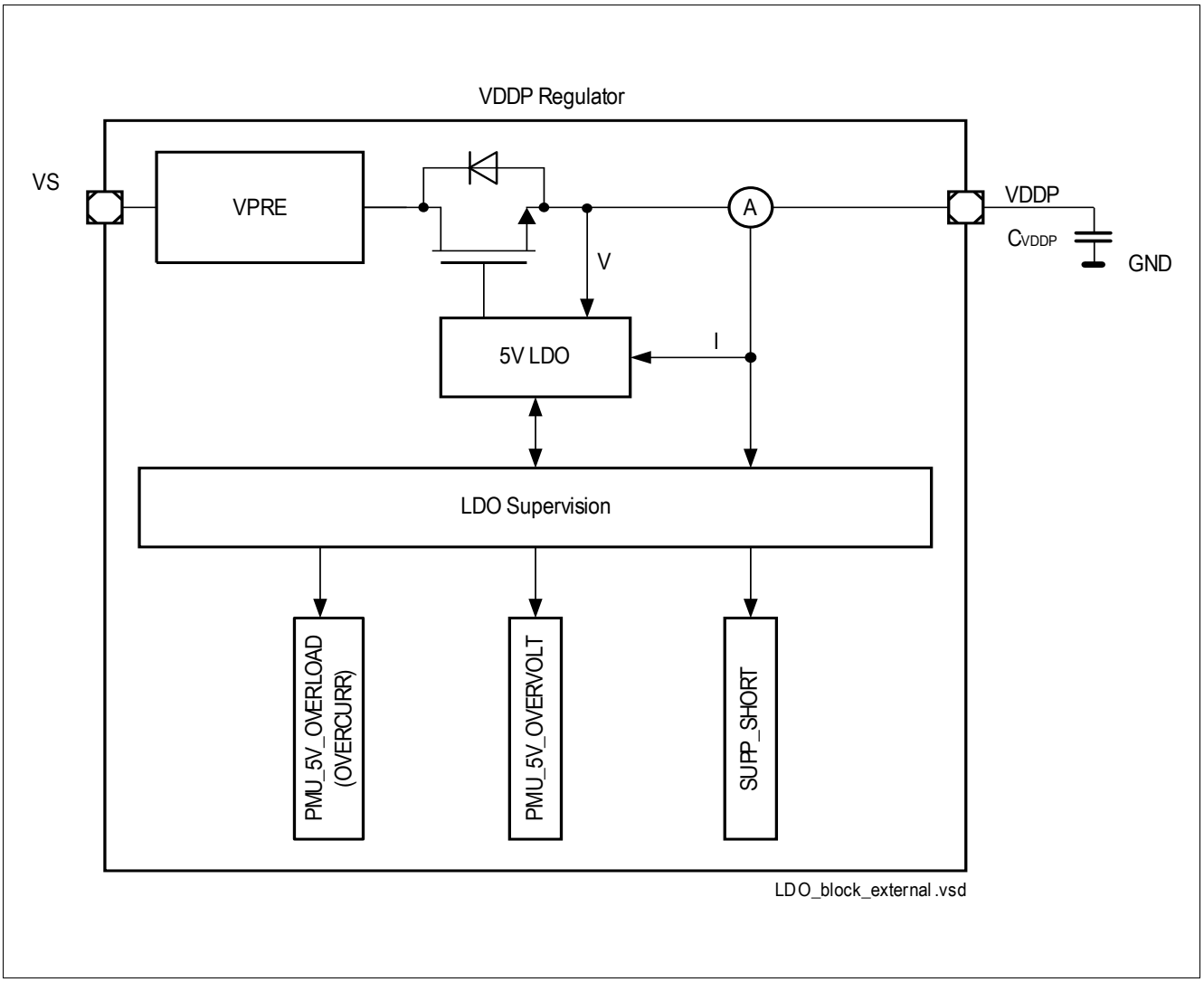


Figure 10 Module Block Diagram of VDDP Voltage Regulator

6.3.2 Voltage Regulator 1.5V (VDDC)

This module represents the 1.5 V voltage regulator, which provides the supply for the microcontroller core, digital peripherals and other chip internal analog 1.5 V functions (e.g. ADC).

Features

- 1.5 V low-drop voltage regulator
- Overcurrent monitoring and Shutdown with MCU signalling (Interrupt)
- Overvoltage monitoring with MCU signalling (Interrupt)
- Undervoltage monitoring with MCU signalling (interrupt)
- Undervoltage monitoring with reset
- Overtemperature Shutdown with MCU signalling (Interrupt)
- Pull Down Current Source at the output for Sleep Mode only

The output capacitor C_{VDDC} is mandatory to ensure a proper regulator functionality.

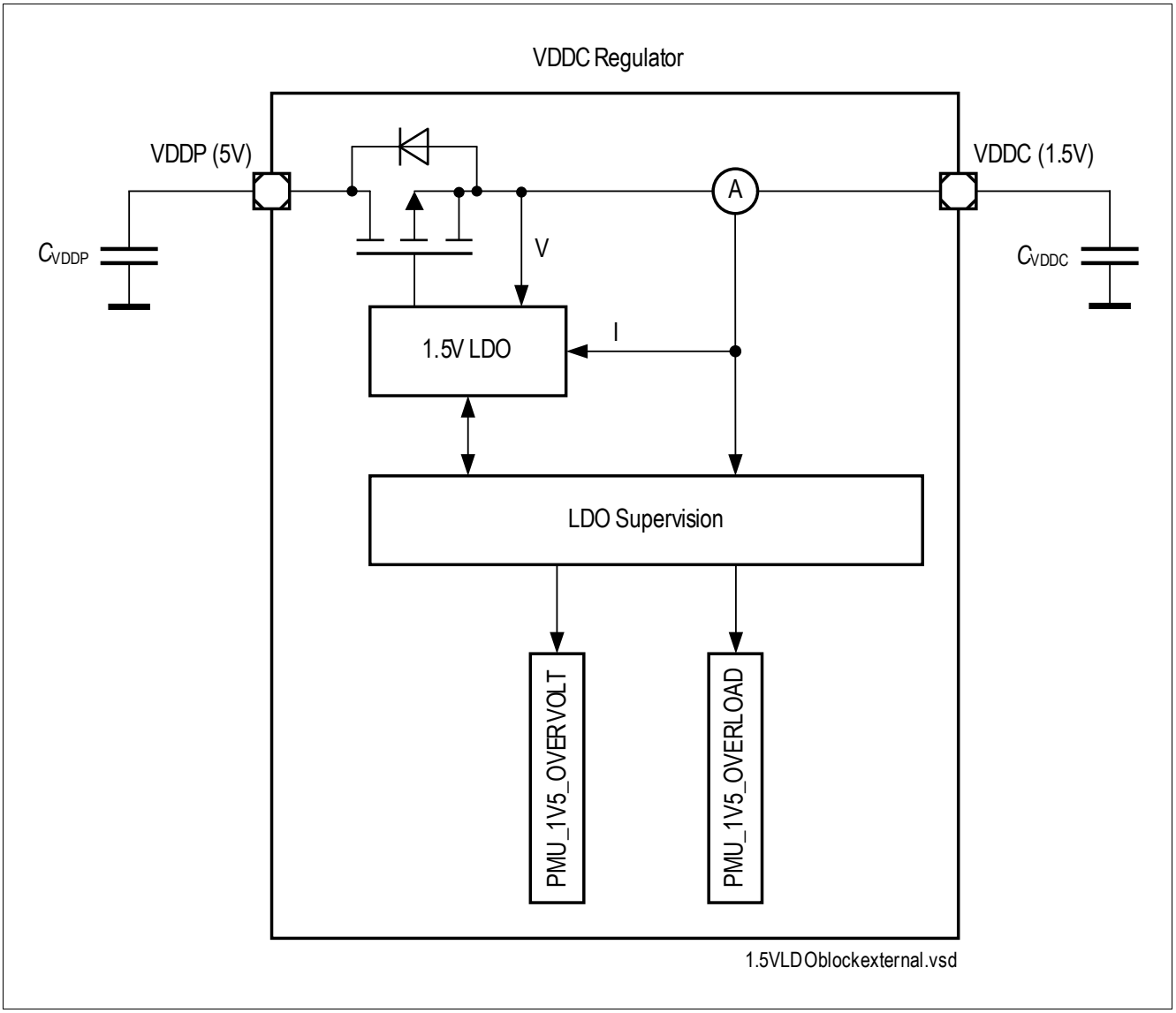


Figure 11 Module Block Diagram of VDDC Voltage Regulator

6.3.3 External Voltage Regulator 5.0V (VDDEXT)

This module represents the 5 V voltage regulator, which serves as a supply for external circuits. It can be used e.g. to supply an external sensor, LEDs or potentiometers.

Features

- Switchable (by software) 5 V low-drop voltage regulator
- Switch-on undervoltage blanking time in order to drive small capacitive loads
- Intrinsic current limitation
- Undervoltage monitoring and shutdown with MCU signalling (Interrupt)
- Overtemperature Shutdown with MCU signalling (Interrupt)
- Resistive discharge path at the output if the regulator is off
- Cyclic sense option together with GPIOs
- Low current mode available to ensure reduced stop mode current consumption. In this mode current capability is reduced to I_{VDDEXT_LCM}

The output capacitor C_{VDDEXT} is mandatory to ensure a proper regulator functionality.

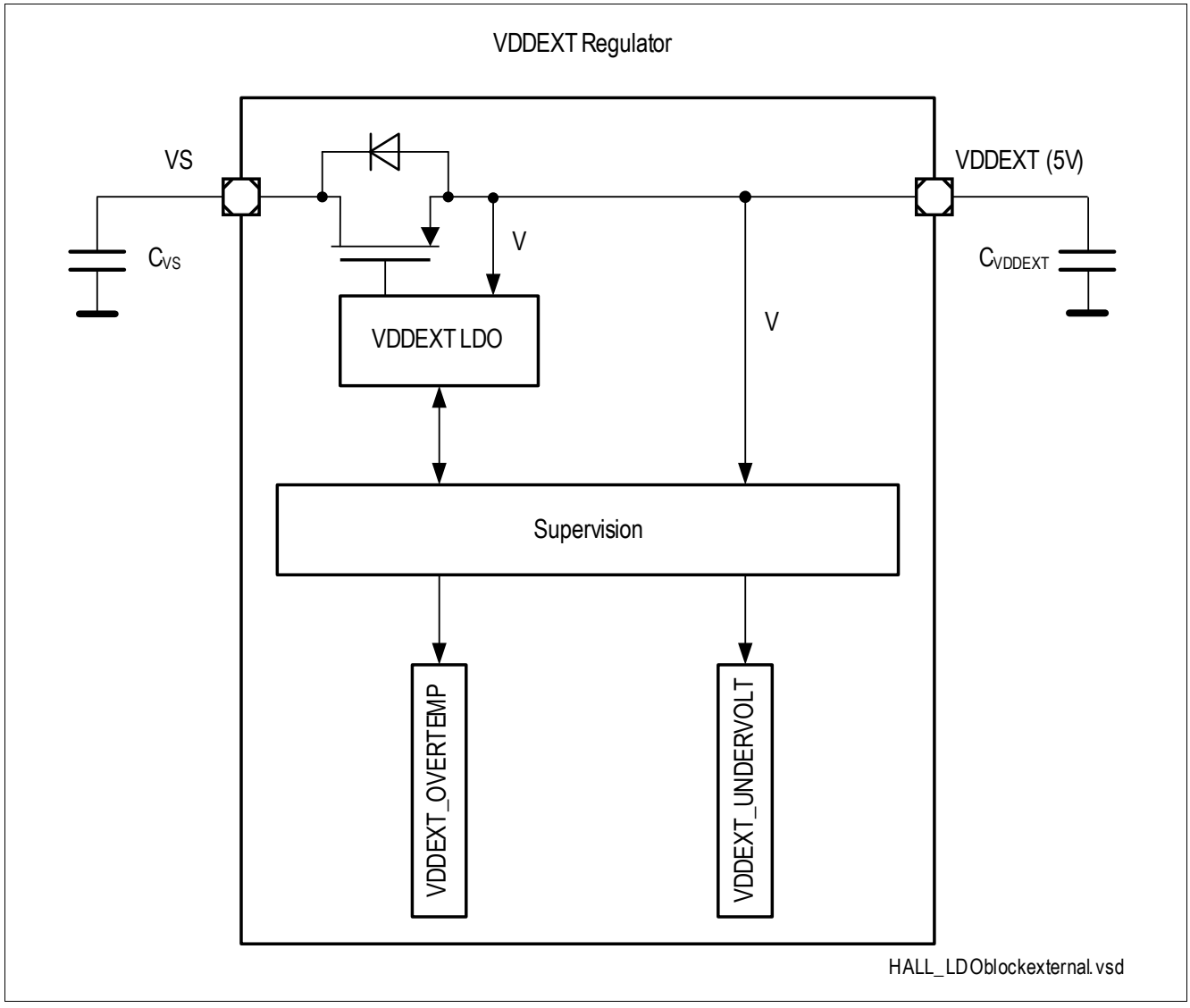


Figure 12 Module Block Diagram

6.3.4 Power-on Reset Concept

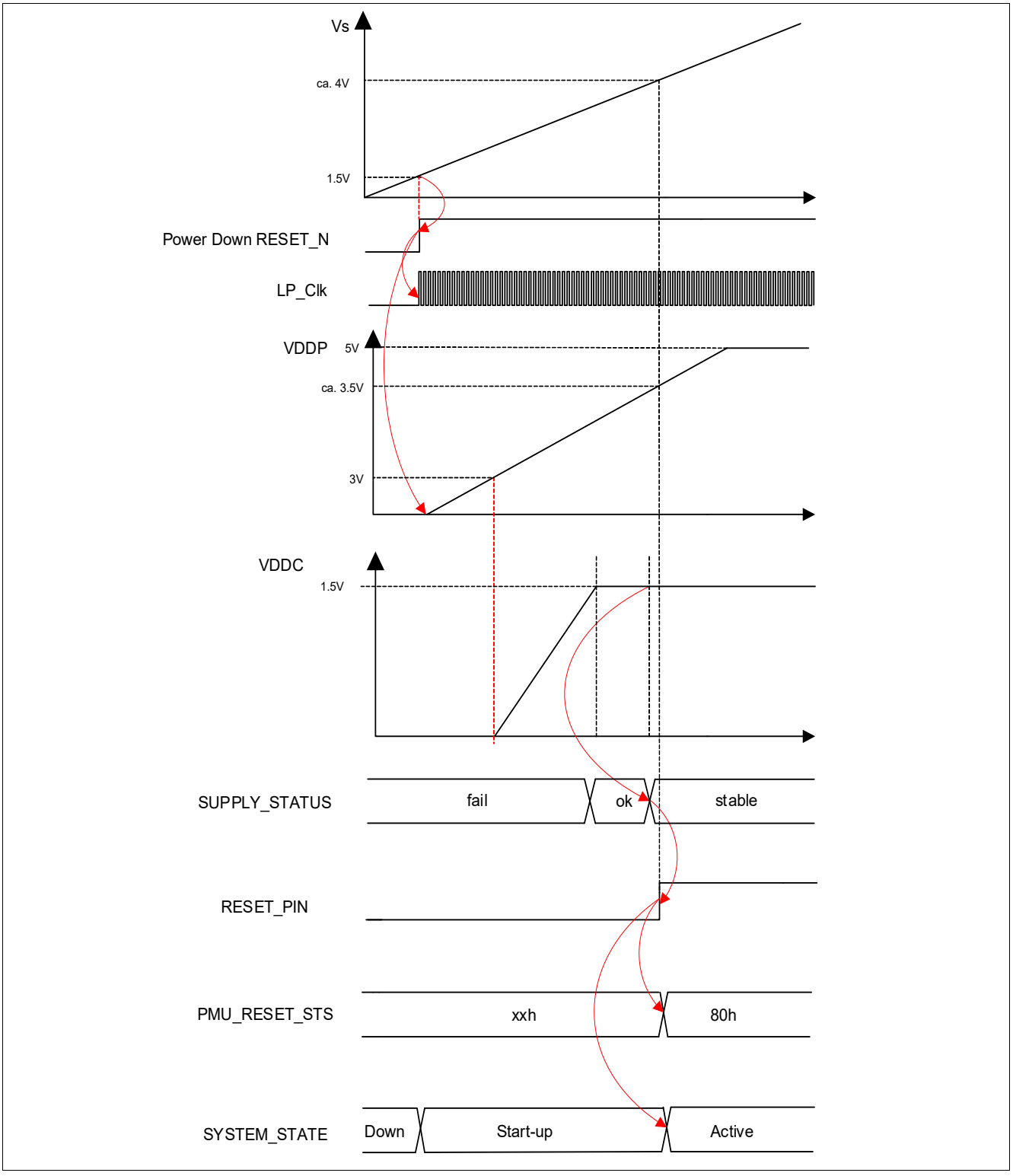


Figure 13 Power-on Reset Concept

6.3.5 PMU Register Overview

The PMU registers must be accessed wordwise. Otherwise a hardfault is triggered.

Table 5 Register Address Space for PMU Registers

Module	Base Address	End Address	Note
PMU	50004000_H	$50004FFF_H$	Power Management Unit Registers

The registers are addressed wordwise.

6.3.6 Register Definition

Table 6 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Register Definition, Power Supply Generation Register			
PMU_SUPPLY_STS	Voltage Reg Status Register	008_H	$0000\ 0000_H$
Register Definition, VDDEXT Control Register			
PMU_VDDEXT_CTRL	VDDEXT Control	$00C_H$	$0000\ 0000_H$

The registers are addressed wordwise.

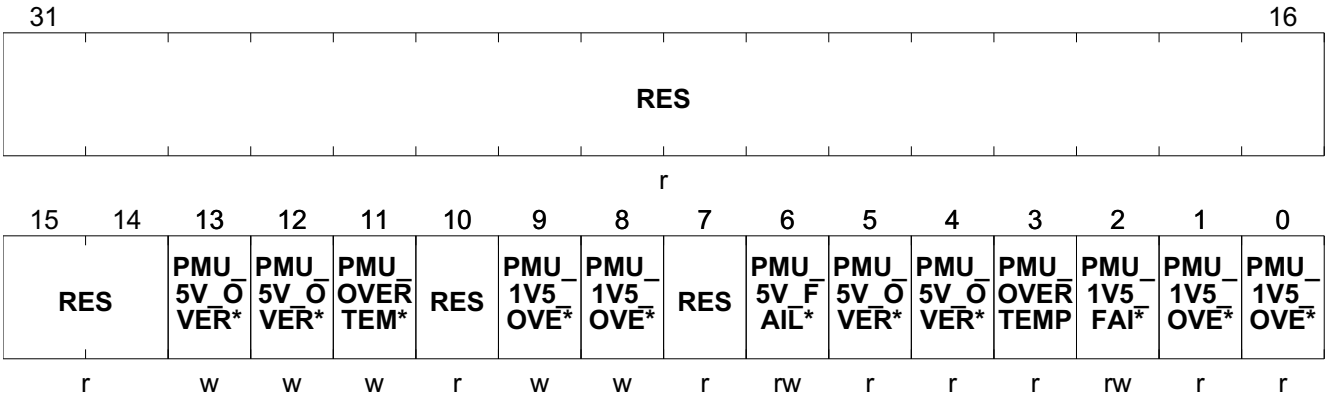
6.3.6.1 Power Supply Generation Register

The following register is dedicated to control the voltage regulators VDDP, VDDC. It provides an overview about the status of the two voltage supplies.

Voltage Reg Status Register

The PMU_SUPPLY_STS register shows the overvoltage and overload condition of VDDP and VDDC. To use this information as interrupt sources it must be selected explicitly in this register.

PMU_SUPPLY_STS	Offset	Reset Value
Voltage Reg Status Register	**008_H**	**see Table 7**



Field	Bits	Type	Description
RES	31:14	r	**Reserved** Always read as 0
PMU_5V_OVERLOAD_SC	13	w	**Overload at VDDP regulator Status clear** 0_B **No Clear**, Overload status not cleared 1_B **Clear**, Overload status cleared
PMU_5V_OVERVOLT_SC	12	w	**Overvoltage at VDDP regulator Status clear** 0_B **No Clear**, Overvoltage status not cleared 1_B **Clear**, Overvoltage status cleared
PMU_OVERTEMP_SC	11	w	**Overtemperature Status clear** 0_B **No Clear**, Overtemperature status not cleared 1_B **Clear**, Overtemperature status cleared
RES	10	r	**Reserved** Always read as 0
PMU_1V5_OVERLOAD_SC	9	w	**Overload at VDDC regulator Status clear** 0_B **No Clear**, Overload status not cleared 1_B **Clear**, Overload status cleared
PMU_1V5_OVERVOLT_SC	8	w	**Overvoltage at VDDC regulator Status clear** 0_B **No Clear**, Overvoltage status not cleared 1_B **Clear**, Overvoltage status cleared
RES	7	r	**Reserved** Always read as 0
PMU_5V_FAIL_EN	6	rw	**Enabling of VDDP status information as interrupt source** 0_B **Disable**, No interrupts are generated 1_B **Enable**, Interrupts are generated
PMU_5V_OVERLOAD	5	r	**Overload at VDDP regulator** This bit is set by hardware and can only be cleared by software. 0_B **No overload**, 1_B **Overload**,

Field	Bits	Type	Description
PMU_5V_OVERVOLT	4	r	**Overvoltage at VDDP regulator** This bit is set by hardware and can only be cleared by software. 0_B **No overvoltage**, 1_B **Overvoltage**,
PMU_OVERTEMP	3	r	**PMU Overtemperature** This bit is set by hardware and can only be cleared by software. 0_B **No overtemperature**, 1_B **Overtemperature**,
PMU_1V5_FAIL_EN	2	rw	**Enabling of VDDC status information as interrupt source** 0_B **Disable**, No interrupts are generated 1_B **Enable**, Interrupts are generated
PMU_1V5_OVERLOAD	1	r	**Overload at VDDC regulator** This bit is set by hardware and can only be cleared by software. 0_B **No overload**, 1_B **Overload**,
PMU_1V5_OVERVOLT	0	r	**Overvoltage at VDDC regulator** This bit is set by hardware and can only be cleared by software. 0_B **No overvoltage**, 1_B **Overvoltage**,

Table 7 RESET of PMU_SUPPLY_STS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

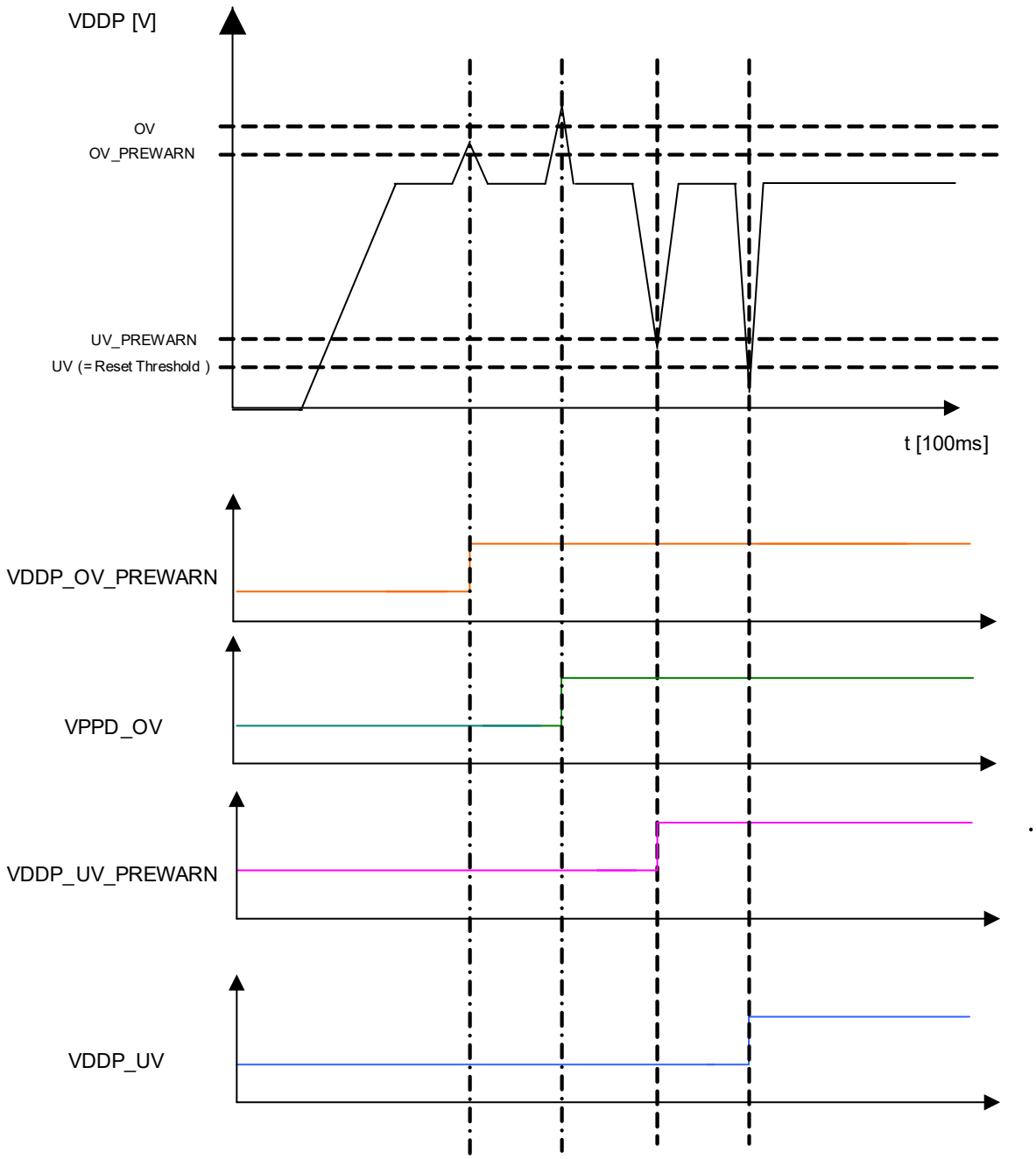


Figure 14 VDDP

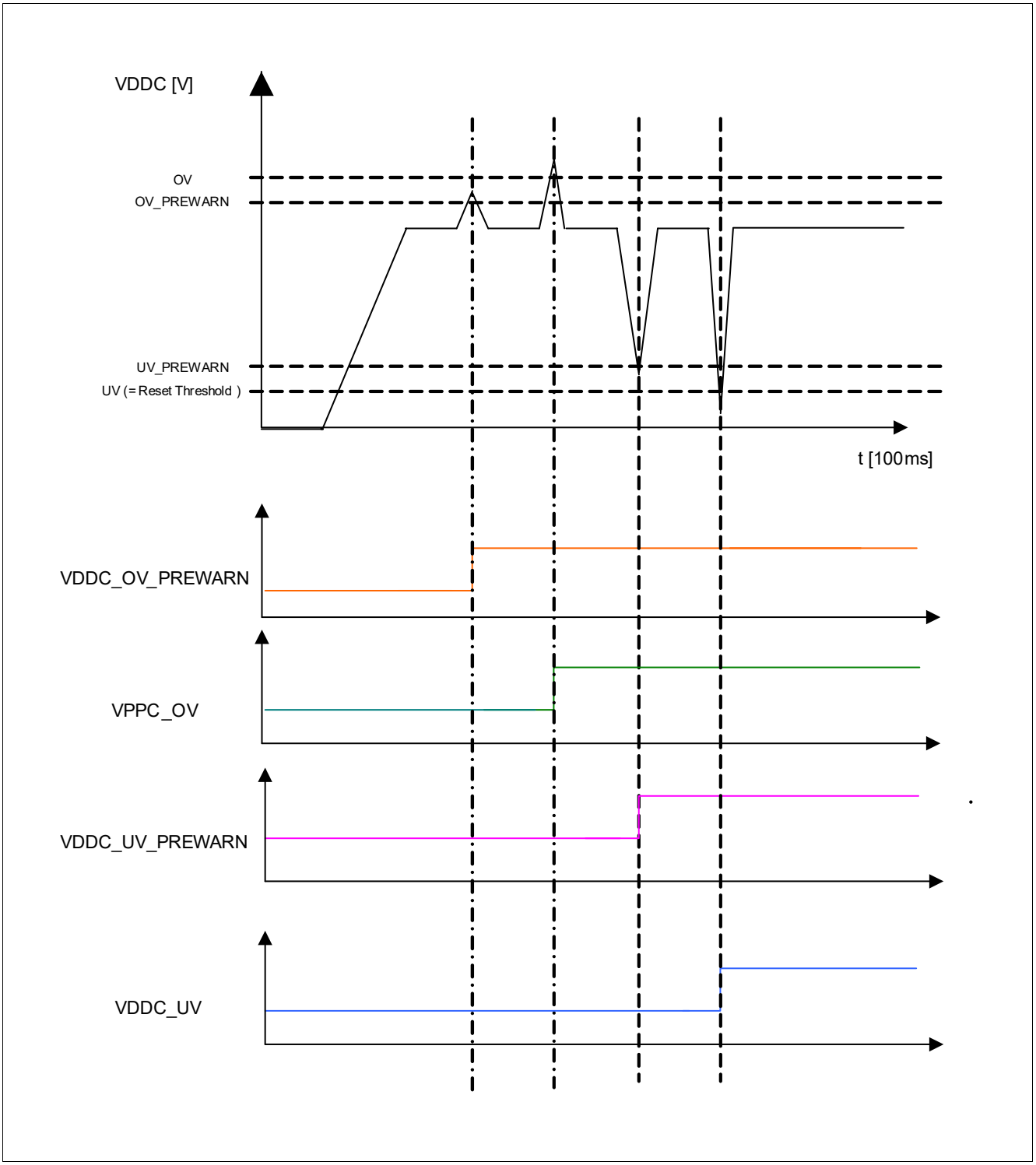


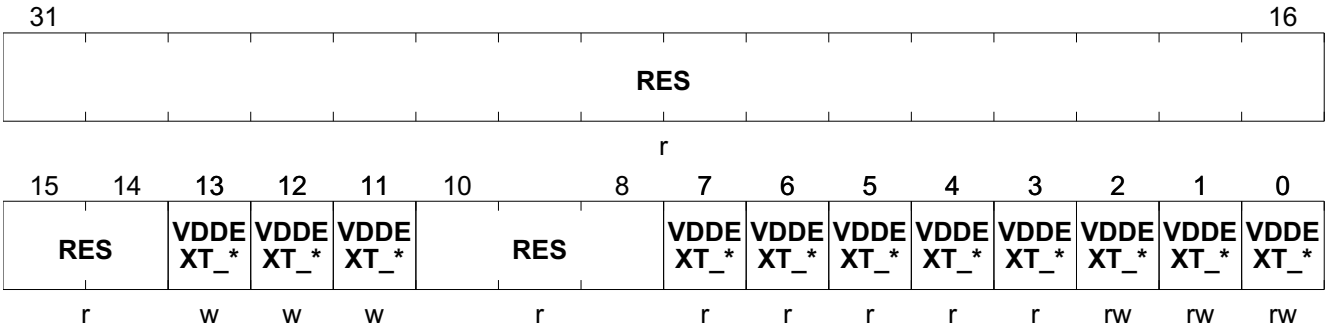
Figure 15 VDDC

6.3.6.2 VDDEXT Control Register

The VDDEXT can be fully controlled by the following SFR Register, including all diagnosis functions. .

VDDEXT Control

PMU_VDDEXT_CTRL	Offset	Reset Value
VDDEXT Control	$00C_H$	see Table 8



Field	Bits	Type	Description
RES	31:14	r	**Reserved** Always read as 0
VDDEXT_OT_SC	13	w	**VDDEXT Supply Overtemperature Status clear** 1_B **No Clear**, VDDEXT Overtemperature status not cleared 0_B **Clear**, VDDEXT Overtemperature status cleared
VDDEXT_UV_ISC	12	w	**VDDEXT Supply Undervoltage Interrupt Status clear** 1_B **No Clear**, VDDEXT Undervoltage not cleared 0_B **Clear**, VDDEXT Undervoltage cleared
VDDEXT_OT_ISC	11	w	**VDDEXT Supply Overtemperature Interrupt Status clear** 1_B **No Clear**, VDDEXT OverTemperature not cleared 0_B **Clear**, VDDEXT OverTemperature cleared
RES	10:8	r	**Reserved** Always read as 0
VDDEXT_STABLE	7	r	**VDDEXT Supply Stable** *Note: This bit is RESET_TYPE_3* 0_B **VDDEXT not in stable condition**, 1_B **VDDEXT in stable condition**,
VDDEXT_OT	6	r	**VDDEXT Supply Overtemperature** *Note: This bit is RESET_TYPE_3* 0_B **VDDEXT not in overtemperature condition**, 1_B **VDDEXT in overtemperature condition**,

Field	Bits	Type	Description
VDDEXT_OT_STS	5	r	**VDDEXT Supply Overtemperature Status** *Note: This bit is RESET_TYPE_3* 0_B **VDDEXT not in overtemperature condition**, 1_B **VDDEXT in overtemperature condition**,
VDDEXT_UV_IS	4	r	**VDDEXT Supply Undervoltage Interrupt Status** *Note: This bit is RESET_TYPE_4* 0_B **VDDEXT not in undervoltage condition**, 1_B **VDDEXT in undervoltage condition**,
VDDEXT_OT_IS	3	r	**VDDEXT Supply OverTemperature Interrupt Status** *Note: This bit is RESET_TYPE_4* 0_B **VDDEXT no overtemperature condition**, 1_B **VDDEXT overtemperature condition**,
VDDEXT_FAIL_EN	2	rw	**Enabling of VDDEXT Supply status information as interrupt source** *Note: This bit is RESET_TYPE_3* 0_B **Disable**, VDDEXT fail interrupts are disable 1_B **Enable**, VDDEXT fail Interrupts are enable
VDDEXT_CYC_EN	1	rw	**VDDEXT Supply for Cyclic Sense Enable** *Note: To use VDDEXT Supply for cyclic sense the bits VDDEXT_CYC_EN AND VDDEXT_ENABLE must be set.* *This bit is RESET_TYPE_3* 0_B **Disable**, VDDEXT for cyclic sense disable 1_B **Enable**, VDDEXT for cyclic sense enable
VDDEXT_ENABLE	0	rw	**VDDEXT Supply Enable** *Note: This bit is RESET_TYPE_3* 0_B **Disable**, VDDEXT Supply disable 1_B **Enable**, VDDEXT supply enable

Table 8 RESET of PMU_VDDEXT_CTRL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		
RESET_TYPE_4	00000000_H	RESET_TYPE_4		

6.4 Power Control Unit

The Power Control Unit is the controlling instance of the system power supply generation (PSG). It offers important fail safe features, which are described in the next chapter.

6.4.1 Power Control Unit - Fail Safe Scenarios

The PMU handles several different failure scenarios, listed below and described in the following chapters:

- Fail safe mode (Sleep Mode) in case of power failure .
- Fail safe mode (Sleep Mode) in case of 5 consecutive watchdog service failures.
- Fail safe mode (Sleep Mode) in case of overcurrent on voltage regulators VDDP or VDDC.
- 2 level monitoring (prewarning and reset) of voltage regulators output voltages (VDDP, VDDC).
- Wake-up from Stop Mode with cyclic sense in case of VDDEXT regulator failures.
- Wake-up from Stop Mode in case of hardware reset on RESET pin .

6.4.1.1 Power Supervision Function of PCU

The power supervision feature of the PCU is mainly responsible for monitoring the voltage regulators VDDP and VDDC. In case of voltage regulator malfunction the PCU tries first to recover to normal operation. If this is not possible the fail safe mode (sleep mode) is entered and the device is wakeable by the MONx inputs.

After a wake-up, if the PMU can be successfully restarted and code execution will be possible, the user is able to determine the occured failure scenario by checking the corresponding bits in the **PMU_WFS** register:

- **SUPP_TMOUT** is set if a timeout occured while waking up from stop mode and waiting for VDDP and VDDC to be stable.
- **SUPP_SHORT** is set if a short circuit at VDDP or VDDC is detected without being in VS undervoltage condition. Note: in active mode the PMU tries 5 times to restart the voltage regulators before entering sleep mode.

6.4.1.2 Watchdog (WDT1) Fail Safe

The PCU supervises the failure information of the system watchdog (WDT1). In case the watchdog is not serviced or serviced in a wrong way (in the following denominated as “not serviced Watchdog”) the MCU is reset and the error counter “error_wdt” is increased by one. The PMU itself stays in the Active Mode and after the reset the application software takes over the system control. If the software doesn’t service the system watchdog then the described procedure starts again. After the watchdog is not serviced five times during one Active Mode period the PMU sends the embedded system to Sleep Mode. The PMU detects the transition to the Sleep Mode as safety fallback and the Sleep Mode can be terminated by two ways: first by a LIN-wake or by a rising/falling edge at a MON pin, second cyclic wake is issued after a sleep time of 1 s. The error counter is reset when the system is sent to Sleep Mode or Stop Mode by a corresponding software command.

If the system can be successfully restarted, the cause of failure can be again checked by reading the **PMU_WFS** register. The bit **WDT1_SEQ_FAIL** signals the described failure.

6.4.1.3 Main Regulators Fail Safe

If one of the voltage regulators needs to deliver too much current, a stable operation of the supply voltage is not given. In this case the overcurrent detection of VDDP and VDDC will ensure that the system will enter Sleep Mode.

If the overcurrent condition is gone, a wake-up can be invoked, then the system will startup and work properly. Afterwards the corresponding failure flags **PMU_1V5_OVL** and PMU_5V_OVL can be checked.

6.4.1.4 VDDEXT Failure

If VDDEXT is used in combination with the GPIOs as a supply e.g. for the switches, there are several error cases possible, which are: Overtemperature, undervoltage. Those error cases may lead to the generation of false wake-up events or to missed wake-up events. To avoid these scenarios, errors on the VDDEXT voltage regulator would automatically revive the system from Stop Mode. The errors are signalled in the **PMU_WAKE_STATUS** register.

6.4.1.5 Wake-Up from Stop Mode with Reset Fail Safe

One fail safe measure to wake-up the embedded system from the Stop-Mode can be executed by hardware reset. If there is a reset request on the reset-pin then the PMU goes to Active Mode. Simultaneously, the embedded system gets a reset which is shown by forcing the bidirectional reset-pin. The reset-pin goes high again if the PMU releases the MCU reset. This event is shown in the reset status register as a hard-reset together with a wake-up reset. In case of a fail condition at one of the voltage regulators the PMU also goes to Active Mode. After that the PMU starts the supply fail-safe procedure which is described in the Active Mode section. The described sequence can be seen in the picture below.

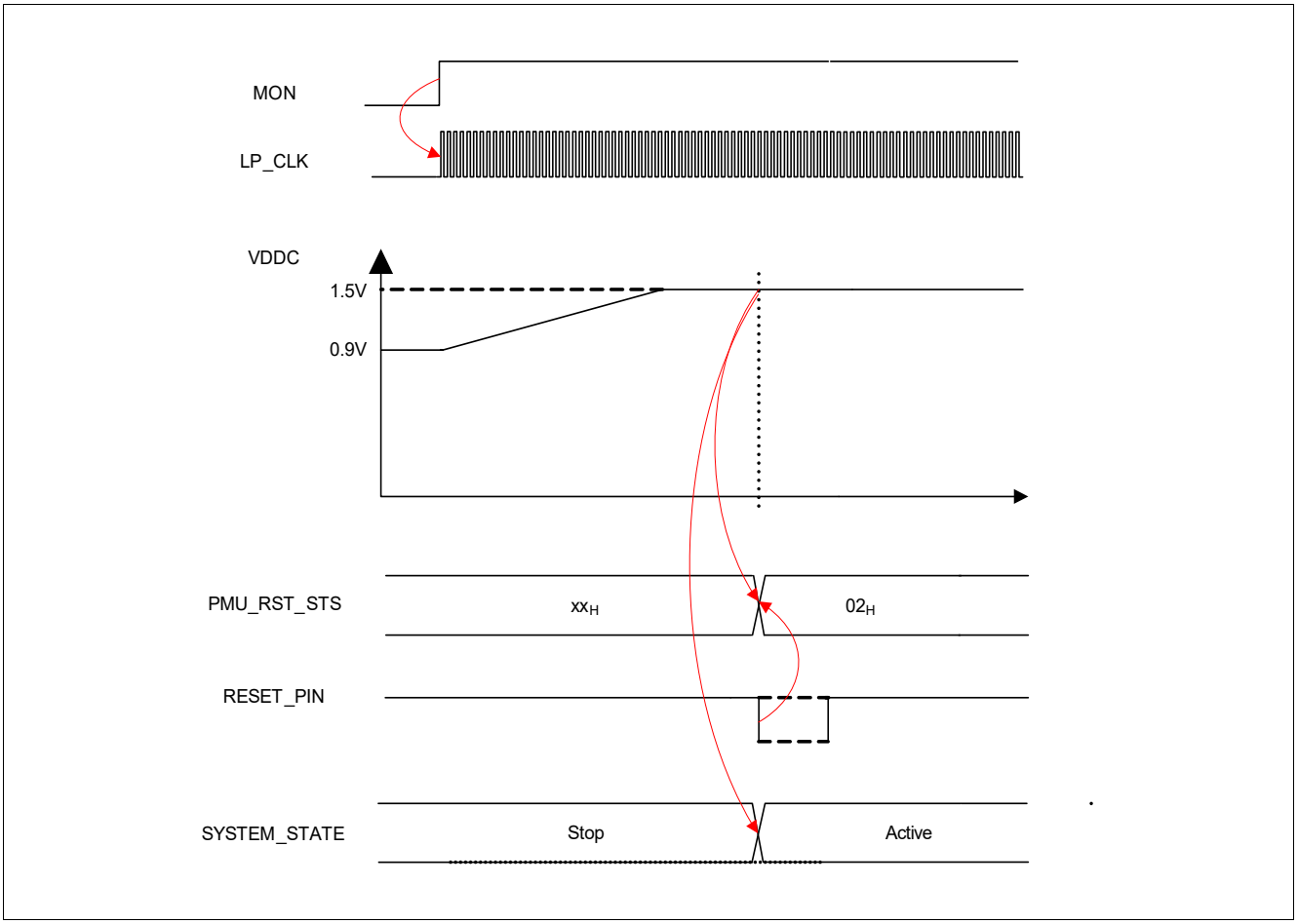


Figure 16 Stop Mode Exit Timing

6.4.1.6 Register Definition

Table 9 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Register Definition, PMU System Fail Register			
PMU_HIGHSIDE_CTRL	High-Side Control Register	$05C_H$	0000 0000_H
PMU_WFS	WFS System Fail Register	070_H	0000 0000_H
PMU_OT_CTRL	Overtemperature Control Register	054_H	0000 $000F_H$

The registers are addressed wordwise.

6.4.1.6.1 PMU System Fail Register

This register is dedicated for the control of the PMU Peripherals

WFS System Fail Register

Note: The register PMU_WFS is also cleared when **PMU_RESET_STS**.SYS_FAIL is cleared

PMU_WFS	Offset	Reset Value
WFS System Fail Register	**070_H**	**see Table 10**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RES															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RES							PMU_OT_*	LP_CLKWD	WDT1_SE*	SYS_OT	SYS_CLK*	PMU_5V_*	PMU_1V5*	SUPP_TM*	SUPP_SH*
r							rh	r	rh	rh	rh	rh	rh	rh	rh

Field	Bits	Type	Description
RES	31:9	r	**Reserved** Always read as 0
PMU_OT_FAIL	8	rh	**PMU Overtemperature Indication Flag** Indicates PMU Overtemperature Condition 0_B **No Overtemperature**, PMU ok 1_B **Overtemperature**, PMU Overtemperature

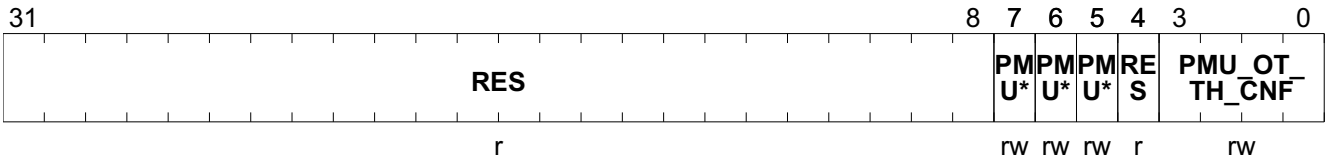
Field	Bits	Type	Description
LP_CLKWD	7	r	**LP_CLKWD** Low power clock 0_B **ok**, 1_B **fail**,
WDT1_SEQ_FAIL	6	rh	**External Watchdog (WDT1) Sequential Fail** Indicates that Watchdog is not serviced 5 times 0_B **No Fail**, System working properly 1_B **Sequential Watchdog Fail**, 5 consecutive watchdog fails
SYS_OT	5	rh	**System Overtemperature Indication Flag** Indicates System Overtemperature Condition 0_B **No Overtemperature**, System ok 1_B **Overtemperature**, System Overtemperature
SYS_CLK_WDT	4	rh	**System Clock (f_{sys})Watchdog Fail** Indicates a system clock watchdog fail 0_B **No System Clock Fail**, f_{sys} ok 1_B **System Clock Fail**, f_{sys} failed
PMU_5V_OVL	3	rh	**VDDP Overload Flag** Indicates Overload Condition at VDDP 0_B **No Overload**, VDDP ok 1_B **Overload**, VDDP Overload
PMU_1V5_OVL	2	rh	**VDDC Overload Flag** Indicates Overload Condition at VDDC 0_B **No Overload**, VDDC ok 1_B **Overload**, Hall VDDC Overload
SUPP_TMOUT	1	rh	**Supply Time Out** Indicates that a timeout occured while waking up from stop mode and waiting for VDDP and VDDC to be stable. 0_B **Main Supply ok**, VDDP or VDDC are in expected range 1_B **Main Supply fail**, VDDP or VDDC do not have stable operating point
SUPP_SHORT	0	rh	**Supply Short** Indicates that a short circuit at VDDP or VDDC was detected without being in VS undervoltage condition. 0_B **Main Supply ok**, VDDP or VDDC are in expected range 1_B **Main Supply short**, VDDP or VDDC are in short circuit condition

Table 10 RESET of PMU_WFS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_0	00000000_H	RESET_TYPE_0		

Overtemperature Control Register

PMU_OT_CTRL	Offset	Reset Value
Overtemperature Control Register	054_H	see Table 11



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Always read as 0
PMU_OT_EN	7	rw	**PMU Overtemperature Detection Enable** 0_B **Disable**, Overtemperature detection disabled 1_B **Enable**, Overtemperature detection enabled
PMU_OT_WAKE_EN	6	rw	**PMU Wake On Overtemperature Enable** 0_B **Disable**, no Wake-Up on OT condition 1_B **Enable**, Wake-Up on OT condition
PMU_OT_INT_EN	5	rw	**PMU Overtemperature Interrupt Enable** 0_B **No Interrupt**, Interrupt on OT disabled 1_B **Interrupt**, Interrupt on OT enabled
RES	4	r	**Reserved** Always read as 0
PMU_OT_TH_CNF	3:0	rw	**PMU Overtemperature threshold** 0000_B**0**, 131°C 0001_B**1**, 136°C 0010_B**2**, 141°C 0011_B**3**, 146°C 0100_B**4**, 152°C 0101_B**5**, 157°C 0110_B**6**, 163°C 0111_B**7**, 169°C 1000_B**8**, 175°C 1001_B**9**, 181°C 1010_B**10**, 187°C 1011_B**11**, 193°C 1100_B**12**, 200°C 1101_B**13**, 206°C 1110_B**14**, 214°C 1111_B**15**, 221°C

Table 11 RESET of PMU_OT_CTRL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_2	$0000000F_H$	RESET_TYPE_2		
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

High-Side Control Register

PMU_HIGHSIDE_CTRL	Offset	Reset Value
Highside Control Register	$05C_H$	see Table 12

Field	Bits	Type	Description
RES	31:3	r	**Reserved** Always read as 0
HS1_CYC_EN	2	rw	**High-Side 1 switch enable for cyclic sense** 0_B **Disable**, 1_B **Enable**,
RES	1:0	r	**Reserved** Always read as 0

Table 12 RESET of PMU_HIGHSIDE_CTRL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_2	00000000_H	RESET_TYPE_2		

6.5 Wake-up Management Unit (WMU)

The Wake-up Management Unit (WMU) is mainly responsible for handling the wake-up events on LIN, HV-Monitoring Inputs (MON1 - MON4), Hardware reset and all GPIOs belonging to Port 0 and Port 1. Following wake scenarios are possible:

- **Wake-up over Port 0 and Port 1 pins:** they can be configured for rising edge triggered and falling edge triggered wake-up events. This configuration can be used to wake-up the device from normal Stop Mode and Stop Mode with cyclic sense option. To bias the GPIOs, VDDEXT as current source can be used. The wake-up feature from Sleep Mode in combination with GPIOs is not possible.
- **Wake-up over Hardware reset pin:** It can be used to wake-up the device from Stop Mode. The wake-up feature from Sleep Mode is not possible.
- **Wake-up over MON1 - MON4 Pins:** the MONx Pins can be configured for rising edge triggered and falling edge triggered wake-up events. This setup can be used to wake-up the device from Stop Mode with or without cyclic sense, but also a wake-up from Sleep Mode with or without cyclic sense is possible.
- **LIN:** is a normal wake-up source and has no configuration possibilities.
- **Wake-up on VDDEXT fail from Stop Mode:** will be performed in case of VDDEXT failures described in Chapter **Power Control Unit - Fail Safe Scenarios**.

Note:

1. *Port 2 pins cannot invoke any wake-up.*

2. *None of the GPIOs is supplied during Sleep Mode, therefore wake-up is not possible through them.*

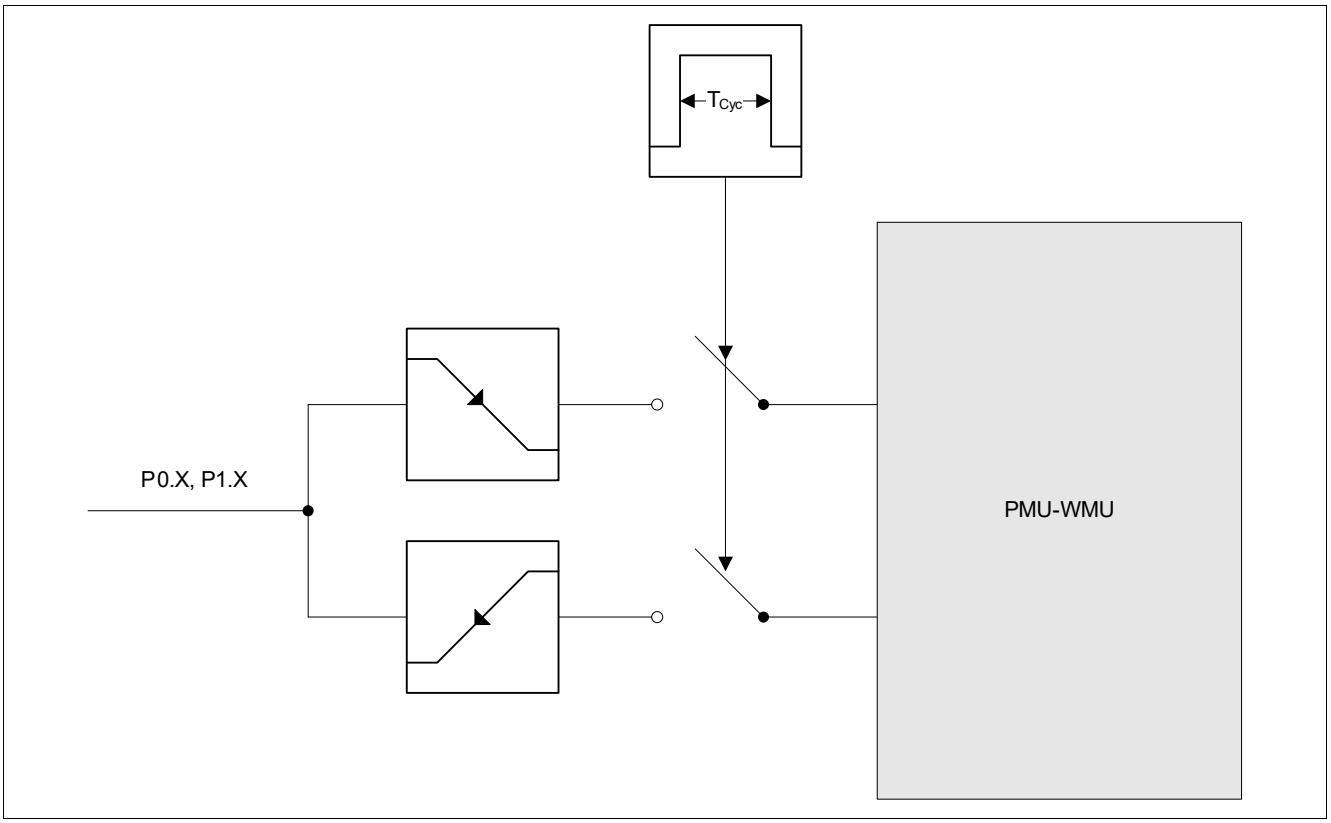


Figure 17 Block Diagram of Wake-up Management Unit in Cyclic Sense Mode with VDDEXT.

6.5.1 Register Definition

These registers are for wake-up control of all wake-up capable general purpose inputs outputs
The WMU is fully controllable by the below listed SFR Registers.

Table 13 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Register Definition, PMU Wake Up Configuration Register			
PMU_LIN_WAKE_EN	LIN Wake Enable	050_H	$0000\ 0000_H$
PMU_CNF_WAKE_FILTER	PMU Wake-up Timing Register	$0AC_H$	$0000\ 0000_H$
PMU_WAKE_CNF_GPIO0	Wake Configuration GPIO Port 0 Register	$0BC_H$	$0000\ 0000_H$
PMU_WAKE_CNF_GPIO1	Wake Configuration GPIO Port 1 Register	$0CC_H$	$0000\ 0000_H$
Register Definition, PMU Wake Up Status Register			
PMU_WAKE_STATUS	Main wake status register	000_H	$0000\ 0000_H$
Register Definition, GPIO Port Wake Up Status Register			
PMU_GPIO_WAKE_STATUS	GPIO Port wake status register	004_H	$0000\ 0000_H$

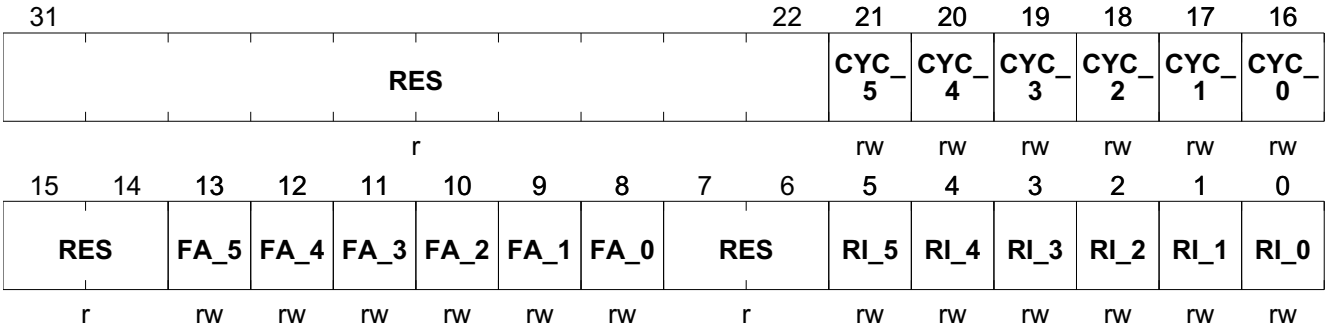
The registers are addressed wordwise.

6.5.1.1 PMU Wake Up Configuration Register

This register is dedicated for the control of the PMU Peripherals

Wake Configuration GPIO Port 0 Register

PMU_WAKE_CNF_GPIO0	Offset	Reset Value
Wake Configuration GPIO Port 0 Register	$0BC_H$	see Table 14



Field	Bits	Type	Description
RES	31:22	r	**Reserved** Always read as 0
CYC_5	21	rw	**GPIO0_5 input for cycle sense enable** 1_B **ENABLE**, input for cycle sense enabled 0_B **DISABLE**, input for cycle sense disabled
CYC_4	20	rw	**GPIO0_4 input for cycle sense enable** 1_B **ENABLE**, input for cycle sense enabled 0_B **DISABLE**, input for cycle sense disabled
CYC_3	19	rw	**GPIO0_3 input for cycle sense enable** 1_B **ENABLE**, input for cycle sense enabled 0_B **DISABLE**, input for cycle sense disabled
CYC_2	18	rw	**GPIO0_2 input for cycle sense enable** 1_B **ENABLE**, input for cycle sense enabled 0_B **DISABLE**, input for cycle sense disabled
CYC_1	17	rw	**GPIO0_1 input for cycle sense enable** 1_B **ENABLE**, input for cycle sense enabled 0_B **DISABLE**, input for cycle sense disabled
CYC_0	16	rw	**GPIO0_0 input for cycle sense enable** 1_B **ENABLE**, input for cycle sense enabled 0_B **DISABLE**, input for cycle sense disabled
RES	15:14	r	**Reserved** Always read as 0
FA_5	13	rw	**Port 0_5 Wake-up on Falling Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled

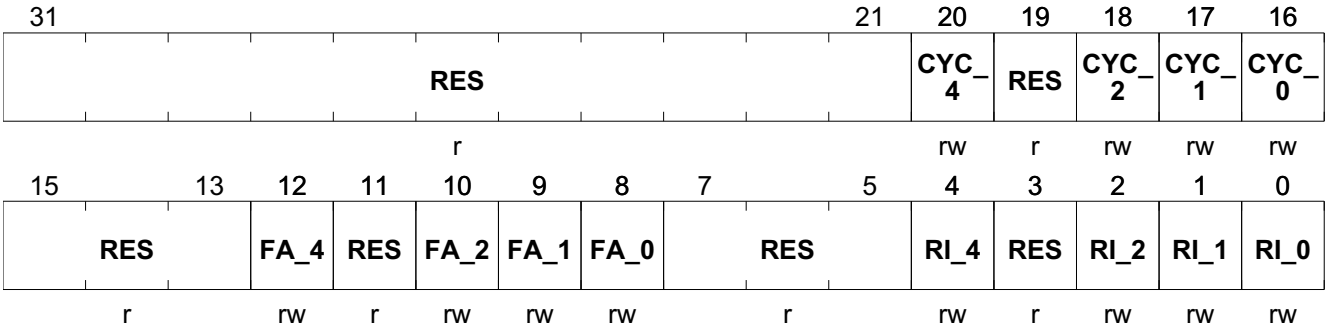
Field	Bits	Type	Description
FA_4	12	rw	**Port 0_4 Wake-up on Falling Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled
FA_3	11	rw	**Port 0_3 Wake-up on Falling Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled
FA_2	10	rw	**Port 0_2 Wake-up on Falling Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled
FA_1	9	rw	**Port 0_1 Wake-up on Falling Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled
FA_0	8	rw	**Port 0_0 Wake-up on Falling Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled
RES	7:6	r	**Reserved** Always read as 0
RI_5	5	rw	**Port 0_5 Wake-up on Rising Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled
RI_4	4	rw	**Port 0_4 Wake-up on Rising Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled
RI_3	3	rw	**Port 0_3 Wake-up on Rising Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled
RI_2	2	rw	**Port 0_2 Wake-up on Rising Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled
RI_1	1	rw	**Port 0_1 Wake-up on Rising Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled
RI_0	0	rw	**Port 0_0 Wake-up on Rising Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled

Table 14 RESET of PMU_WAKE_CNF_GPIO0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Wake Configuration GPIO Port 1 Register

PMU_WAKE_CNF_GPIO1	Offset	Reset Value
Wake Configuration GPIO Port 1 Register	$0CC_H$	see Table 15



Field	Bits	Type	Description
RES	31:21	r	**Reserved** Always read as 0
CYC_4	20	rw	**GPIO1_4 input for cycle sense enable** 1_B **ENABLE**, input for cycle sense enabled 0_B **DISABLE**, input for cycle sense disabled
RES	19	r	**Reserved** Always read as 0
CYC_2	18	rw	**GPIO1_2 input for cycle sense enable** 1_B **ENABLE**, input for cycle sense enabled 0_B **DISABLE**, input for cycle sense disabled
CYC_1	17	rw	**GPIO1_1 input for cycle sense enable** 1_B **ENABLE**, input for cycle sense enabled 0_B **DISABLE**, input for cycle sense disabled
CYC_0	16	rw	**GPIO1_0 input for cycle sense enable** 1_B **ENABLE**, input for cycle sense enabled 0_B **DISABLE**, input for cycle sense disabled
RES	15:13	r	**Reserved** Always read as 0
FA_4	12	rw	**Port 1_4 Wake-up on Falling Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled
RES	11	r	**Reserved** Always read as 0
FA_2	10	rw	**Port 1_2 Wake-up on Falling Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled
FA_1	9	rw	**Port 1_1 Wake-up on Falling Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled

Field	Bits	Type	Description
FA_0	8	rw	**Port 1_0 Wake-up on Falling Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled
RES	7:5	r	**Reserved** Always read as 0
RI_4	4	rw	**Port 1_4 Wake-up on Rising Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled
RES	3	r	**Reserved** Always read as 0
RI_2	2	rw	**Port 1_2 Wake-up on Rising Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled
RI_1	1	rw	**Port 1_1 Wake-up on Rising Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled
RI_0	0	rw	**Port 1_0 Wake-up on Rising Edge enable** 1_B **ENABLE**, wake-up enabled 0_B **DISABLE**, wake-up disabled

Table 15 RESET of PMU_WAKE_CNF_GPIO1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

PMU Wake-up Timing Register

These registers are for wake-up control of all wake-up capable general purpose inputs outputs

PMU_CNF_WAKE_FILTER	Offset	Reset Value
PMU Wake-up Timing Register	$0AC_H$	see Table 16

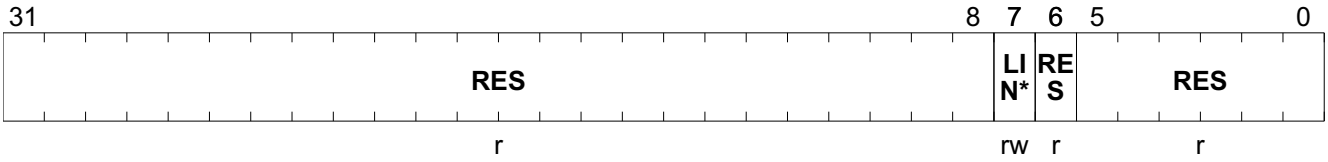
Field	Bits	Type	Description
RES	31:4	r	**Reserved** Always read as 0.
CNF_GPIO_FT	3:2	rw	**Wake-up Filter time for General Purpose IO** Selects the filter time for the Wake-up 00_B **10_us**, 10 µs filter time 01_B **20_us**, 20 µs filter time 10_B **40_us**, 40 µs filter time 11_B **5_us**, 5 µs filter time
CNF_MON_FT	1	rw	**Wake-up Filter time for Monitoring Inputs** Selects the filter time for the Wake-up 0_B **20_us**, 20 µs filter time 1_B **40_us**, 40 µs filter time
CNF_LIN_FT	0	rw	**Wake-up Filter time for LIN WAKE** Selects the filter time for the Wake-up 0_B **50_us**, 50 µs filter time 1_B **30_us**, 30 µs filter time

Table 16 RESET of PMU_CNF_WAKE_FILTER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_2	00000000_H	RESET_TYPE_2		

LIN Wake Enable

PMU_LIN_WAKE_EN	Offset	Reset Value
LIN Wake Enable	050_H	see Table 17



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Always read as 0
LIN_WAKE_EN	7	rw	**Lin Wake enable** 0_B **Wake Disabled**, 1_B **Wake enabled**,
RES	6	r	**Reserved** Always read as 0
RES	5:0	r	**Reserved** Always read as 0

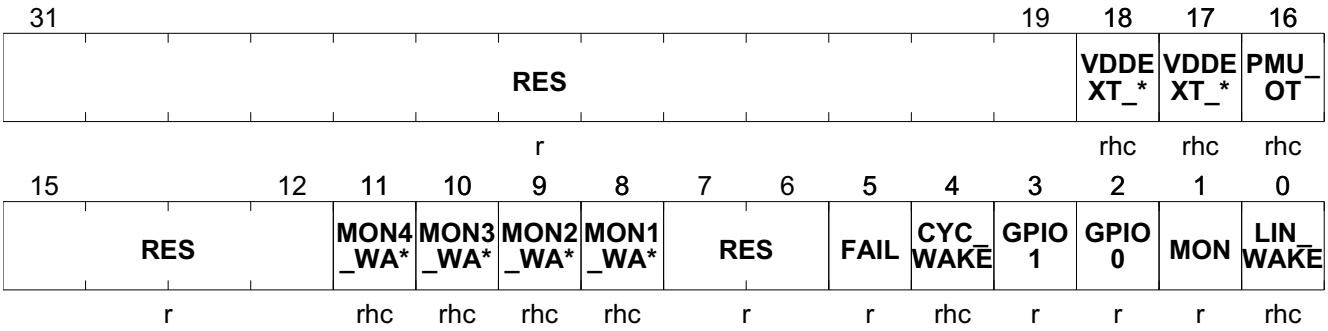
Table 17 RESET of PMU_LIN_WAKE_EN

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_2	00000000_H	RESET_TYPE_2		

6.5.1.2 PMU Wake Up Status Register

Main wake status register

PMU_WAKE_STATUS	Offset	Reset Value
Main wake status register	**000_H**	**see Table 18**



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0

Field	Bits	Type	Description
VDDEXT_UV	18	rhc	**Wake VDDEXT Undervoltage** *Note: this register is cleared automatically by read operation* 0_B **No wake-up detected**, 1_B **wake-up detected**,
VDDEXT_OT	17	rhc	**Wake VDDEXT Overtemperature** *Note: this register is cleared automatically by read operation* 0_B **No wake-up detected**, 1_B **wake-up detected**,
PMU_OT	16	rhc	**Wake PMU Overtemperature** *Note: this register is cleared automatically by read operation* 0_B **No wake-up detected**, 1_B **wake-up detected**,
RES	15:12	r	**Reserved** Always read as 0
MON4_WAKE_STS	11	rhc	**Status of MON4** *Note: this register is cleared automatically by read operation* 0_B **No wake-up detected**, 1_B **wake-up detected**,
MON3_WAKE_STS	10	rhc	**Status of MON3** *Note: this register is cleared automatically by read operation* 0_B **No wake-up detected**, 1_B **wake-up detected**,
MON2_WAKE_STS	9	rhc	**Status of MON2** *Note: this register is cleared automatically by read operation* 0_B **No wake-up detected**, 1_B **wake-up detected**,

Field	Bits	Type	Description
MON1_WAKE_STS	8	rhc	**Status of MON1** *Note: this register is cleared automatically by read operation* 0_B **No wake-up detected**, 1_B **wake-up detected**,
RES	7:6	r	**Reserved** Always read as 0
FAIL	5	r	**Wake-up after any Fail, which is a logical OR combination of PMU_OT, VDDEXT_OT, VDDEXT_UV** 0_B **No Wake-up occurred**, 1_B **Wake-up occurred**,
CYC_WAKE	4	rhc	**Wake-up caused by Cyclic Wake** *Note: this register is cleared automatically by read operation* 0_B **No Wake-up occurred**, 1_B **Wake-up occurred**,
GPIO1	3	r	**Wake-up via GPIO1 which is a logical OR combination of all Wake_STS_GPIO1 bits** 0_B **No Wake-up occurred**, 1_B **Wake-up occurred**,
GPIO0	2	r	**Wake-up via GPIO0 which is a logical OR combination of all Wake_STS_GPIO0 bits** 0_B **No Wake-up occurred**, 1_B **Wake-up occurred**,
MON	1	r	**Wake-up via MON which is a logical OR combination of all Wake_STS_MON bits** 0_B **No Wake-up occurred**, 1_B **Wake-up occurred**,
LIN_WAKE	0	rhc	**Wake-up via LIN- Message** *Note: this register is cleared automatically by read operation* 0_B **No Wake-up occurred**, 1_B **Wake-up occurred**,

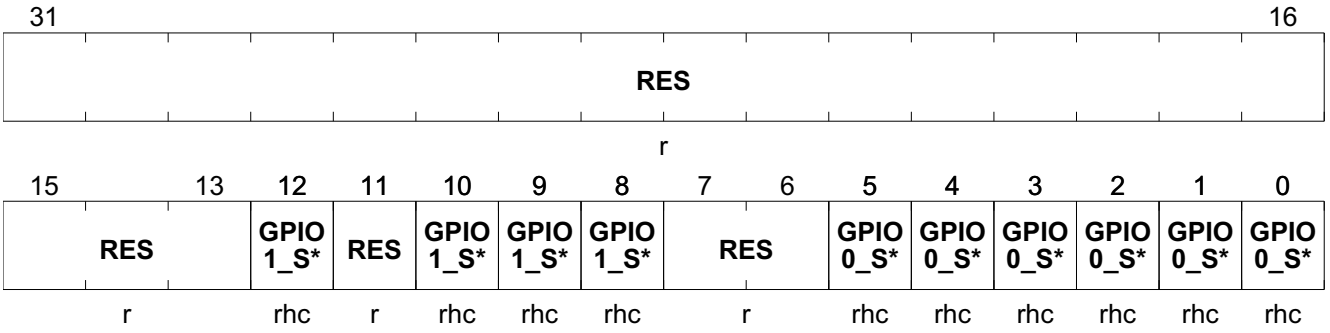
Table 18 RESET of PMU_WAKE_STATUS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_1	00000000_H	RESET_TYPE_1		

6.5.1.3 GPIO Port Wake Up Status Register

GPIO Port wake status register

PMU_GPIO_WAKE_STATUS	Offset	Reset Value
GPIO Port wake status register	004_H	see Table 19



Field	Bits	Type	Description
RES	31:13	r	**Reserved** Always read as 0
GPIO1_STS_4	12	rhc	**Wake GPIO1_4** *Note: This flag is cleared by read operation.* 0_B **No wake-up detected**, 1_B **wake-up detected**,
RES	11	r	**Reserved** Always read as 0
GPIO1_STS_2	10	rhc	**Wake GPIO1_2** *Note: This flag is cleared by read operation.* 0_B **No wake-up detected**, 1_B **wake-up detected**,
GPIO1_STS_1	9	rhc	**Wake GPIO1_1** *Note: This flag is cleared by read operation.* 0_B **No wake-up detected**, 1_B **wake-up detected**,
GPIO1_STS_0	8	rhc	**Wake GPIO1_0** *Note: This flag is cleared by read operation.* 0_B **No wake-up detected**, 1_B **wake-up detected**,
RES	7:6	r	**Reserved** Always read as 0

Field	Bits	Type	Description
GPIO0_STS_5	5	rhc	**Status of GPIO0_5** *Note: This flag is cleared by read operation.* 0_B **No wake-up detected**, 1_B **wake-up detected**,
GPIO0_STS_4	4	rhc	**Status of GPIO0_4** *Note: This flag is cleared by read operation.* 0_B **No wake-up detected**, 1_B **wake-up detected**,
GPIO0_STS_3	3	rhc	**Status of GPIO0_3** *Note: This flag is cleared by read operation.* 0_B **No wake-up detected**, 1_B **wake-up detected**,
GPIO0_STS_2	2	rhc	**Status of GPIO0_2** *Note: This flag is cleared by read operation.* 0_B **No wake-up detected**, 1_B **wake-up detected**,
GPIO0_STS_1	1	rhc	**Status of GPIO0_1** *Note: This flag is cleared by read operation.* 0_B **No wake-up detected**, 1_B **wake-up detected**,
GPIO0_STS_0	0	rhc	**Status of GPIO0_0** *Note: This flag is cleared by read operation.* 0_B **No wake-up detected**, 1_B **wake-up detected**,

Table 19 RESET of PMU_GPIO_WAKE_STATUS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_1	00000000_H	RESET_TYPE_1		

6.6 Cyclic Management Unit (CMU)

The cyclic management unit is responsible for controlling the timing sequence in cyclic sense or cyclic wake operation. The unit operates with the LP_CLK2 clock.

6.6.1 Cyclic Sense Mode

To select a dedicated MONx pin for cyclic sense mode, the bits MONx_EN, MONx_CYC, and on or both of the MONx_RISE and MONx_FALL bits must be set in the **PMU_MON_CNF1** register. In this configuration the wake-up information of this MON pin is only accepted during the sensing time where the HS_CYC_ON (internal HSx_ON gating signal) is high (see **Figure 18**). The activation of the cyclic sense mode and the sensing time where the enable signal is active, will be set in the **PMU_SLEEP** register. The flags inside the **PMU_SLEEP** register are used to enable the cyclic sense mode (CYC_SENSE_EN) and to configure the dead time (CYC_SENSE_M03, CYC_SENSE_E01: T_{Dead}) and the sample delay of the wake inputs and thus the on-time (CYC_SENSE_S_DEL: T_{On}).

After a valid wake-up event the start-up sequence is similar to the asynchronous wake-up and the system enters the Start-up Mode automatically too. If the PMU detects a wake-up during Cyclic Sense then the enable signal of the current source (HS) stays active as long the application software doesn't disable these signals.

Figure 18 illustrates the principle of the cyclic sense mode. Here a High Side switch is used as current source together with a MONx pin as a wake-up source. The same timing flow can also be applied for cyclic operation with VDDEXT and all GPIOs from Port 0 and Port 1.

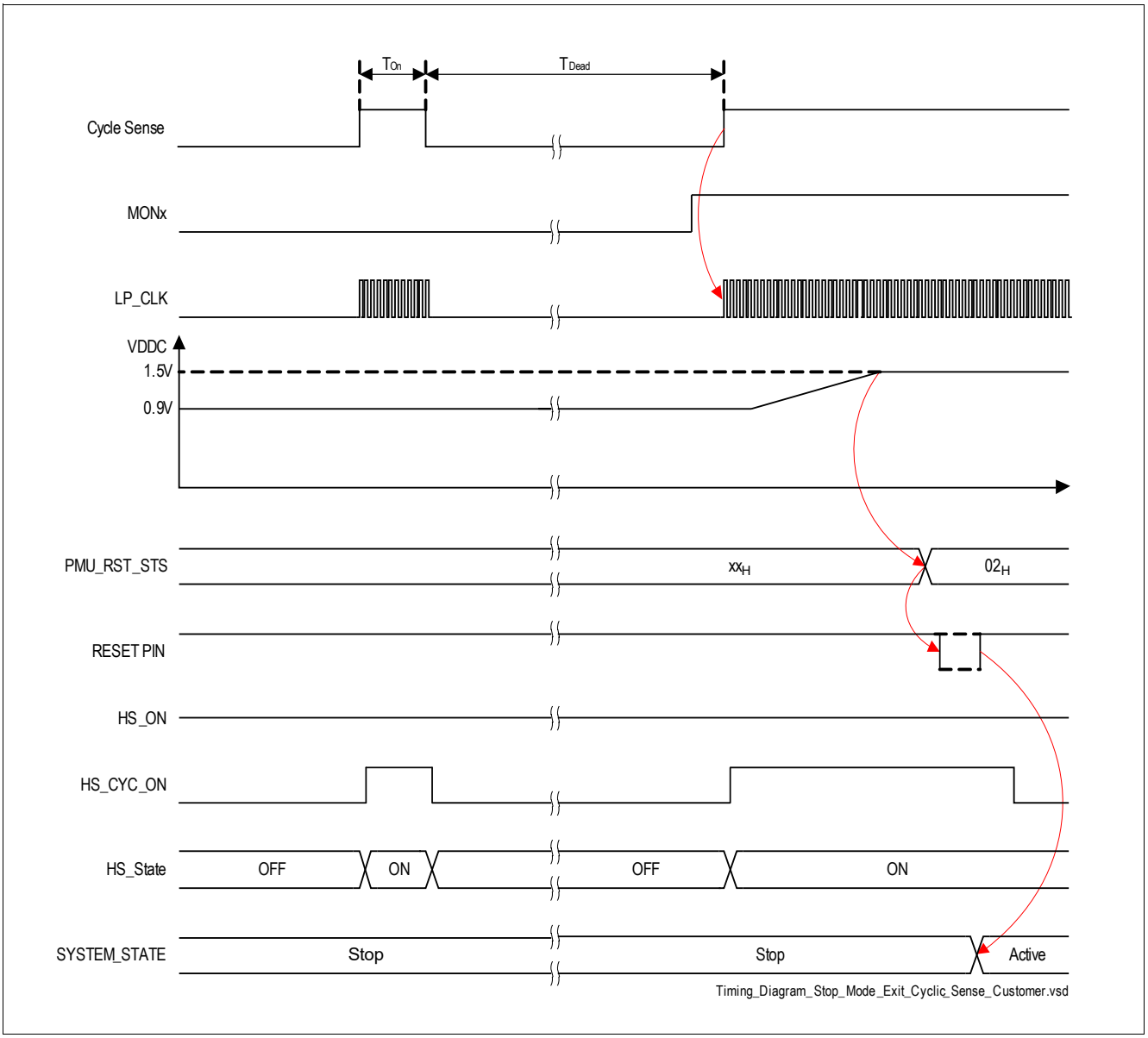


Figure 18 Timing Diagram for Cyclic Sense

6.6.1.1 Configuration of Cyclic Sense Mode

The configuration of cyclic sense mode is shown in **Figure 19**.

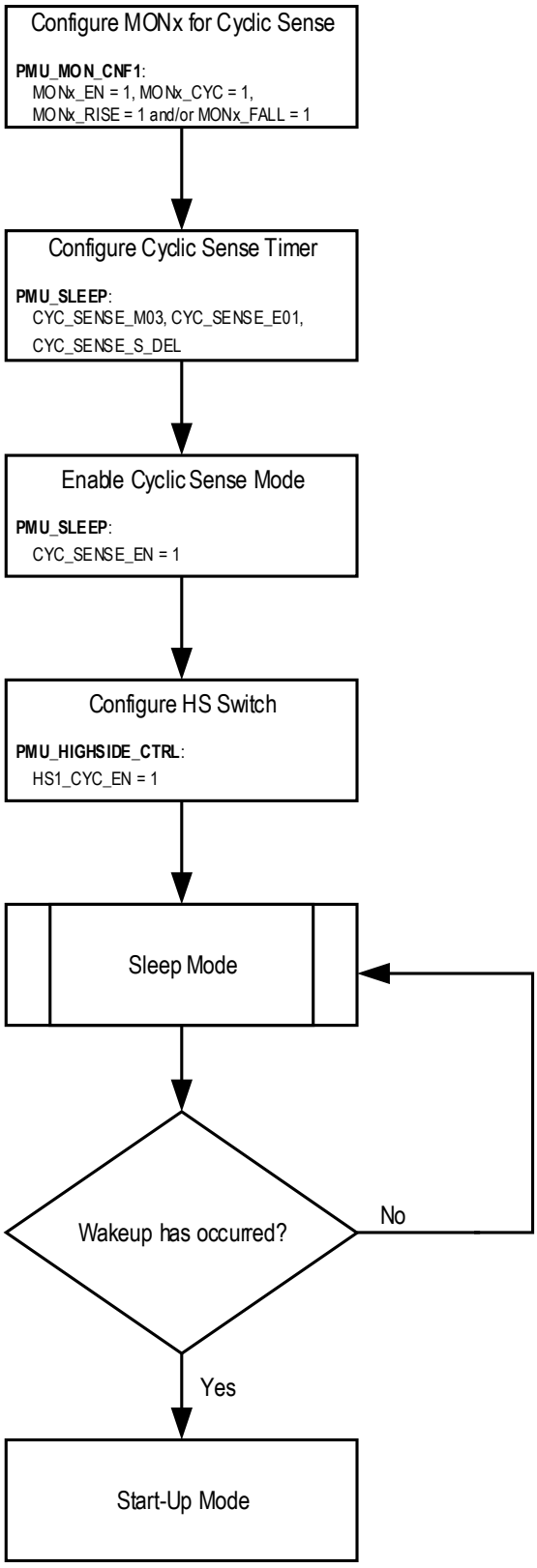


Figure 19 Configuration Flow of cyclic sense mode

6.6.2 Cyclic Wake Mode

Cyclic Wake mode provides a synchronous wake-up after a predefined time interval in Sleep Mode or Stop Mode. Once the time interval is elapsed the PMU enters the Startup Mode and proceeds to Active Mode where the software takes over the system control. The cyclic wake interval is set in the **PMU_SLEEP**-XSFR.

6.6.3 Register Definition

Table 20 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Register Definition, Cyclic Mode Configuration Registers (CYCMU)			
PMU_SLEEP	PMU Sleep Behavior Register	020_H	$0037\ 0004_H$
PMU_DRV_CTRL	PMU Bridge Driver Control	024_H	$0000\ 0050_H$

The registers are addressed wordwise.

6.6.3.1 Cyclic Mode Configuration Registers (CYCMU)

Cyclic Sense Mode Configuration

The off time (dead time) in Cyclic Sense Mode is calculated by following formula:

$$4^{(E1E0)} \cdot (M3M2M1M0+1) \cdot 2ms$$

where E1E0 represents the exponent, which can be configured by the register bits PMU_SLEEP.CYC_SENSE_E01<1:0>. M3M2M1M0 represents the mantissa configurable by the register bits PMU_SLEEP.CYC_SENSE_M03<3:0>. With this setting a time range between

- minimum 2 ms and
- maximum 2048 ms

can be configured. In addition to the off time (dead time) a sample delay for the sensing period can be configured. The sample delay applies after the corresponding supply (HS/VDDEXT) used in the cyclic mode is turned on to the time when the wake inputs (MONx/GPIOx) are sensed. The delay time can be configured in the PMU_SLEEP.CYC_SENSE_S_DEL register.

Cyclic Wake Mode Configuration

The off time (dead time) in Cyclic Wake Mode is calculated by following formula:

$$4^{(E1E0)} \cdot (M3M2M1M0+1) \cdot 2ms$$

where E1E0 represents the exponent, which can be configured by the register bits PMU_SLEEP.CYC_WAKE_E01<1:0>. M3M2M1M0 represents the mantissa configurable by the register bits PMU_SLEEP.CYC_WAKE_M03<3:0>. With this setting a time range between

- minimum 2 ms and
- maximum 2048 ms

can be configured.

Note: all timings in the cyclic modes are derived from LP_CLK2. The values used in the register description are typical values. Their variation is depending on the variation of LP_CLK2.

PMU Sleep Behavior Register

PMU_SLEEP	Offset	Reset Value
PMU Sleep Behavior Register	020_H	see Table 21

31	27	26	24	23	22	21	20	19	16
RES		CYC_SENSE_S_DEL		RES		CYC_WAKE_E01		CYC_WAKE_M03	
r		rw		r		rw		rw	

15	14	13	12	11	8	7	4	3	2	1	0
RES	RES	CYC_SENSE_E01		CYC_SENSE_M03		RES		CYC_SEN*	CYC_WAK*	EN_0V9_N	WAKE_W_*
r	r	rw		rw		r		rw	rw	rw	rw

Field	Bits	Type	Description
RES	31:27	r	**Reserved** Always read as 0
CYC_SENSE_S_DEL	26:24	rw	**Sample Delay in Cyclic Sense** Delay time after HS/VDDEXT is turned on to the time when MONx/GPIOx are sensed. 000_B **0**, 18 µs typ. 001_B **1**, 27 µs typ. 010_B **2**, 36 µs typ. 011_B **3**, 45 µs typ. 100_B **4**, 63 µs typ. 101_B **5**, 81 µs typ. 110_B **6**, 99 µs typ. 111_B **7**, 144 µs typ.
RES	23:22	r	**Reserved** Always read as 0
CYC_WAKE_E01	21:20	rw	**Exponent** 00_B **0**, Exponent value is 0 01_B **1**, Exponent value is 1 10_B **2**, Exponent value is 2 11_B **3**, Exponent value is 3
CYC_WAKE_M03	19:16	rw	**Mantissa** Mantissa value is calculated as CYC_WAKE_M03 +1 0000_B**1**, Mantissa value is 1 1111_B**16**, Mantissa value is 16
RES	15	r	**Reserved** Always read as 0
RES	14	r	**Reserved** Always read as 0
CYC_SENSE_E01	13:12	rw	**Exponent** 00_B **0**, Exponent value is 0 01_B **1**, Exponent value is 1 10_B **2**, Exponent value is 2 11_B **3**, Exponent value is 3
CYC_SENSE_M03	11:8	rw	**Mantissa** Mantissa value is calculated as CYC_SENSE_M03 +1 0000_B**1**, Mantissa value is 1 1111_B**16**, Mantissa value is 16
RES	7:4	r	**Reserved** Always read as 0
CYC_SENSE_EN	3	rw	**Enabling Cyclic Sense** This bit enables the cyclic sense feature for the power save modes. 0_B **Disable**, Cyclic Sense disabled 1_B **Enable**, Cyclic Sense enabled

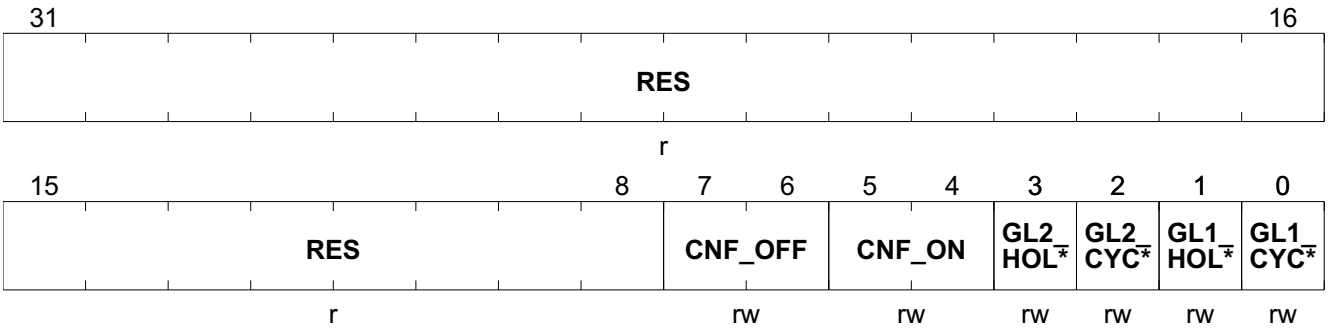
Field	Bits	Type	Description
CYC_WAKE_EN	2	rw	**Enabling Cyclic Wake** This bit enables the cyclic wake feature for the power save modes. 0_B **Disable**, Cyclic Wake disabled 1_B **Enable**, Cyclic Wake enabled
EN_0V9_N	1	rw	**Enables the reduction of the VDDC regulator output to 0.9 V during Stop-Mode** 0_B **Enable**, Output voltage reduction enabled 1_B **Disable**, Output voltage reduction disabled
WAKE_W_RST	0	rw	**Wake-up with reset execution** Enables the Stop-Exit with reset execution 0_B **No Reset**, Stop-Exit without reset execution 1_B **Reset**, Stop-Exit with reset execution

Table 21 RESET of PMU_SLEEP

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_2	01370004_H	RESET_TYPE_2		

Bride Driver Control

PMU_DRV_CTRL	Offset	Reset Value
PMU Bridge Driver Control	024_H	see Table 22



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Always read as 0
CNF_OFF	7:6	rw	**CNF_OFF Function** Cyclic off time for GL1_CYC_ON and GL2_CYC_ON 00_B **400 us**, 01_B **800 us**, 10_B **2000 us**, 11_B **4000 us**,

Field	Bits	Type	Description
CNF_ON	5:4	rw	**CNF_ON Function** Cyclic on time for GL1_CYC_ON and GL2_CYC_ON 00_B **50 us**, 01_B **100 us**, 10_B **200 us**, 11_B **400 us**,
GL2_HOLD_ON	3	rw	**GL2 Hold Mode On** This bit enables the hold mode of GL2 0_B **Disable**, 1_B **Enable**,
GL2_CYC_ON	2	rw	**GL2 Cyclic On** This bit enables the cyclic on at GL2 0_B **Disable**, 1_B **Enable**,
GL1_HOLD_ON	1	rw	**GL1 Hold Mode On** This bit enables the hold mode of GL1 0_B **Disable**, 1_B **Enable**,
GL1_CYC_ON	0	rw	**GL1 Cyclic On** This bit enables the cyclic on at GL1 0_B **Disable**, 1_B **Enable**,

Table 22 RESET of PMU_DRV_CTRL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_0	00000050_H	RESET_TYPE_0		

6.7 Reset Management Unit (RMU)

The RMU controls the reset behavior of the entire device. The master reset of the device is the power-on reset of the PMU itself. This reset is generated by the Power Down Supply and it is released when the battery voltage (Vs) reaches the minimum supply voltage for Active Mode. Then the PMU starts the sequence to power-up the supply generation module which ends with the release of the MCU reset. If this status is reached then the embedded system will work in Active Mode. This scenario is signalled by the PMU_VS_POR flag in the **PMU_RESET_STS**. The figure below shows the power-on reset behavior.

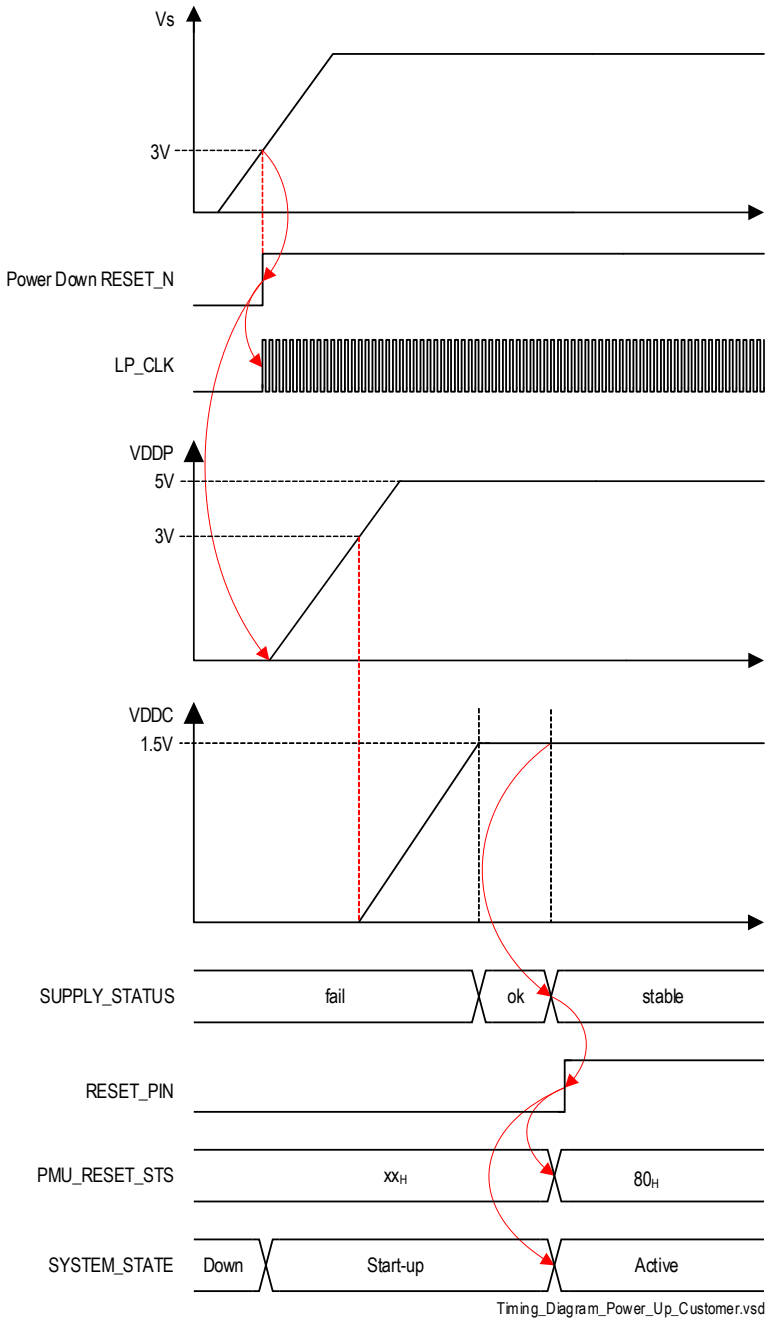


Figure 20 Power-On and Startup Behavior of Reset

In case of a Sleep Mode exit a similar sequence used for battery ramp-up starts. If this sequence ends successfully then the PMU also releases the reset of the MCU. From the MCU point of view there is no difference to the battery ramp-up. Only inside of the RMU the identification bit **PMU_SleepEx** is set instead of the power-on identification bit.

In the default configuration the wake-up from Stop Mode works without reset. To wake-up with reset the corresponding **SFR** bit WAKE_W_RST inside the **PMU_SLEEP** register must be configured. With this configuration the wake-up signal sets the dedicated identification bit PMU_WAKE which can be checked by the application software.

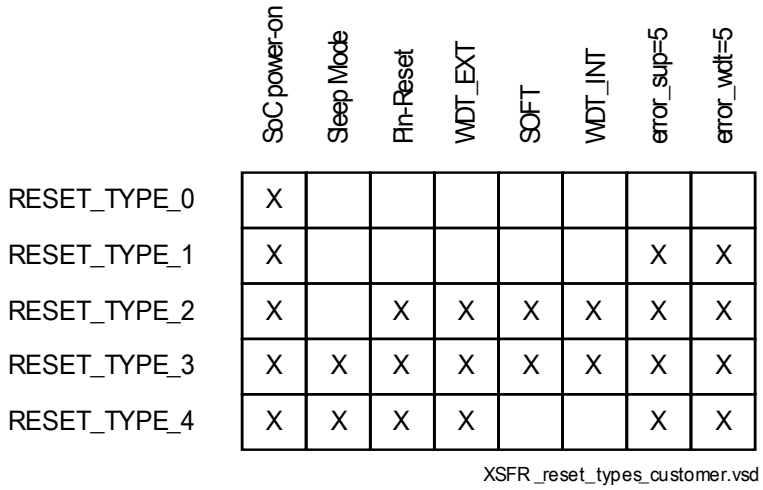
The third hardware related reset source is the pin-reset. The pad itself is supplied by the VDDP domain which is available in Active Mode and Stop Mode. Therefore the reset-pin can be used in Active Mode and Stop Mode only. Due to the bidirectional use of the pin itself the pin-reset request is gated during the execution of another reset request (e.g. soft-reset). For this purpose the pin-reset request must be stable for more than 500 ns (see **Figure 18**). In case of a pin-reset request during Stop Mode the PMU goes to Active Mode and sends the wake-up signal to the MCU. At this time the reset status register also gets an update by setting bit PMU_PIN, which signals the described reset source. All other reset sources can only have an impact on the system behavior in Active Mode.

The reset request caused by a system watchdog, which was not serviced is also processed as a hardware related reset although this reset request is implicitly controlled by user software. The system watchdog only works in Active Mode. In this case it expects a periodic trigger (window watchdog) from the user software. If the trigger is missing then the PMU gets the signal that the watchdog was not serviced which sets the identification bit PMU_ExtWDT from WDT1. After some clock cycles of the PMU internal oscillator LP_CLK the PMU resets the MCU. The prioritization of the described reset sources is done according to the architecture and the functionality of the embedded system itself.

The software-reset and the reset request caused by the MCU internal watchdog are controlled explicitly by user software and can be used only in Active Mode. From the system point of view both of these reset sources have the lowest priority. The software related reset is executed within two MCU clock cycles which is required by the CPU architecture. The system clock of the PMU works independently of the MCU clock. Due to these system conditions the PMU processes the software related resets asynchronously to its internal system clock. The software-reset is flagged by the PMU_SOFT bit. The MCU internal watchdog is signalled by the PMU_IntWDT bit. Both flags are located in the above mentioned **PMU_RESET_STS** register.

Another reset source is the PSG module. In case the main voltage regulators (VDDP and VDDC) will fail, the system will execute a system reset and enter Sleep Mode afterwards. This case is flagged by setting the indication bit SYS_FAIL.

Reset types are combinations of the above described resets. The reset of an XSFR register is depending on the corresponding reset type. Other registers (all SFRs except NMI status flags) are always reset independent of the reset type. The figure below shows this combination of resets.



	SoC power-on	Sleep Mode	Pin-Reset	WDT_EXT	SOFT	WDT_INT	error_sup=5	error_wdt=5
RESET_TYPE_0	X							
RESET_TYPE_1	X						X	X
RESET_TYPE_2	X		X	X	X	X	X	X
RESET_TYPE_3	X	X	X	X	X	X	X	X
RESET_TYPE_4	X	X	X	X			X	X

Figure 21 Reset Types of SFRS provided by the RMU

Out of these above listed resets mainly five reset types are derived:

- **RESET_TYPE_0** contains:
  - PMU_VS_POR: this reset is issued when the power down supply detects undervoltage
- RESET_TYPE_1 is an OR of:
  - PMU_VS_POR
  - PMU_FAIL: this reset is issued when the VDDC or VDDP supply have a failure
  - WDT_FAIL: this reset is issued when WDT1 is not triggered consecutively 5 times properly
- RESET_TYPE_2 is an OR of:
  - PMU_VS_POR
  - PMU_PIN: this reset is issued when the RESET-Pin is pulled down
  - PMU_ExtWDT: this reset is a WDT1 related reset
  - PMU_IntWDT: this reset is an internal WDT issued reset
  - PMU_SOFT: this reset is a software related reset
  - PMU_Wake: this reset is a stop wake-up related reset
  - PMU_FAIL
  - WDT_FAIL
- RESET_TYPE_3 is an OR of:
  - PMU_VS_POR
  - PMU_PIN
  - PMU_ExtWDT
  - PMU_IntWDT
  - PMU_SOFT
  - PMU_Wake
  - PMU_SleepEx: this reset is a sleep wake-up related reset

- PMU_FAIL
- WDT_FAIL

Every register has its own reset type listed. In the Power Management Unit SFRs following reset types are used:

- RESET_TYPE_0
- RESET_TYPE_1
- RESET_TYPE_2
- RESET_TYPE_3

6.7.1 Register Definition

Table 23 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Register Definition, Reset Management Unit Registers (RMU)			
PMU_RESET_STS	Reset Status Register	010_H	$0000\ 0000_H$
PMU_CNF_RST_TFB	Reset Blind Time Register	$06C_H$	$0000\ 0001_H$

The registers are addressed wordwise.

6.7.1.1 Reset Management Unit Registers (RMU)

The Reset Pin is a bidirectional signal. Every reset will be signaled on that pin for a few 100 ns. In order to avoid any reset deadlock situation there is a programmable reset blind time, where no hardware pin reset will be recognized. The reset blind time envelopes the phase, where the reset pin acts as an active reset output. The functionality of the reset blind time is sketched below:

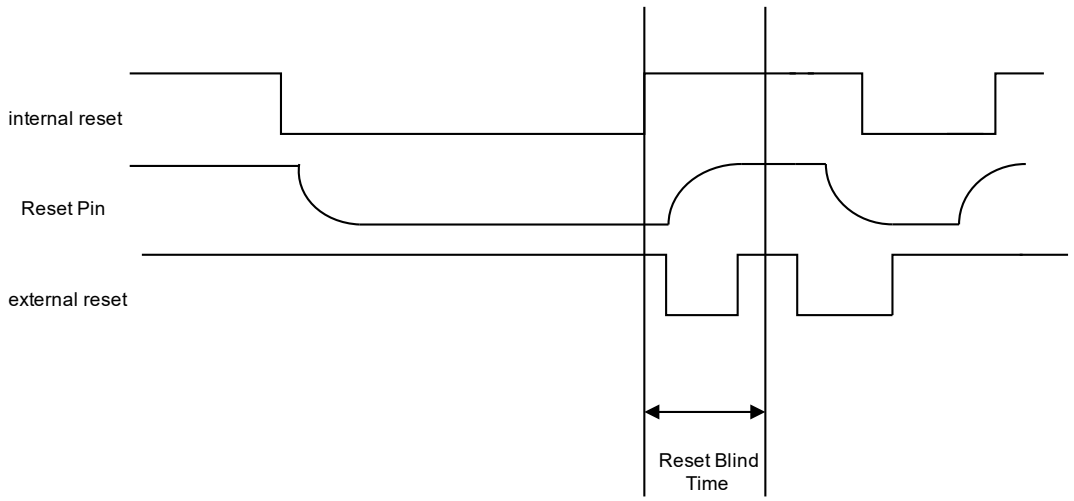
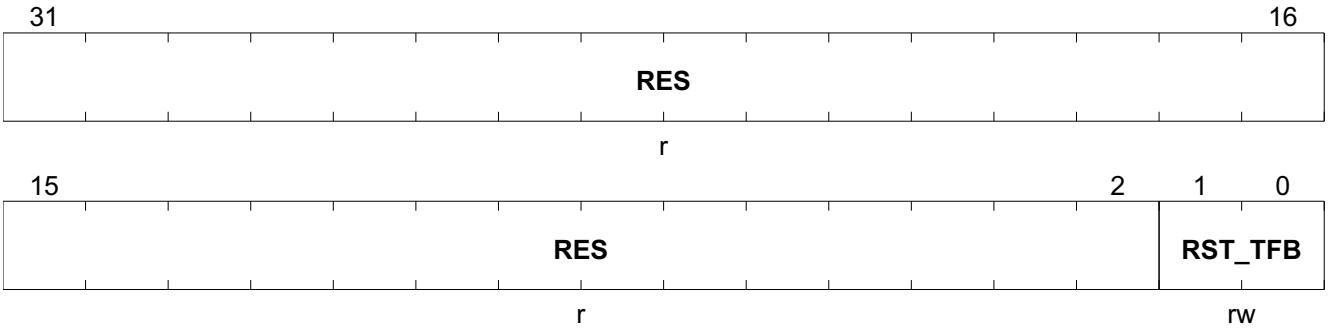


Figure 22 Reset blind time

Reset Blind Time Register

PMU_CNF_RST_TFB	Offset	Reset Value
Reset Blind Time Register	$06C_H$	see Table 24



Field	Bits	Type	Description
RES	31:2	r	**Reserved** Always read as 0
RST_TFB	1:0	rw	**Reset Pin Blind Time Selection Bits** These bits select the blind time for the reset input sampling. 00_B **RST_TFB_0**, 10 µs typ. 01_B **RST_TFB_1**, 30 µs typ. 10_B **RST_TFB_2**, 180 µs typ. 11_B **RST_TFB_3**, 1220 µs typ.

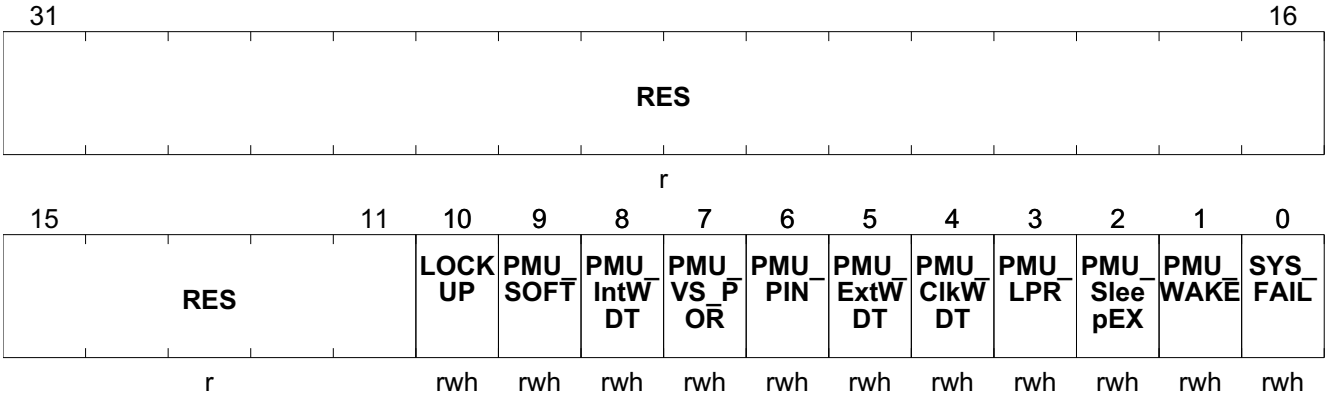
Table 24 RESET of PMU_CNF_RST_TFB

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_1	00000001_H	RESET_TYPE_1		

Reset Status Register

The PMU_RESET_STS register shows every executed reset request. The PMU writes the corresponding register bit of an executed reset. To clear the information of the PMU_RST_STS register the user must overwrite the corresponding bit with a logic zero. The register is reset by RESET_TYPE_1.

PMU_RESET_STS	Offset	Reset Value
Reset Status Register	010_H	see Table 25



Field	Bits	Type	Description
RES	31:11	r	**Reserved** Always read as 0
LOCKUP	10	rwh	**Lockup-Reset Flag** 0_B **No Reset**, No Lockup-Reset executed 1_B **Reset**, Lockup-Reset executed
PMU_SOFT	9	rwh	**Soft-Reset Flag** 0_B **No Reset**, No Soft-Reset executed 1_B **Reset**, Soft-Reset executed
PMU_IntWDT	8	rwh	**Internal Watchdog Reset Flag** 0_B **No Reset**, No Internal Watchdog reset executed 1_B **Reset**, Internal Watchdog reset executed
PMU_VS_POR	7	rwh	**Power-On Reset Flag** *Note: This flag is also set if the PMU restarts from fail safe sleep mode caused by WDT1_SEQ_FAIL or VDDP overload or VDDC overload* 0_B **No Reset**, No Power-On reset executed 1_B **Reset**, Power-On reset executed
PMU_PIN	6	rwh	**PIN-Reset Flag** 0_B **No Reset**, No PIN-Reset executed 1_B **Reset**, PIN-Reset executed

Field	Bits	Type	Description
PMU_ExtWDT	5	rwh	**External Watchdog (WDT1) Reset Flag** 0_B **No Reset**, No External Watchdog reset executed 1_B **Reset**, External Watchdog reset executed
PMU_ClkWDT	4	rwh	**Clock Watchdog (CLKWDT) Reset Flag** 0_B **No Reset**, No Clock Watchdog reset executed 1_B **Reset**, Clock Watchdog reset executed
PMU_LPR	3	rwh	**Low Priority Resets (see PMU_RST_STS2)** 0_B **No Reset**, No Low Priority reset executed 1_B **Reset**, Low Priority reset executed
PMU_SleepEX	2	rwh	**Flag which indicates a reset caused by Sleep-Exit** 0_B **No Reset**, No reset caused by Sleep-Exit executed 1_B **Reset**, Reset caused by Sleep-Exit executed
PMU_WAKE	1	rwh	**Flag which indicates a reset caused by Stop-Exit** *Note: Stop-Exit with reset must be configured explicitly in the PMU_WAKE-UP_CTRL register*[1)] 0_B **No Reset**, No reset caused by Stop-Exit executed 1_B **Reset**, Reset caused by Stop-Exit executed
SYS_FAIL	0	rwh	**Flag which indicates a reset caused by a System Fail reported in the corresponding Fail Register** *Note: This flag is also cleared if the PMU_WFS register is read.* 0_B **No Reset**, No reset caused by System Fail executed 1_B **Reset**, Reset caused by System Fail executed

1) Otherwise this flag is not set. The flag is always set in case of pin reset in Stop Mode (in combination with the flag PMU_PIN).

Table 25 RESET of PMU_RESET_STS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_1	00000000_H	RESET_TYPE_1		
RESET_TYPE_0	00000000_H	RESET_TYPE_0		

6.8 PMU Data Storage Area

The PMU provides the possibility for the system to store data in registers which will retain their values, when the device is set to sleep mode. In sum there are 12 x 8 Bit available.

6.8.1 Register Definition

Table 26 Register Overview

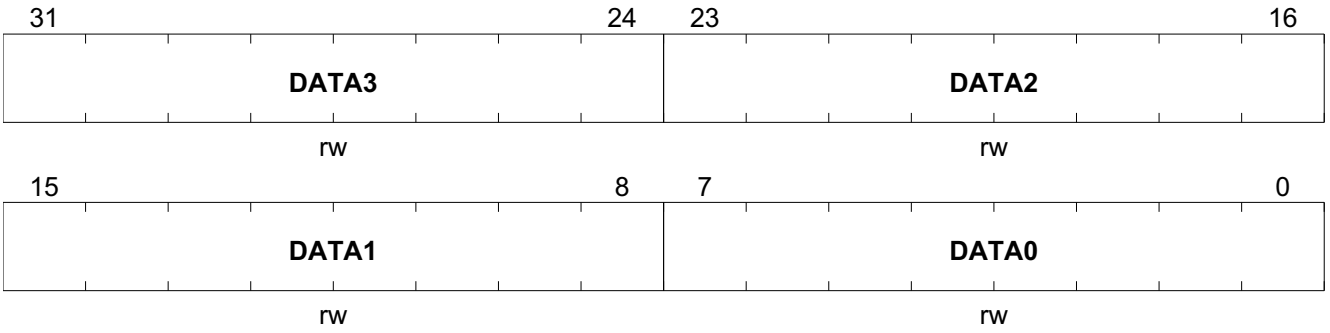
Register Short Name	Register Long Name	Offset Address	Reset Value
Register Definition, Data Storage Registers			
PMU_GPUDATA0to3	General Purpose User DATA0to3	$0C0_H$	$0000\ 0000_H$
PMU_GPUDATA4to7	General Purpose User DATA4to7	$0C4_H$	$0000\ 0000_H$
PMU_GPUDATA8to11	General Purpose User DATA8to11	$0C8_H$	$0000\ 0000_H$

The registers are addressed wordwise.

6.8.1.1 Data Storage Registers

General Purpose User DATA0to3 Storage Register

PMU_GPUDATA0to3	Offset	Reset Value
General Purpose User DATA0to3	$0C0_H$	see Table 27



Field	Bits	Type	Description
DATA3	31:24	rw	**DATA3 Storage Byte** 4th byte of storage area
DATA2	23:16	rw	**DATA2 Storage Byte** 3rd byte of storage area

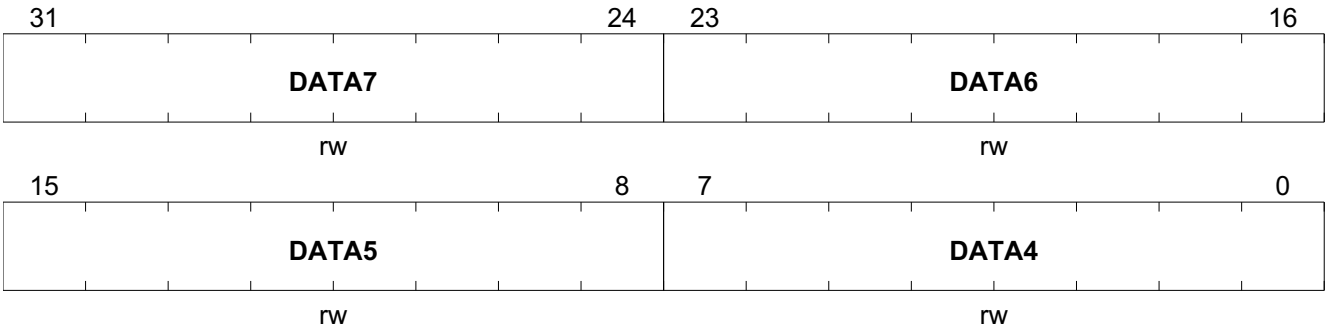
Field	Bits	Type	Description
DATA1	15:8	rw	**DATA1 Storage Byte** 2nd byte of storage area
DATA0	7:0	rw	**DATA0 Storage Byte** 1st byte of storage area

Table 27 RESET of PMU_GPUDATA0to3

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_0	00000000_H	RESET_TYPE_0		

General Purpose User DATA4to7 Storage Register

PMU_GPUDATA4to7	Offset	Reset Value
General Purpose User DATA4to7	**$0C4_H$**	**see Table 28**



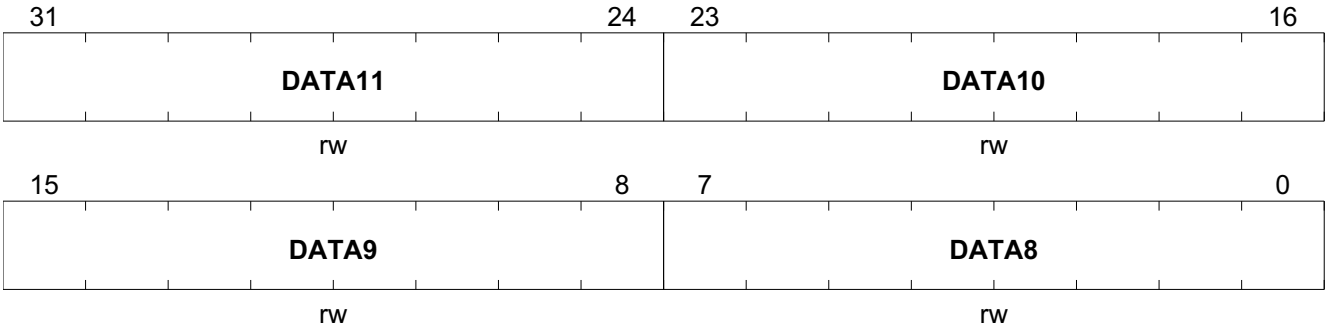
Field	Bits	Type	Description
DATA7	31:24	rw	**DATA7 Storage Byte** 8th byte of storage area
DATA6	23:16	rw	**DATA6 Storage Byte** 7th byte of storage area
DATA5	15:8	rw	**DATA5 Storage Byte** 6th byte of storage area
DATA4	7:0	rw	**DATA4 Storage Byte** 5th byte of storage area

Table 28 RESET of PMU_GPUDATA4to7

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_0	00000000_H	RESET_TYPE_0		

General Purpose User DATA8to11 Storage Register

PMU_GPUDATA8to11	Offset	Reset Value
General Purpose User DATA8to11	**$0C8_H$**	**see Table 29**



Field	Bits	Type	Description
DATA11	31:24	rw	**DATA11 Storage Byte** 12th byte of storage area
DATA10	23:16	rw	**DATA10 Storage Byte** 11th byte of storage area
DATA9	15:8	rw	**DATA9 Storage Byte** 10th byte of storage area
DATA8	7:0	rw	**DATA8 Storage Byte** 9th byte of storage area

Table 29 RESET of PMU_GPUDATA8to11

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_0	00000000_H	RESET_TYPE_0		

7 System Control Unit - Digital Modules (SCU-DM)

7.1 Features

- Flexible clock configuration features
- Reset management of all system resets
- System modes control for all power modes (active, power down, sleep)
- Interrupt enabling for many system peripherals
- General purpose input output control
- Debug mode control of system peripherals

7.2 Introduction

The System Control Unit (SCU) supports all central control tasks in the TLE985xQX. The SCU is made up of the following sub-modules:

- Clock System and Control (CGU) (see **Chapter 7.3**)
- Reset Control (RCU) (see **Chapter 7.4**)
- Power Management (PCU) (see **Chapter 7.5**)
- Interrupt Management (ICU) (see **Chapter 7.6**)
- General Port Control (see **Chapter 7.7**)
- Flexible Peripheral Management (see **Chapter 7.9**)
- Module Suspend Control (see **Chapter 7.10**)
- Watchdog Timer (WDT) (see **Chapter 7.14**)
- Error Detection and Correction in Data Memory (see **Chapter 7.14**)
- Miscellaneous Control (see **Chapter 7.15**)
- Register Mapping (see **Chapter 7.2.2**)

7.2.1 Block Diagram

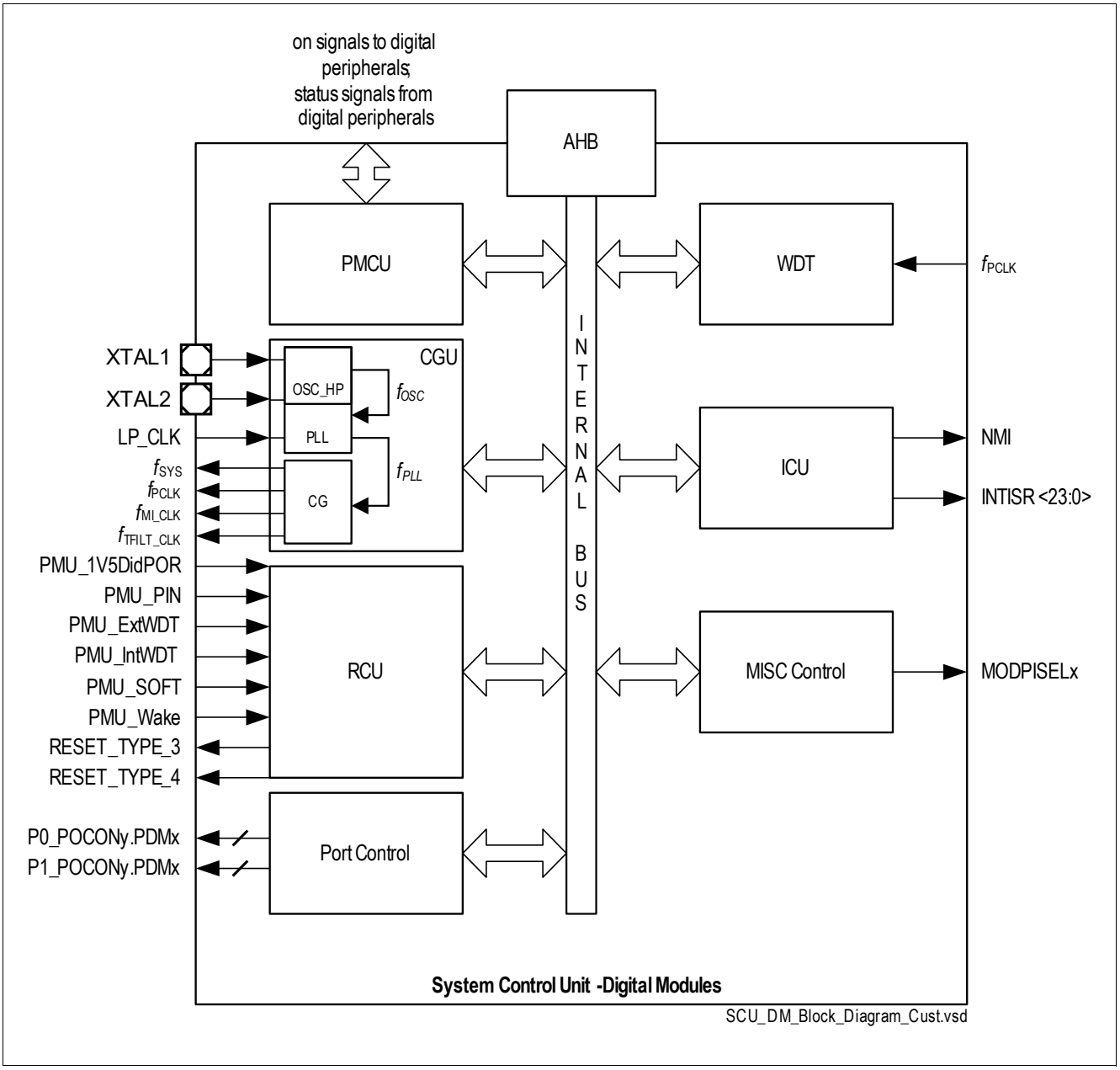


Figure 23 System Control Unit - Digital Modules Block Diagram

IO description of SCU_DM:

- CGU:
  - f_{sys}; system clock
  - LP_CLK; low-power backup clock
- RCU:
  - 1V5DidPOR; Undervoltage reset of power down supply
  - PMU_PIN; Reset generated by reset pin
  - PMU_ExtWDT; WDT1 reset
  - PMU_IntWDT; WDT (SCU) reset

  - PMU_SOFT; Software reset
  - PMU_Wake; Stop Mode exit with reset
  - Reset_Type_3; Peripheral reset (contains all resets)
  - Reset_Type_4; Peripheral reset (without SOFT and WDT reset)
- Port Control:
  - P0_POCONy.PDMx; driver strength control
  - P1_POCONy.PDMx; driver strength control
- MISC:
  - MODPISELx; Mode selection registers for UART (source selection) and Timer (trigger or count selection)
- WDT (Watchdog Timer in SCU-DM): f_{SYS}; System clock

7.2.2 SCU Register Overview

This chapter contains an overview of all SCU Registers.

7.2.2.1 Register Map

Table 31 lists the addresses of the SCU SFRs.

Table 30 shows the SCU module base address.

Table 30 Register Address Space

Module	Base Address	End Address	Note
SCU	5000 5000_H	5000 $5FFF_H$	

Table 31 Register Overview SCU Module

Register Short Name	Register Long Name	Offset Address	Reset Value
SCU_NMISRCLR	NMI Status Clear Register	000_H	see **Table 81**
SCU_IRCON0	Interrupt Request Register 0	004_H	see **Table 66**
SCU_IRCON1	Interrupt Request Register 1	008_H	see **Table 68**
SCU_IRCON2	Interrupt Request Register 2	$00C_H$	see **Table 70**
SCU_IRCON3	Interrupt Request Register 3	010_H	see **Table 72**
SCU_IRCON4	Interrupt Request Register 4	014_H	see **Table 74**
SCU_NMISR	NMI Status Register	018_H	see **Table 80**
SCU_IEN0	Interrupt Enable Register 0	$01C_H$	see **Table 60**
SCU_VTOR	Vector Table Reallocation Register	020_H	see **Table 61**
SCU_NMICON	NMI Control Register	024_H	see **Table 62**
SCU_EXICON0	External Interrupt Control Register 0	028_H	see **Table 63**
SCU_EXICON1	External Interrupt Control Register 1	$02C_H$	see **Table 64**
SCU_MODIEN1	Peripheral Interrupt Enable Register 1	030_H	see **Table 82**
SCU_MODIEN2	Peripheral Interrupt Enable Register 2	034_H	see **Table 83**
SCU_MODIEN3	Peripheral Interrupt Enable Register 3	038_H	see **Table 84**
SCU_MODIEN4	Peripheral Interrupt Enable Register 4	$03C_H$	see **Table 85**
SCU_PMCON0	Power Mode Control Register 0	040_H	see **Table 57**
SCU_PLL_CON	PLL Control Register	044_H	see **Table 42**
SCU_CMCON1	Clock Control Register 1	048_H	see **Table 43**
SCU_CMCON2	Clock Control Register 2	$04C_H$	see **Table 44**
SCU_WDTCON	Watchdog Timer Control Register	050_H	see **Table 111**
SCU_APCLK_CTRL	Analog Peripheral Clock Control Register	054_H	see **Table 46**
SCU_APCLK	Analog Peripheral Clock Register	058_H	see **Table 47**
SCU_APCLK_STS	Analog Peripheral Clock Status Register	$05C_H$	see **Table 51**
SCU_PMCON	Peripheral Management Control Register	060_H	see **Table 99**
SCU_APCLK_SCLR	Analog Peripheral Clock Status Clear Register	064_H	see **Table 52**
SCU_RSTCON	Reset Control Register	068_H	see **Table 55**

Table 31 Register Overview SCU Module (cont'd)

Register Short Name	Register Long Name	Offset Address	Reset Value
SCU_ADC1_CLK	ADC1 Peripheral Clock Register	$06C_H$	see Table 49
SCU_SYSCON0	System Control Register 0	070_H	see Table 45
SCU_SYS_STRTUP_STS	System Startup Status Register	074_H	see Table 119
SCU_WDTREL	Watchdog Timer Reload Register	078_H	see Table 110
SCU_WDTWINB	Watchdog Window-Boundary Count	$07C_H$	see Table 113
SCU_WDT	Watchdog Timer	080_H	see Table 112
SCU_BCON1, dedicated for UART1	Baud Rate Control Register 1	088_H	see Table 101
SCU_BGL1, dedicated for UART1	Baud Rate Timer/Reload Register, Low Byte 1	$08C_H$	see Table 103
SCU_BG1, dedicated for UART1	Baud Rate Timer/Reload Register	090_H	see Table 105
SCU_LINST, dedicated for UART1	LIN Status Register	094_H	see Table 107
SCU_BCON2, dedicated for UART2	Baud Rate Control Register 2	098_H	see Table 102
SCU_BGL2, dedicated for UART2	Baud Rate Timer/Reload Register, Low Byte 2	$09C_H$	see Table 104
SCU_BG2, dedicated for UART2	Baud Rate Timer/Reload Register	$0A0_H$	see Table 106
SCU_LINSCLR, dedicated for UART1	LIN Status Clear Register	$0A4_H$	see Table 108
SCU_ID	Identity Register	$0A8_H$	see Table 122
SCU_PASSWD	Password Register	$0AC_H$	see Table 117
SCU_OSC_CON	OSC Control Register	$0B0_H$	see Table 41
SCU_COCON	Clock Output Control Register	$0B4_H$	see Table 53
SCU_MODPISEL	Peripheral Input Select Register	$0B8_H$	see Table 90
SCU_MODPISEL1	Peripheral Input Select Register 1	$0BC_H$	see Table 91
SCU_MODPISEL2	Peripheral Input Select Register 2	$0C0_H$	see Table 92
SCU_MODSUSP	Module Suspend Control Register	$0C8_H$	see Table 100
SCU_GPT12PISEL	GPT12 Peripheral Input Select Register	$0D0_H$	see Table 97
SCU_EDCCON	Error Detection and Correction Control Register	$0D4_H$	see Table 114
SCU_EDCSTAT	Error Detection and Correction Status Register	$0D8_H$	see Table 115
SCU_MEMSTAT	Memory Status Register	$0DC_H$	see Table 123
SCU_NVM_PROT_STS	NVM Protection Status Register	$0E0_H$	see Table 120
SCU_MEM_ACC_STS	Memory Access Status Register	$0E4_H$	see Table 121
SCU_P0_POCON0	Port Output Control Register	$0E8_H$	see Table 94
SCU_WAKECON	Wakeup Interrupt Control Register	$0EC_H$	see Table 65
SCU_IRCON5	Interrupt Control Register 5	$0F0_H$	see Table 76

Table 31 Register Overview SCU Module (cont'd)

Register Short Name	Register Long Name	Offset Address	Reset Value
SCU_TCCR	Temperature Compensation Control Register	$0F4_H$	see Table 96
SCU_P1_POCON0	Port Output Control Register	$0F8_H$	see Table 95
SCU_MODPISEL4	Peripheral Input Select Register 4	$0FC_H$	see Table 98
SCU_XTAL_CTRL	XTAL Control Register	100_H	see Table 93
SCU_EDCSCLR	Error Detection and Correction Status Clear Register	$10C_H$	see Table 116
SCU_STACK_OVFCLR	Stack Overflow Status Clear Register	$12C_H$	see Table 127
SCU_STACK_OVF_CTRL	Stack Overflow Control Register	144_H	see Table 124
SCU_STACK_OVF_ADDR	Stack Overflow Control Register	148_H	see Table 125
SCU_STACK_OVF_STS	Stack Overflow Status	$14C_H$	see Table 126
SCU_BRDRV_CLK	BDrv Peripheral Clock Register	150_H	see Table 50
SCU_GPT12IEN	General Purpose Timer 12 Interrupt Enable Register	$15C_H$	see Table 87
SCU_GPT12IRC	Timer and Counter Control/Status Register	160_H	see Table 78
SCU_IRCON0CLR	Interrupt Request 0 Clear Register	178_H	see Table 67
SCU_IRCON1CLR	Interrupt Request 1 Clear Register	$17C_H$	see Table 69
SCU_GPT12ICLR	Timer and Counter Control/Status Clear Register	180_H	see Table 79
SCU_MONIEN	Monitoring Input Interrupt Enable Register	$18C_H$	see Table 86
SCU_IRCON2CLR	Interrupt Request 2 Clear Register	190_H	see Table 71
SCU_IRCON3CLR	Interrupt Request 3 Clear Register	194_H	see Table 73
SCU_IRCON4CLR	Interrupt Request 4 Clear Register	198_H	see Table 75
SCU_IRCON5CLR	Interrupt Request 5 Clear Register	$19C_H$	see Table 77

The registers are addressed wordwise.

7.3 Clock Generation Unit

The Clock Generation Unit (CGU) provides a flexible clock generation for TLE985xQX. During user program execution the frequency can be programmed for an optimal ratio between performance and power consumption. Therefore the power consumption can be adapted to the actual application state.

The CGU in the TLE985xQX consists of one oscillator circuit (OSC_HP), a Phase-Locked Loop (PLL) module including an internal oscillator (OSC_PLL) and a Clock Control Unit (CCU). The CGU can convert a low-frequency input/external clock signal to a high-frequency internal clock.

The system clock f_{SYS} is generated out of the following selectable clocks:

- PLL clock output f_{PLL}
- Direct clock from oscillator OSC_HP f_{OSC}
- Direct output of internal Oscillator f_{INTOSC}
- Low precision clock f_{LP_CLK} (HW-enabled for startup after reset and during power-down wake-up sequence)

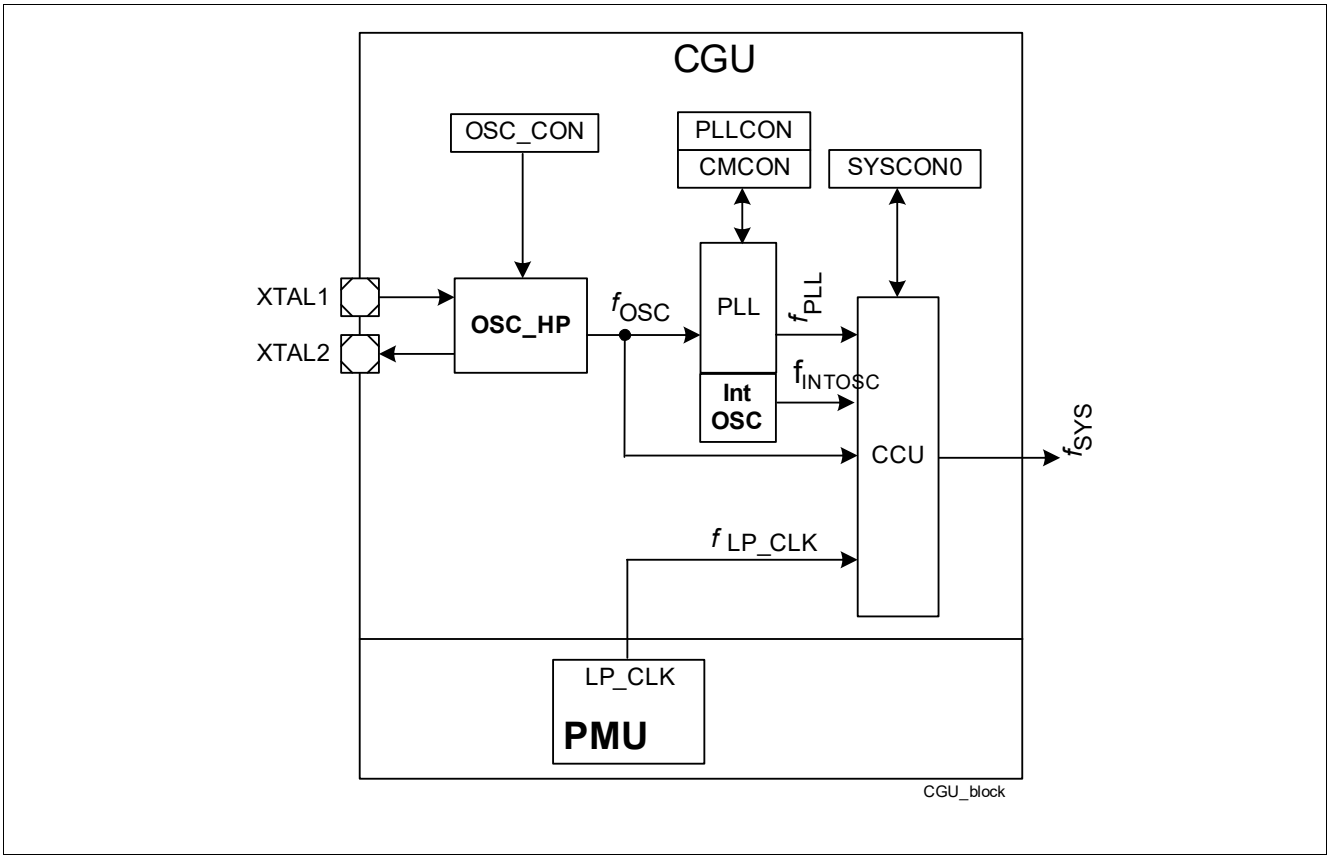


Figure 24 Clock Generation Unit Block Diagram

The following sections describe the different parts of the CGU.

7.3.1 Low Precision Clock

The clock source LP_CLK is a low-precision RC oscillator (LP-OSC, see f_{LP_CLK}) that is enabled by hardware as an independent clock source for the TLE985xQX startup after reset and during the power-down wake-up sequence. There is no user configuration possible on f_{LP_CLK}.

7.3.2 High Precision Oscillator Circuit (OSC_HP)

The high precision oscillator circuit, designed to work with both an external crystal oscillator or an external stable clock source, consists of an inverting amplifier with XTAL1 as input, and XTAL2 as output.

Figure 25 shows the recommended external circuitries for both operating modes, External Crystal Mode and External Input Clock Mode.

7.3.2.1 External Input Clock Mode

When supplying the clock signal directly, not using an external crystal and bypassing the oscillator, the input frequency needs to be equal to or greater than 4 MHz if the PLL VCO part is used.

When using an external clock signal it must be connected to XTAL1. XTAL2 is left open (unconnected).

7.3.2.2 External Crystal Mode

When using an external crystal, its frequency can be within the range of 4 MHz to 16 MHz. An external oscillator load circuitry must be used, connected to both pins, XTAL1 and XTAL2. It consists normally of the two load capacitances C1 and C2, for some crystals a series damping resistor might be necessary. The exact values and related operating range are dependent on the crystal and have to be determined and optimized together with the crystal vendor using the negative resistance method. As starting point for the evaluation, the following load cap values may be used:

Table 32 External CAP Capacitors

Fundamental Mode Crystal Frequency (approx., MHz)	Load Caps C_1, C_2 (pF)
4	33
8	18
12	12
16	10

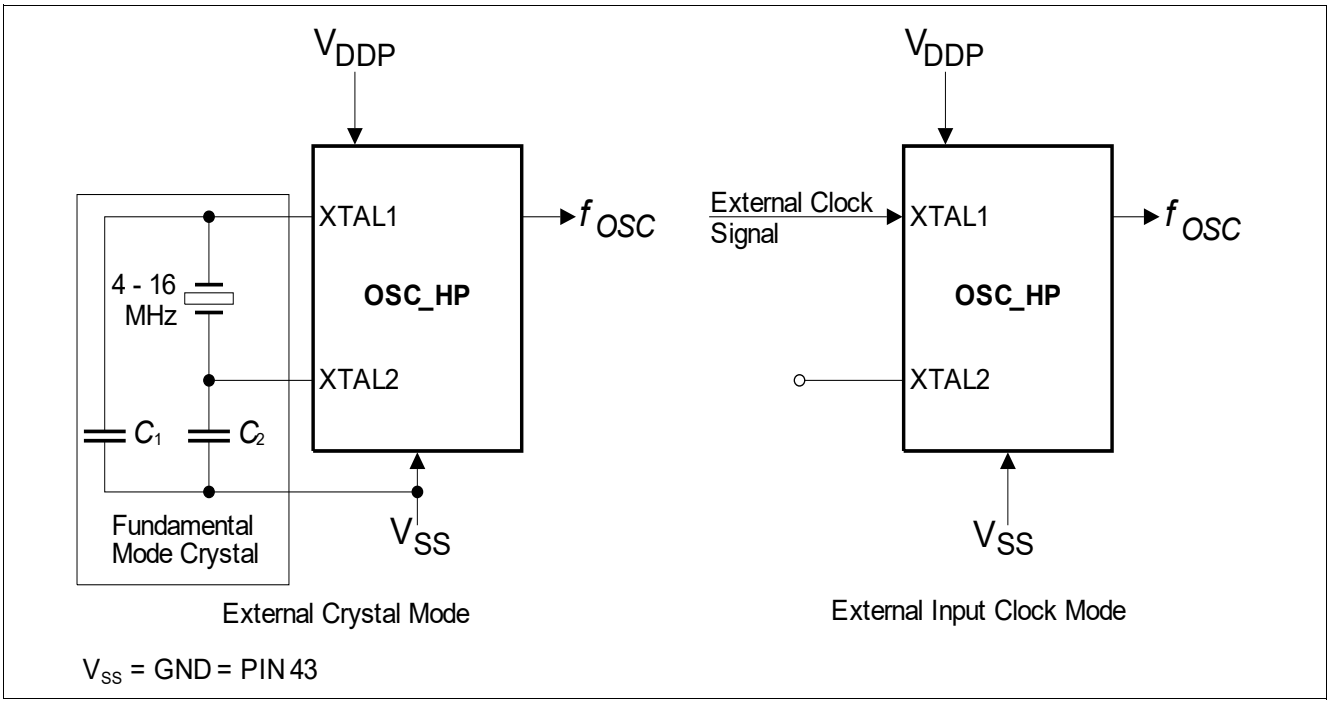


Figure 25 TLE985xQX External Circuitry for the OSC_HP

7.3.3 Phase-Locked Loop (PLL) Module

This section describes the TLE985xQX PLL module.

The clock f_{PLL} and f_{MI} is generated in one of the following PLL configured mode:

- Prescaler Mode, also called VCO Bypass Mode
- Normal Mode
- Freerunning Mode

7.3.3.1 Features

Following is an overview of the PLL features/functions:

- Programmable clock generation PLL
- Loop filter
- Wide range of input frequency (divided by user configurable 6 bit **P** divider)
- Wide VCO frequency tunning range - VCO frequency
- VCO lock detection
- Oscillator run detection
- Wide range of ouput frequency - Output frequency
- 8 bit VCO output frequency feedback divider **N**
- 3 bit VCO output frequency feedback divider **K2** and 1 bit output divider **K1**
- Oscillator Watchdog
- Prescaler Mode
- Freerunning Mode
- Normal Mode
- Sleep Mode automatically activated during device power-save mode
- Glitchless switching between both K-Dividers
- Glitchless switching between Normal Mode and Prescaler Mode
- Internal Oscillator for oscillator watchdog
- Internal Oscillator as clock source

7.3.3.2 PLL Functional Description

The following figure shows the PLL block structure.

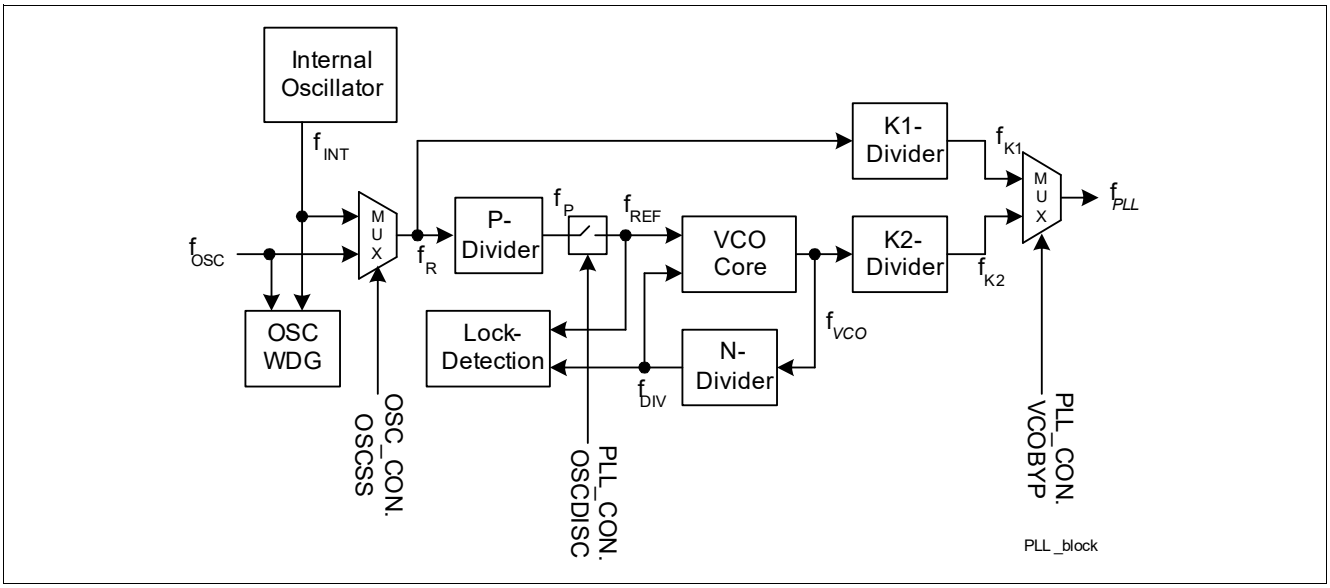


Figure 26 PLL Block Diagram

The reference frequency f_R can be selected to be taken either from the internal oscillator f_{INT} or from an external clock source f_{OSC}.

The PLL uses up to three dividers to set the system frequency in a flexible way. Each of the three dividers can be bypassed corresponding to the PLL operating mode (based on f_{PLL}):

- Bypassing P, N and K2 dividers; this defines the Prescaler Mode
- Bypassing K1 divider; this defines the Normal Mode
- Bypassing K1 divider and ignoring the P divider; this defines the Freerunning Mode

Table 33 shows the selectable clock source options.

Table 33 Clock Option Selection

VCOBYP	OSCDISC	Mode Selected
0	0	Normal Mode
1	x	Prescaler Mode
0	1	Freerunning Mode

Normal Mode

In Normal Mode the reference frequency f_R is divided down by a factor P, multiplied by a factor N and then divided down by a factor K2.

The output frequency is given by:

$$f_{PLL} = \frac{N}{P \bullet K2} \bullet f_R \quad (7.1)$$

The Normal Mode is selected by the following settings

- PLL_CON.VCOBYP = 0

The Normal Mode is active when

- PLL_CON.VCOBYP = 0
- PLL_CON.OSCDISC = 0

- PLL_CON.LOCK = 1

If f_{PLL} is selected as the clock source for system frequency f_{SYS}, the user should enable PLL in normal mode as default.

Prescaler Mode (VCO Bypass Mode)

In Prescaler Mode the reference frequency f_R is only divided down by a factor K1.

The output frequency is given by

$$f_{PLL} = \frac{f_R}{K1} \tag{7.2}$$

The Prescaler Mode is selected by the following settings

- PLL_CON.VCOBYP = 1
- PLL_CON.OSCDISC = X

The Prescaler Mode is active when

- PLL_CON.VCOBYP = 1
- PLL_CON.OSCDISC = X
- OSC_CON.OSC2L = 0 if f_{OSC} is provided as $f_{R\ (OSC\_CON.OSCSS = 01B)}$

Freerunning Mode

In Freerunning Mode the base frequency output of the Voltage Controlled Oscillator (VCO) $f_{VCObase}$ is only divided down by a factor K2.

The output frequency is given by

$$f_{PLL} = \frac{f_{VCObase}}{K2} \tag{7.3}$$

The Freerunning Mode is enabled by the following settings/conditions

- PLL_CON.VCOBYP = 0 and PLL_CON.LOCK = 0

or

- PLL_CON.VCOBYP = 1 and OSC_CON.OSCSS = 1 and OSC_CON.OSC2L = 1

or

- PLL_CON.VCOBYP = 0
- PLL_CON.OSCDISC = 1
- PLL_CON.LOCK = 0

General Configuration Overview

The divider values and all necessary other values can be configured via the PLL configuration registers.

In TLE985xQX, the P factor can be programed in the range from 4 to 50. **Table 34** shows a few possible values for the P factor and gives the valid output frequency range for the P divider dependent on P and the f_R frequency range:

Table 34 P-Divider Factor

P	f_P for f_R =				
	4 MHz	**5 MHz**	**6 MHz**	**8 MHz**	**10 MHz**
4	1	1.25	not allowed	not allowed	not allowed
5	0.8	1	1.2	not allowed	not allowed
6	not allowed	0.833	1	not allowed	not allowed
8	not allowed	not allowed	not allowed	1	1.25
10	not allowed	not allowed	not allowed	0.8	1

The P-divider output frequency f_P is fed to the Voltage Controlled Oscillator (VCO).The VCO is a part of PLL with a feedback path. A divider in the feedback path (N divider) divides the VCO frequency. The f_{VCO} range is defined by:

Table 35 VCO Range

min. VCO tunning range frequency	max. VCO tunning range frequency	VCO freerunning frequency	Unit
see f_{VCOmin}	see f_{VCOmax}	see $f_{VCOfree}$ [1]	MHz

1) $f_{VCOfree}$ is the free running operation frequency of the PLLVCO, when no input reference clock is available.

The following table shows some examples for possible N loop division rates and gives the valid output frequency range for f_{REF} depending on N and the VCO frequency range. All not allowed combinations are related to the fact that using them the limits of parameter f_{REF} are violated:

Table 36 N Loop Division Rates

N	f_{DIV} for f_{VCO} =					
	48	**72**	**96**	**112**	**136**	**160**
39	1.231	not allowed	not allowed	not allowed	not allowed	not allowed
44	1.091	not allowed	not allowed	not allowed	not allowed	not allowed
48	1.0	not allowed	not allowed	not allowed	not allowed	not allowed
56	0.857	not allowed	not allowed	not allowed	not allowed	not allowed
60	0.800 (marginal)	1.2	not allowed	not allowed	not allowed	not allowed
80	not allowed	0.9	1.2	not allowed	not allowed	not allowed
100	not allowed	not allowed	0.96	1.12	not allowed	not allowed
120	not allowed	not allowed	0.800 (marginal)	0.933	1.133	not allowed
140	not allowed	not allowed	not allowed	0.800 (marginal)	0.971	1.143

Table 36 N Loop Division Rates (cont'd)

N	f_{DIV} for f_{VCO} =					
	48	**72**	**96**	**112**	**136**	**160**
160	not allowed	not allowed	not allowed	not allowed	0.850	1.0
180	not allowed	not allowed	not allowed	not allowed	not allowed	0.889
200	not allowed	not allowed	not allowed	not allowed	not allowed	0.800 (marginal)

Note:

The N-divider output frequency f_{DIV} is then compared with f_{REF} in the phase detector logic, within the VCO logic. The phase detector determines the difference between the two clock signals and accordingly controls the output frequency of the VCO, f_{VCO}.

Note: Due to this operation, the VCO clock of the PLL has a frequency which is a multiple of f_{DIV}. The factor for this is controlled through the value applied to the N-divider in the feedback path. For this reason this factor is often called a multiplier, although it actually controls division.

The output frequency of the VCO, f_{VCO}, is divided by K2 to provide the final desired output frequency f_{PLL}. **Table 37** shows the output frequency range depending on the K2 divisor and the VCO frequency range:

Table 37 K2 Divisor Table

K2	f_{PLL} for f_{VCO} =						Duty Cycle [%]
	48	**72**	**96**	**112**	**136**	**160**	
2	24.0	36.0	not allowed	not allowed	not allowed	not allowed	50
3	16.0	24.0	32.0	37.333	not allowed	not allowed	40
4	12.0	18.0	24.0	28.0	34.0	40.0	50
5	9.6	14.4	19.2	22.4	27.2	32.0	44
6	8.0	12.0	16.0	18.667	22.667	26.667	50
7	6.857	10.286	13.714	16.0	19.429	22.857	45.71
8	6.0	9.0	12.0	14.0	17.0	20.0	50
9	5.333	8.0	10.667	12.444	15.111	17.778	46.67

Notes

1. *The whole range in between two f_{vco} columns in the above table is allowed.*
2. *For divider factors that cause duty cycles far off of 50%, not only the cycle time has to be checked, but also the minimum clock pulse width.*

For the K1-divider the same table is valid as for the K2-divider. The only difference is that not f_{VCO} is used as reference, f_R is used instead.

Table 38 K1 Divisor Table

K1	f_{PLL} for f_R =			Duty Cycle [%]
	5	8	16	
1	5.0	8.0	16.0	40 - 60
2	2.5	4.0	8.0	50
others	not accessible			

For different source oscillator, some examples for f_{PLL} are shown in **Table 39**.

Table 39 System Frequency

f_{PLL} Selected	Oscillator	f_{OSC}	N	P	K	Actual f_{SYS}
40 MHz	On-chip	5 MHz	80	5	2	40 MHz
	External	4 MHz	80	4	2	40 MHz
		6 MHz	80	6	2	40 MHz
		8 MHz	80	8	2	40 MHz
37.5 MHz	On-chip	5 MHz	75	5	2	37.5 MHz
	External	4 MHz	75	4	2	37.5 MHz
		6 MHz	75	6	2	37.5 MHz
		8 MHz	75	8	2	37.5 MHz
36 MHz	On-chip	5 MHz	72	5	2	36 MHz
	External	4 MHz	72	4	2	36 MHz
		6 MHz	72	6	2	36 MHz
		8 MHz	72	8	2	36 MHz
25 MHz	On-chip	5 MHz	50	5	2	25 MHz
	External	4 MHz	50	4	2	25 MHz
		6 MHz	50	6	2	25 MHz
		8 MHz	50	8	2	25 MHz
20 MHz	On-chip	5 MHz	60	5	3	20 MHz
	External	4 MHz	60	4	3	20 MHz
		6 MHz	60	6	3	20 MHz
		8 MHz	60	8	3	20 MHz
16 MHz	On-chip	5 MHz	64	5	4	16 MHz
	External	4 MHz	64	4	4	16 MHz
		6 MHz	64	6	4	16 MHz
		8 MHz	64	8	4	16 MHz

7.3.3.3 Oscillator Watchdog

The oscillator watchdog monitors the external incoming clock f_{OSC}. Only incoming frequencies that are too low (below 300 kHz) to enable a stable operation of the VCO circuit are detected.

As reference clock the internal oscillator (OSC_PLL) frequency f_{INT} is used and therefore the internal oscillator must be put into operation.

By setting bit OSC_CON.OSCWDTRST the oscillator watchdog can be restarted without a reset of the complete PLL. The detection status output is only valid after some cycles of f_{INT}.

7.3.3.4 PLL VCO Lock Detection

The PLL has a lock detection that supervises the VCO part of the PLL in order to differentiate between stable and instable VCO circuit behavior. The lock detector marks the VCO circuit and therefore the output f_{VCO} of the VCO as instable if the two inputs f_{REF} and f_{DIV} differ too much. Changes in one or both input frequencies below a level are not marked by a loss of lock because the VCO can handle such small changes without any problem for the system. **Table 40** shows values below that the lock is not lost for different input values.

Table 40 Loss of VCO Lock Definition

Maximum Allow Changing		
df_{DIV}/dt for f_{REF} =		(7.4)
0.8 MHz	**1 MHz**	**1.25 MHz**
≤ 0.54 kHz/µs	≤ 0.96 kHz/µs	≤ 1.49 kHz/µs

7.3.3.5 Internal Oscillator (OSC_PLL)

The PLL internal oscillator is used for two different purposes:

Providing a Input Clock to the PLL

OSC_PLL operates at nominal frequency of 5 MHz.

The OSC_PLL can be used as input clock for all PLL modes. This is controlled and configured via OSC_CON.OSCSS.

Operating the Oscillator Watchdog

The input frequency for the PLL direct from OSC_HP (XTAL), is supervised using the OSC_PLL as reference frequency. For more information see **Section 7.3.3.3**.

7.3.3.6 Switching PLL Parameters

The following restriction applies when changing PLL parameters via the PLL_CON register:

- Prescaler Mode (VCO bypass) may be enabled at any time, however, it has to be ensured that the maximum operating frequency of the device will not be exceeded.
- Before switching NDIV, the Prescaler Mode has to be selected.
- K1DIV as well as K2DIV may be switched at any time, however, it has to be ensured that the maximum operating frequency of the device will not be exceeded.
- Only one parameter should be switched at one register write operation.
- Before switching the input clock source via OSC_CON.OSCSS, the Prescaler Mode has to be selected. Due to a following potential oscillator watchdog event, the PLL may switch to Freerunning Mode. The procedure to set up the PLL in normal operation follows that as stated in **Section 7.3.3.8**.
- Before deselecting the Prescaler Mode, the RESLD bit has to be set and then the LOCK flag has to be checked. Only when the LOCK flag is set again, the Prescaler Mode may be deselected.

Note: PDIV and NDIV can also be switched in Normal Mode. When changing NDIV, it must be regarded that the VCO clock fVCO may exceed the target frequency until the PLL becomes locked. After changing PDIV or NDIV, it must be waited for the PLL lock condition. This procedure is typically used for increasing the VCO clock step-by-step.

7.3.3.7 Oscillator Watchdog Event or PLL Loss of Lock Detection

In case of detection of too low frequency of the external clock source f_{OSC}, the OSC-Too-Low flag (OSC_CON.OSC2L) is set. If enabled by NMICON.NMIOWD, a trap request to the CPU is activated correspondingly only in these two cases: 1) When PLL is in Prescaler Mode and OSCSS = 01 selecting f_{OSC} as PLL input clock source and SYSCON0.SYSCLKSEL selects PLL clock output as the system frequency, or 2) When SYSCON0.SYSCLKSEL selects f_{OSC} as the system frequency. With these 2 cases and the OSC2L condition, the OWD NMI flag FNMIOWD in NMISR is set.

Note: Do not restart the oscillator watchdog detection by setting bit OSC_CON.OSCWDTRST while PLL is in Prescaler Mode, as the detection status (OSC_CON.OSC2L) takes some time to be stable.

A oscillator watchdog event normally leads to a following PLL loss-of-lock detection.

If PLL is not the system clock source (SYSCON0.SYSCLKSEL deselects PLL or PLL is in Prescaler Mode) when the loss-of-lock is detected, only the lock flag is reset (PLL_CON.LOCK = 0). No loss-of-lock NMI is generated and no further action is taken. Otherwise if PLL is selected as clock source for system frequency and VCOBYP = 0, the PLL loss-of-lock NMI flag FNMIPLL in NMISR is set. If enabled by NMICON.NMIPLL, an NMI trap request to the CPU is activated. In addition, the lock flag is reset. Note that in the first place, the LOCK flag has to be set first before a loss-of-lock NMI request is generated. This avoids a potential PLL loss-of-lock NMI request after device power-on reset.

On an oscillator watchdog event when PLL is in Prescaler Mode and external clock (OSC_HP) is selected as PLL clock input; or on PLL loss-of-lock detection when PLL is in Normal Mode, the PLL will be switched to run in the Freerunning Mode on the VCO base frequency divided by K2, which is enforced by hardware until the Prescaler Mode is (re-)selected.

Due to the above, the PLL shall only run in Prescaler Mode when changing the PLL configuration or switching between PLL operation modes.

7.3.3.8 Oscillator Watchdog Event or Loss of Lock Recovery

In case of oscillator watchdog NMI, user software can first check if the PLL remains locked. If not, the clock system can be reconfigured again by executing the following sequence as the OWD NMI routine:

1. Restart the oscillator watchdog detection by setting bit OSC_CON.OSCWDTRST
2. Wait until OSC_CON.OSC2L is clear
3. When bit OSC_CON.OSC2L is cleared, then
   a) The Prescaler Mode has to be selected (PLL_CON.VCOBYP = 1)
   b) Setting the restart lock detection bit PLL_CON.RESLD = 1
   c) Waiting until the PLL VCO part becomes locked (PLL_CON.LOCK = 1)
   d) When the LOCK is set again, the Prescaler Mode can be deselected (PLL_CON.VCOBYP = 0) and normal PLL operation is resumed.
4. Clear the OWD NMI flag FNMIOWD.

In the general case of PLL loss-of-lock or to re-configure the PLL settings, user software can try to configure the clock system again by executing the following sequence:

1. If input clock source is from XTAL (f_{OSC} from OSC_HP), ensure the input frequency is above threshold by checking OSC_CON.OSC2L.
2. The Prescaler Mode has to be selected (PLL_CON.VCOBYP = 1)
3. If desired, (re-)configure the PLL divider settings.
4. Setting the restart lock detection bit PLL_CON.RESLD = 1
5. Waiting until the PLL VCO part becomes locked (PLL_CON.LOCK = 1)
6. When the LOCK is set again, the Prescaler Mode can be deselected (PLL_CON.VCOBYP = 0) and normal PLL operation is resumed.
7. Clear the PLL loss-of-lock NMI flag FNMIPLL.

7.3.4 Clock Control Unit

The Clock Control Unit (CCU) receives the clock from the PLL f_{PLL}, the external input clock f_{OSC}, the internal input clock f_{INTOSC}, or the low-precision input clock f_{LP_CLK}. The system frequency is derived from one of these clock sources.

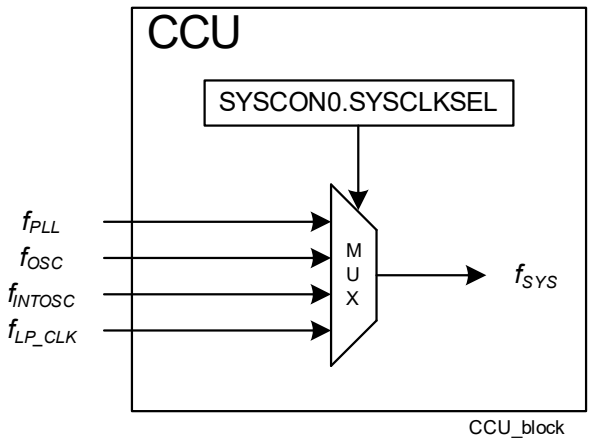


Figure 27 Clock Inputs to Clock Control Unit

The CCU generates all necessary clock signals within the microcontroller from the system clock. It consists of:

- Clock slow down circuitry
- Centralized enable/disable circuit for clock control

In normal running mode, the main module frequencies (synchronous unless otherwise stated) are as follows:

- System frequency, f_{SYS} = up to 40 MHz (measurement interface clock MI_CLK is derived from this clock)
- CPU clock (CCLK, SCLK) = up to 40 MHz (divide-down of NVM access clock)
- NVM access clock (NVMACCCLK) = up to 40 MHz
- Peripheral clock (PCLK, PCLK2, NVMCLK) = up to 40 MHz (equals CPU clock; must be same or higher)

Some peripherals are clocked by PCLK, others clocked by PCLK2 and the NVM is clocked by both NVMCLK and NVMACCCLK. During normal running mode, PCLK = PCLK2 = NVMCLK = CCLK. On wake-up from power-down mode, PCLK2 is restored similarly like NVMCLK, whereas PCLK is restored only after PLL is locked.

For optimized NVM access (read/write) with reduced wait state(s) and with respect to system requirements on CPU operational frequency, bit field NVMCLKFAC is provided for setting the frequency factor between the NVM access clock NVMACCCLK and the CPU clock CCLK.

For the slow down mode, the operating frequency is reduced using the slow down circuitry with clock divider setting at the bit field CLKREL. Bit field CLKREL is only effective when slow down mode is enabled via SFR bit PMCON0.SD bit. Note that the slow down setting of bit field CLKREL correspondingly reduces the NVMACCCLK clock. Slow down setting does not influence the erase and write cycles for the NVM.

Peripherals UART1, UART2, T2 and T21 are not influenced by CLKREL and either not by NVMCLKFAC, to allow functional LIN communication in slow down mode.

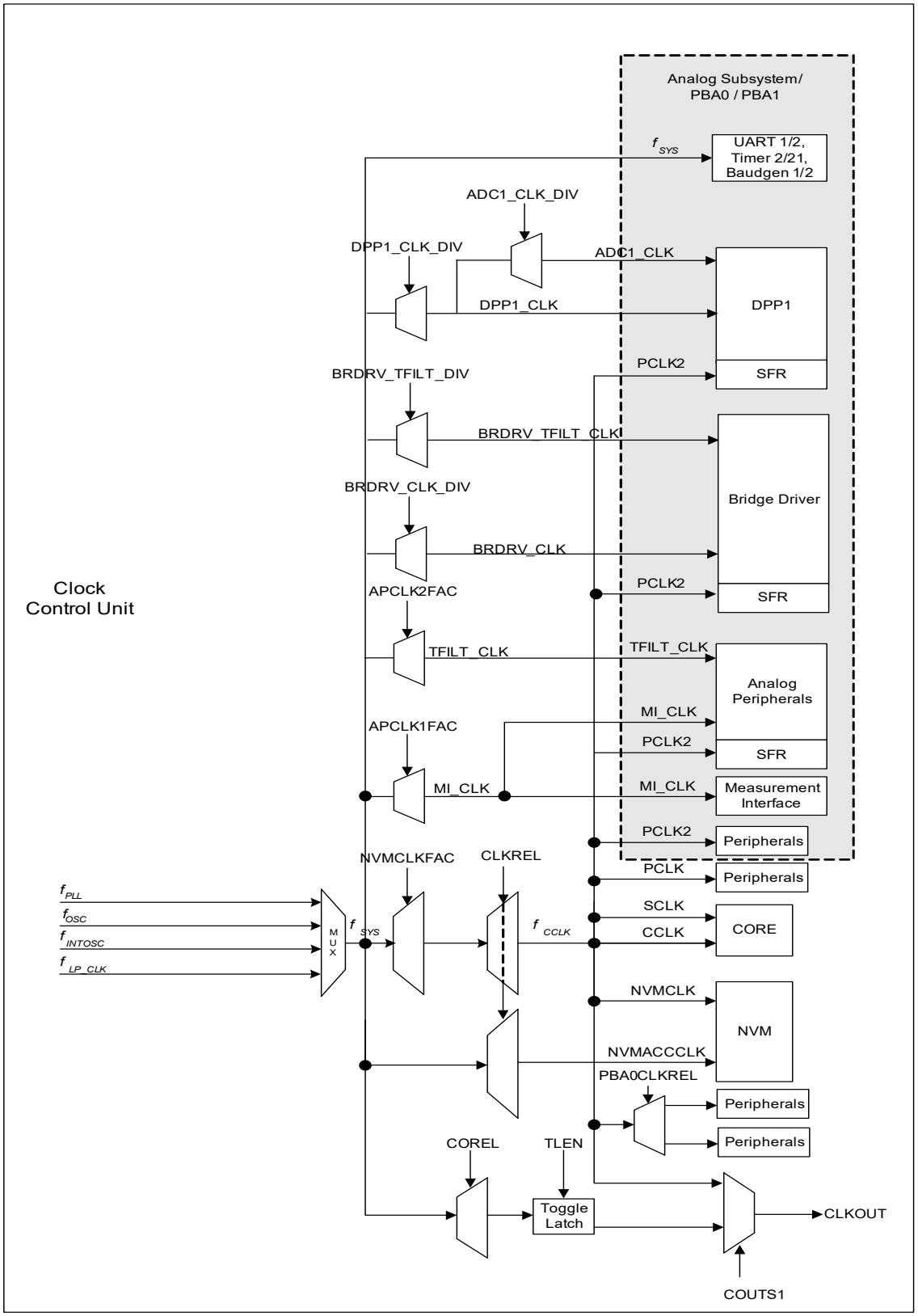


Figure 28 Clock Generation from f_{sys}; CLKOUT Generation

7.3.4.1 Clock Tree

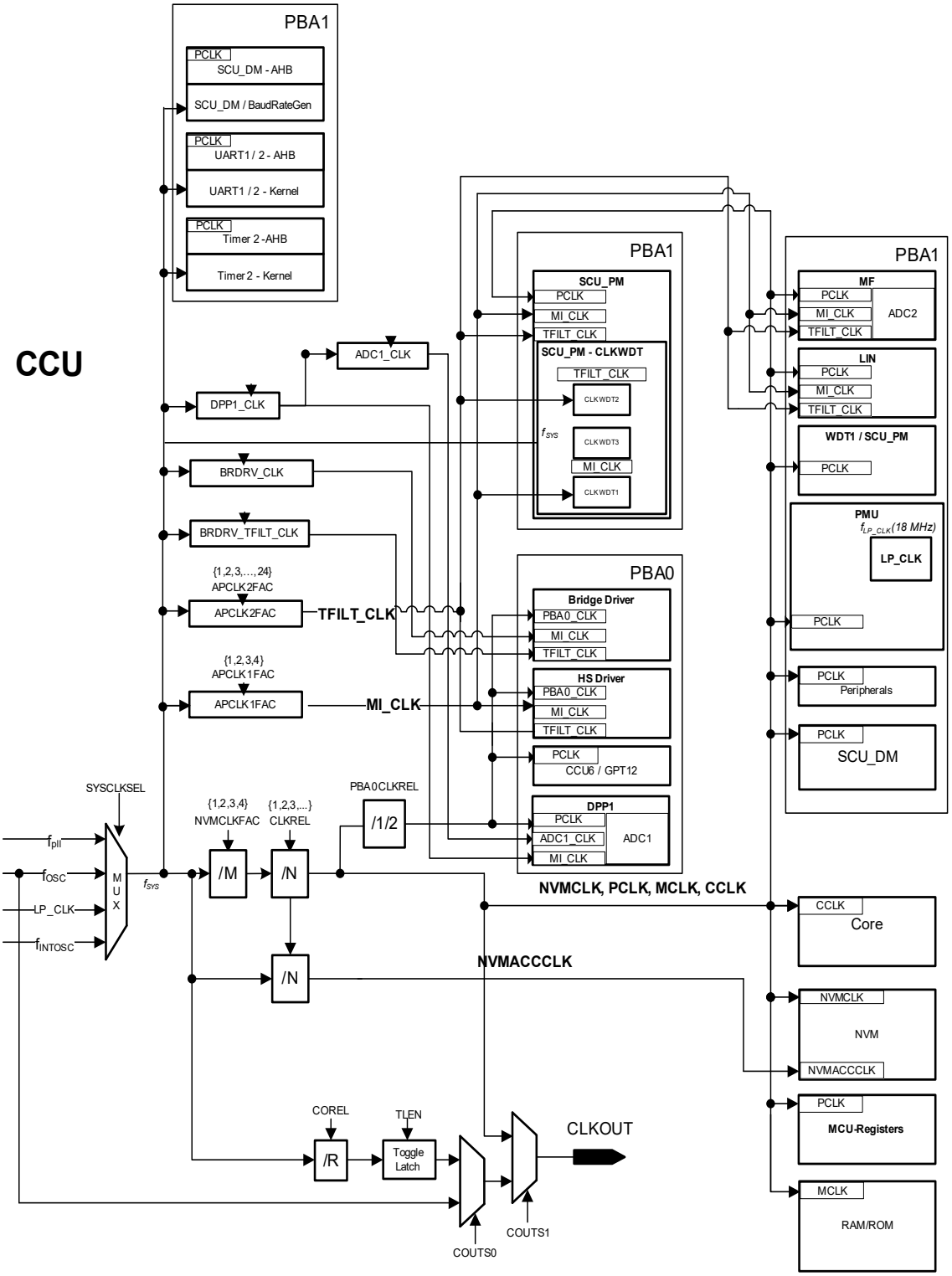


Figure 29 Clock Tree

7.3.4.2 Startup Control for System Clock

Typically when the TLE985xQX starts up after reset, the LP_CLK is selected by hardware to provide the system frequency f_{SYS}. CPU runs based on this system frequency during startup operation by boot firmware (unless otherwise specified and configured by firmware). Meanwhile, the system clock input is switched to the PLL output. With user boot configuration, the PLL is configured with internal oscillator (5 MHz) as input, by default. User code can modify the default PLL configuration as required.

The exception to the above is with resets that do not reset the clock system, which are watchdog timer (WDT) reset and soft reset. With these resets, the previous user configuration of PLL and clock system is retained across the reset.

Note: In the event the PLL fails to lock during startup operation, the LP_CLK continues to provide the system clock input. The system clock input source is indicated by the register bit field SYSCON0.SYSCLKSEL.

7.3.5 External Clock Output

An external clock output is provided as CLKOUT. This output clock can be enabled/disabled via bit COCON.EN. One of three clock sources (f_{CCLK} or f_{SYS}/n or f_{OSC}) can be selected for output, configured via bit fields COCON.COUTS1 and COUTS0.

If COUTS1 = 0 (independent on COUTS0), the output clock is f_{CCLK}. Otherwise, if COUTS0 = 0, the output clock is from oscillator output frequency; if COUTS0 = 1, the clock output frequency is chosen by the bit field COREL which selects the n divider factor on f_{SYS}. Under this selection, the clock output frequency can further be divided by 2 using a toggle latch (TLEN = 1), the resulting output frequency has 50% duty cycle.

7.3.6 CGU Registers

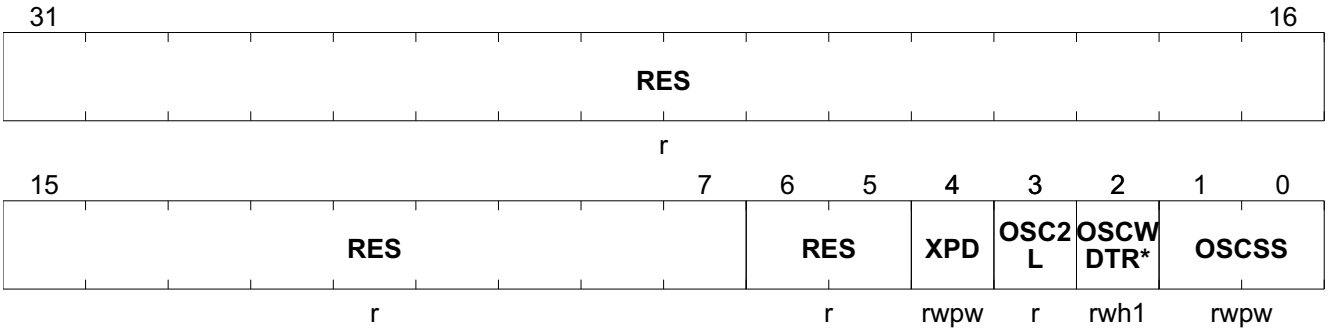
The registers of the clock generation unit for PLL and oscillator control are not affected by the watchdog timer (WDT) reset and soft reset. Therefore the system clock configuration and frequency is maintained across these types of reset.
Unless otherwise stated, the reset value as stated for the following registers apply only with Power-On reset, Brown-Out reset, Hard reset, WDT1 reset or Wake-up reset.

7.3.6.1 PLL Oscillator Register

These registers control the setting and trimming of OSC_PLL, the Power Down of XTAL (OSC_HP) and the control and status monitor of oscillator watchdog.

OSC Control Register

SCU_OSC_CON	Offset	Reset Value
OSC Control Register	$0B0_H$	see Table 41



Field	Bits	Type	Description
RES	31:7	r	**Reserved** This bit field is always read as zero.
RES	6:5	r	**Reserved** This bit field is always read as zero.
XPD	4	rwpw	**XTAL (OSC_HP) Power Down Control** The XPD bit is a protected bit. When the Protection Scheme is activated, this bit cannot be written directly. For more information on Protection Scheme, see Section 7.15. *Note: When XPD is set, switch of clock source to internal oscillator has to be done asynchronous.* 0_B **Enabled** , XTAL (OSC_HP) is not powered down. 1_B **Power down**, XTAL (OSC_HP) is powered down.

Field	Bits	Type	Description
OSC2L	3	r	**OSC-Too-Low Condition Flag** The Oscillator Watchdog monitors the $f_{OSC.}$ On OSC-too-low detection (OSC2L: 0 →1) and VCOBYP = 1 and OSCSS = 01, PLL switches to freerunning mode. On above condition, and when f_{OSC} is selected as the system clock source, hardware switches the system clock source to PLL (**SCU_SYSCON0**.SYSCLKSEL is also updated). *Note: OWD NMI request is activated on OSC-too-low condition only in two cases: 1) when VCOBYP = 1 and OSCSS = 01 and SYSCLKSEL selects PLL clock as system clock source; 2) when SYSCLKSEL selects f_{OSC} as system clock source.* 0_B **OSC OK**, f_{OSC} is above threshold. 1_B **OSC too low**, f_{OSC} is below threshold.
OSCWDTRST	2	rwh1	**Oscillator Watchdog Reset** Setting this bit will reset the OSC2L status flag to 1 and restart the oscillator detection. This bit will be automatically reset to 0 and thus always be read back as 0. 0_B **No Reset** , No effect. 1_B **Reset** , Reset OSC2L flag and restart the oscillator watchdog of the PLL.
OSCSS	1:0	rwpw	**Oscillator Source Select** The OSCSS bit is a protected bit. When the Protection Scheme is activated, this bit cannot be written directly. For more information on Protection Scheme, see **Section 7.15**. **Notes** 1. *Synchronous switching of clock source to internal oscillator is not possible when XPD = 1 or no external clock is available (check bit OSC2L).* 2. *Use the 1X option only when the external clock is not available.* 00_B **f_int_sync**, PLL internal oscillator OSC_PLL (f_{INT}) is selected synchronously as f_R. 01_B **Xtal_sync**, XTAL (f_{OSC} from OSC_HP) is selected synchronously as f_R. 10_B **f_int_async**, PLL internal oscillator OSC_PLL (f_{INT}) is selected asynchronously as f_R. 11_B **f_int_async**, PLL internal oscillator OSC_PLL (f_{INT}) is selected asynchronously as f_R.

Table 41 RESET of SCU_OSC_CON

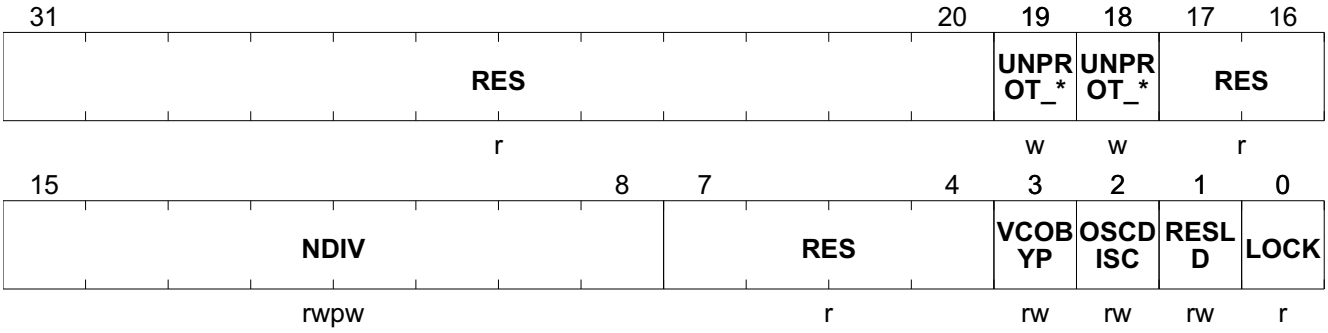
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000010_H	RESET_TYPE_4		

7.3.6.2 PLL Registers

These registers control the PLL configuration or settings.

PLL Control Register

SCU_PLL_CON	Offset	Reset Value
PLL Control Register	044_H	see Table 42



Field	Bits	Type	Description
RES	31:20	r	**Reserved** Returns 0 if read; should be written with 0.
UNPROT_VCOBYP	19	w	**Unprotect write access of VCO_BYP** Writing this Bit within an write access of VCO_BYP will overtake the provided value to VCO_BYP without protection Note: Read is always ‘0’
UNPROT_OSCDISC	18	w	**Unprotect write access of OSC_DISC** Writing this Bit within an write access of OSCDISC will overtake the provided value to OSCDISC without protection Note: Read is always ‘0’
RES	17:16	r	**Reserved** Returns 0 if read; should be written with 0.

Field	Bits	Type	Description
NDIV	15:8	rwpw	**PLL N-Divider** Values from **39 to 200** are used, smaller or higher values are saturated. The NDIV bit is a protected bit. When the Protection Scheme is activated, this bit cannot be written directly. For more information on Protection Scheme, see **Section 7.15**. 00_D **0**, N = 39, ... 38_D **38**, N = 39 39_D **39**, N = 39, ... 80_D **80**, N = 80 (default), ... 200_D **200**, N = 200 201_D **201**, N = 200, ... 255_D **255**, N = 200
RES	7:4	r	**Reserved** Returns 0 if read; should be written with 0.
VCOBYP	3	rw	**PLL VCO Bypass Mode Select** This bit is cleared by hardware when PLL switches to freerunning mode. The VCOBYP bit is a protected bit. When the Protection Scheme is activated, this bit cannot be written directly. For more information on Protection Scheme, see **Section 7.15**. When the bit value changes from 0 to 1, bit OSCDISC = 0. 0_B **Normal**, Normal (or freerunning) operation (default) 1_B **Prescaler**, Prescaler Mode; VCO is bypassed (PLL output clock is derived from input clock divided by K1-divider)
OSCDISC	2	rw	**Oscillator Disconnect** By default after power-on reset, PLL is running in Freerunning Mode (osc is disconnected). The OSCDISC bit is a protected bit. When the Protection Scheme is activated, this bit cannot be written directly. For more information on Protection Scheme, see **Section 7.15**. 0_B **Connect**, Oscillator is connected to the PLL 1_B **Disconnect**, Oscillator is disconnected to the PLL.
RESLD	1	rw	**Restart Lock Detection** Setting this bit will reset the PLL lock status flag and restart the lock detection. This bit will be automatically reset to 0 and thus always be read back as 0. The RESLD bit is a protected bit. When the Protection Scheme is activated, this bit cannot be written directly. For more information on Protection Scheme, see **Section 7.15**. 0_B **No reset**, No effect. 1_B **reset**, Reset lock flag and restart lock detection.

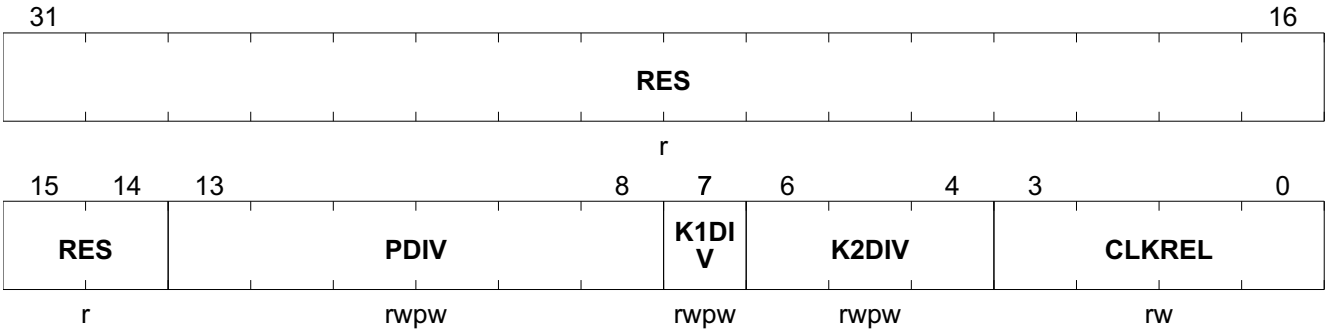
Field	Bits	Type	Description
LOCK	0	r	**PLL Lock Status Flag** **Notes** 1. *In case of a loss of VCO lock the f_{VCO} goes to the upper boundary of the selected VCO band if the reference clock input is greater as expected.* 2. *In case of a loss of VCO lock the f_{VCO} goes to the lower boundary of the selected VCO band if the reference clock input is lower as expected.* 3. *On loss-of-lock detection (LOCK: 1 →0) and when VCOBYP = 0, PLL switches to freerunning mode.* 4. *Loss-of-lock NMI request is activated only on loss-of-lock detection when VCOBYP = 0 and SYSCON0.SYSCLKSEL selects PLL clock as system frequency.* 0_B **Not Locked**, The frequency difference of f_{REF} and f_{DIV} is greater than allowed. The VCO part of the PLL can not lock on a target frequency. 1_B **Locked**, The frequency difference of f_{REF} and f_{DIV} is small enough to enable a stable VCO operation.

Table 42 RESET of SCU_PLL_CON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00005004_H	RESET_TYPE_4		
VARIANT	00005004_H	VARIANT		

Clock Control Register 1

SCU_CMCON1	Offset	Reset Value
Clock Control Register 1	048_H	see Table 43



Field	Bits	Type	Description
RES	31:14	r	**Reserved** Returns 0 if read; should be written with 0.
PDIV	13:8	rwpw	**PLL PDIV-Divider:** The PDIV bit is a protected bit. When the Protection Scheme is activated, this bit cannot be written directly. For more information on Protection Scheme, see **Section 7.15**. Values from **4 to 50** are used, smaller or higher values are saturated. 0_D **0**, P = 4 1_D **1**, P = 4 2_D **2**, P = 4 3_D **3**, P = 4 4_D **4**, P = 4 5_D **5**, P = 5 (default) 6_D **6**, P = 6, ... 50_D **50**, P = 50 51_D **51**, P = 50, ... 63_D **63**, P = 50
K1DIV	7	rwpw	**PLL K1-Divider** The K1DIV bit is a protected bit. When the Protection Scheme is activated, this bit cannot be written directly. For more information on Protection Scheme, see **Section 7.15**. 0_B **div 2**, K1 = 2 1_B **div 1**, K1 = 1

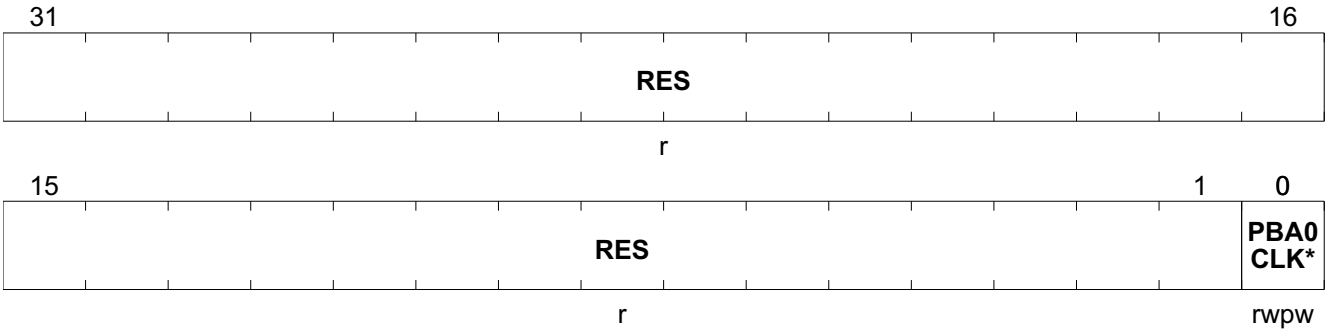
Field	Bits	Type	Description
K2DIV	6:4	rwpw	**PLL K2-Divider** The K2DIV bit is a protected bit. When the Protection Scheme is activated, this bit cannot be written directly. For more information on Protection Scheme, see **Section 7.15**. 0_H **div 2**, K2 = 2 (default) 1_H **div 3**, K2 = 3 2_H **div 4**, K2 = 4 3_H **div 5**, K2 = 5 4_H **div 6**, K2 = 6 5_H **div 7**, K2 = 7 6_H **div 8**, K2 = 8 7_H **div 9**, K2 = 9
CLKREL	3:0	rw	**Slow Down Clock Divider for f_{CCLK} Generation** This setting is effective only when the device is enabled in Slow Down Mode. *Note: f_{SYS} is further divided by the NVMCLKFAC factor to generate f_{CCLK}.* 0000_B **div 1**, f_{sys} 0001_B **div 2**, $f_{sys}/2$ 0010_B **div 3**, $f_{sys}/3$ 0011_B **div 4**, $f_{sys}/4$ 0100_B **div 8**, $f_{sys}/8$ 0101_B **div 16**, $f_{sys}/16$ 0110_B **div 24**, $f_{sys}/24$ 0111_B **div 32**, $f_{sys}/32$ 1000_B **div 48**, $f_{sys}/48$ 1001_B **div 64**, $f_{sys}/64$ 1010_B **div 96**, $f_{sys}/96$ 1011_B **div 128**, $f_{sys}/128$ 1100_B **div 192**, $f_{sys}/192$ 1101_B **div 256**, $f_{sys}/256$ 1110_B **div 384**, $f_{sys}/384$ 1111_B **div 512**, $f_{sys}/512$

Table 43 RESET of SCU_CMCON1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000500_H	RESET_TYPE_4		
VARIANT	00000500_H	VARIANT		

Clock Control Register 2

SCU_CMCON2	Offset	Reset Value
Clock Control Register 2	$04C_H$	see Table 44



Field	Bits	Type	Description
RES	31:1	r	**Reserved** This bit field is always read as zero.
PBA0CLKREL	0	rwpw	**PBA0 Clock Divider** This Flag configures the PBA0 clock divider. The PBA0CLKREL bit is a protected bit.When the Protection Scheme is activated, this bit cannot be written directly. For more information on Protection Scheme, see **Section 7.15**. 0_B **div 1**, divide by 1 1_B **div 2**, divide by 2

Table 44 RESET of SCU_CMCON2

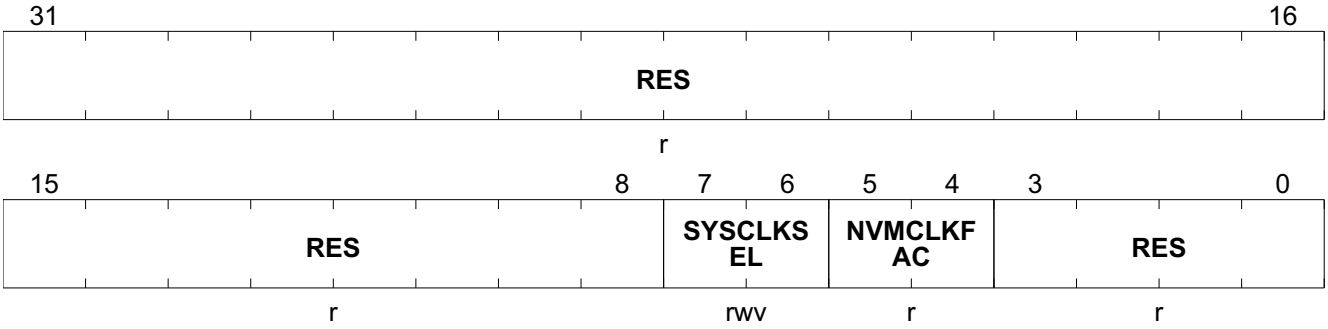
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		

7.3.6.3 System Clock Control Registers

The clock source for the system is selected via register SYSCON0.

System Control Register 0

SCU_SYSCON0	Offset	Reset Value
System Control Register 0	070_H	see Table 45



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
SYSCLKSEL	7:6	rwv	**System Clock Select** *Note: This is a PASSWD protected bit. When the protection scheme (see **Chapter 7.15**) is activated (default), this bit cannot be written directly.* This bit field defines the clock source that is used as system clock for the system operation. *Note: In normal application, it is expected that the system is running on the PLL clock output.* 00_B **f_pll**, The PLL clock output signal f_{PLL} is used 01_B **f_osc**, The direct clock input from f_{OSC} is used 10_B **f_lpclk**, The direct low-precision clock input from f_{LP_CLK} is used. 11_B **f_int**, The direct input from internal oscillator f_{INTOSC} is used

Field	Bits	Type	Description
NVMCLKFAC	5:4	r	**NVM Access Clock Factor** This bit field defines the factor by which the system clock is divided down, with respect to the synchronous NVMACCCLK clock. Note: Can only be changed via dedicated BROM routine. 00_B **div 1**, Divide by 1 01_B **div 2**, Divide by 2 10_B **div 3**, Divide by 3 11_B **div 4**, Divide by 4
RES	3:0	r	**Reserved** Returns 0 if read; should be written with 0.

Table 45 RESET of SCU_SYSCON0

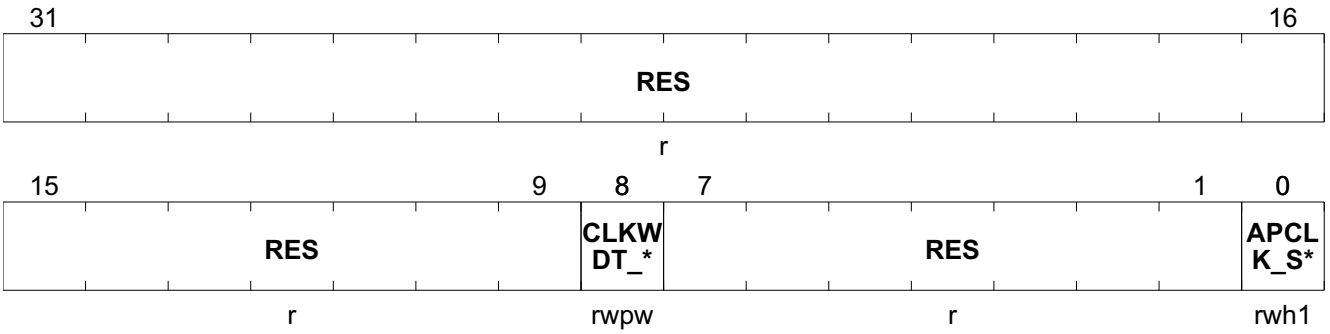
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000080_H	RESET_TYPE_4		

7.3.6.4 Analog Peripherals Clock Control Registers

The clock frequency for the analog modules is selected via register APCLK. The APCLK is used as operating clock for all analog peripherals. For this reason it is important to choose always the required frequency range, if system clock is changed.

Analog Peripheral Clock Control Register

SCU_APCLK_CTRL	Offset	Reset Value
Analog Peripheral Clock Control Register	054_H	see Table 46



Field	Bits	Type	Description
RES	31:9	r	**Reserved** Returns 0 if read; should be written with 0.
CLKWDT_IE	8	rwpw	**Clock Watchdog Interrupt Enable** The CLKWDT_IE bit is a PASSWD protected bit. 0_B **disabled**, Interrupt disabled 1_B **enabled**, Interrupt enabled
RES	7:1	r	**Reserved** Returns 0 if read; should be written with 0.
APCLK_SET	0	rwh1	**Set and Overtake Flag for Clock Settings** This Flag makes the APCLK1, APCLK2 Settings valid. Note: APCLK_SET is cleared by hardware once the clock setting are overtaken 0_B **ignore**, Clock Settings are ignored (previous values are held) 1_B **update**, Clock Settings are overtaken

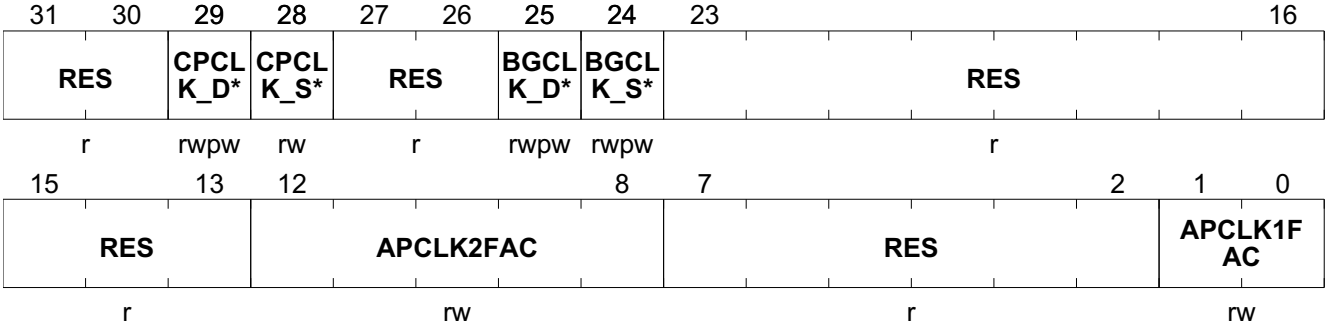
Table 46 RESET of SCU_APCLK_CTRL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		

Analog Peripheral Clock Register

The clock source for the analog modules is selected via register APCLK.

SCU_APCLK	Offset	Reset Value
Analog Peripheral Clock Register	058_H	see Table 47



Field	Bits	Type	Description
RES	31:30	r	**Reserved** Returns 0 if read; should be written with 0.
CPCLK_DIV	29	rwpw	**Charge Pump Clock Divider** 0_B **div 2**, divide by 2 1_B **div 1**, divide by 1 **The CPCLK_DIV bit is a PASSWD protected bit.**
CPCLK_SEL	28	rw	**Charge Pump Clock Selection** 0_B **LP_CLK**, LP_CLK is selected 1_B **f_sys**, f_{sys} is selected Note: If SYSCLKSEL[1] = '1' the default CPCLK_SEL = "0" (LP_CLK) is taken
RES	27:26	r	**Reserved** Returns 0 if read; should be written with 0.
BGCLK_DIV	25	rwpw	**Bandgap Clock Divider** This Flag configures the bandgap clock divider. The BGCLK_DIV bit is a PASSWD protected bit. 0_B **div 2**, divide by 2 1_B **div 1**, divide by 1
BGCLK_SEL	24	rwpw	**Bandgap Clock Selection** This Flag selects the bandgap clock. The BGCLK_SEL bit is a PASSWD protected bit. Note: If SYSCLKSEL[1] = '1' the default BGCLK_SEL = "0" (LP_CLK) is taken 0_B **LP_CLK**, LP_CLK is selected 1_B **f_sys**, f_{sys} is selected
RES	23:13	r	**Reserved** Returns 0 if read; should be written with 0.

Field	Bits	Type	Description
APCLK2FAC	12:8	rw	**Slow Down Clock Divider for TFILT_CLK Generation** This setting is effective only when the APCLK_SET = 1. There are 32 possible clock divider values according to the following examples: 00000_B **div 1**, f_{sys} 00001_B **div 2**, $f_{sys}/2$ 00010_B **div 3**, $f_{sys}/3$ 00011_B **div 4**, $f_{sys}/4$ 00100_B **div 5**, $f_{sys}/5$ 00101_B **div 6**, $f_{sys}/6$ 00110_B **div 7**, $f_{sys}/7$ 00111_B **div 8**, $f_{sys}/8$ 01000_B **div 9**, $f_{sys}/9$ 01001_B **div 10**, $f_{sys}/10$ 01010_B **div 11**, $f_{sys}/11$ 01011_B **div 12**, $f_{sys}/12$, ... 11110_B **div 31**, $f_{sys}/31$ 11111_B **div 32**, $f_{sys}/32$ *Note: The setting is only overtaken if APCLK2FAC setting is greater or equal than APCLK1FAC* **Notes** 1. *If SYSCLKSEL[1:0] = "10"(LP_CLK) the default APCLK2FAC = 8 is taken* 2. *if SYSCLKSEL[1:0] = "11" (pll_40m_clk_i from 80 MHz oscillator) and OSC80MDIV[1] = 0 the value APCLK2FAC = 19 is taken* 3. *if SYCLKSEL[1:0] = "11" (pll_40m_clk_i from 80 MHz oscillator) and OSC80MDIV[1] = 1 the value APCLK2FAC = 9 is taken* 4. *f_{SYS} is further divided by the APCLK2FAC factor to generate TFILT_CLK. The clock should be always at 2 MHz.*
RES	7:2	r	**Reserved** Always read as zero.

Field	Bits	Type	Description
APCLK1FAC	1:0	rw	**Analog Module Clock Factor** This bit field defines the factor by which the system clock is divided down, with respect to the synchronous MI_CLK clock. *Note: The setting is only overtaken if APCLK2FAC setting is greater or equal than APCLK1FAC* 00_B **div 1**, Divide by 1 01_B **div 2**, Divide by 2 10_B **div 3**, Divide by 3 11_B **div 4**, Divide by 4 The APCLK1FAC bit is not a protected bit. This setting is only effective when APCLK_SET = 1. Note: If SYSCLKSEL[1] = '1' (LP_CLK) the default APCLK1FAC = "00" is taken (divide by 1) if SYCLKSEL[1:0] = "11" and OSC80MDIV = 0 the value APCLK1FAC = "01" is taken if SYSCLKSEL[1:0] = "11" and OSC80MDIV[1]= 1 the value APCLK1FAC = "00" is taken

Table 47 RESET of SCU_APCLK

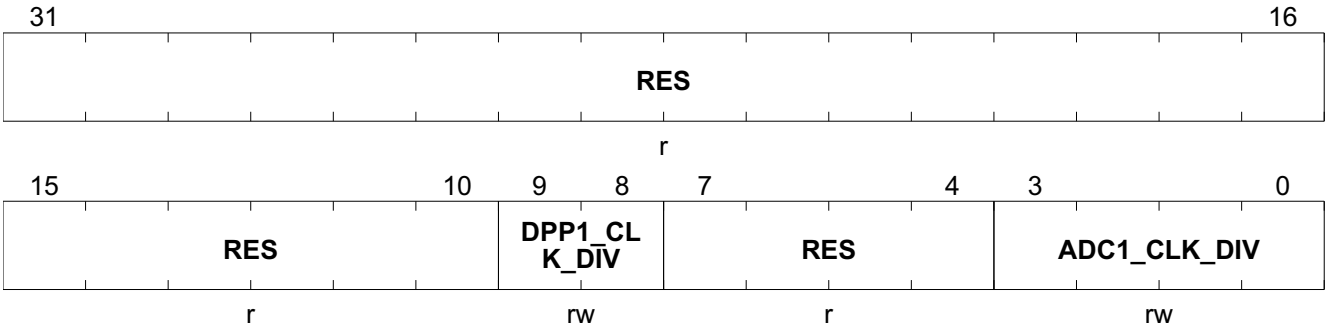
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		
VARIANT	00000000_H	VARIANT		

Table 48 Suggested Value for APCLK

Clock Frequency	APCLK1FAC	APCLK2FAC
18 Mhz (lp_clk)	00_H (default)	08_H (default)
24 Mhz (Pll clk)	00_H	$0B_H$
40 Mhz (Pll clk)	01_H	13_H

ADC1 Peripheral Clock Register

SCU_ADC1_CLK	Offset	Reset Value
ADC1 Peripheral Clock Register	$06C_H$	see Table 49



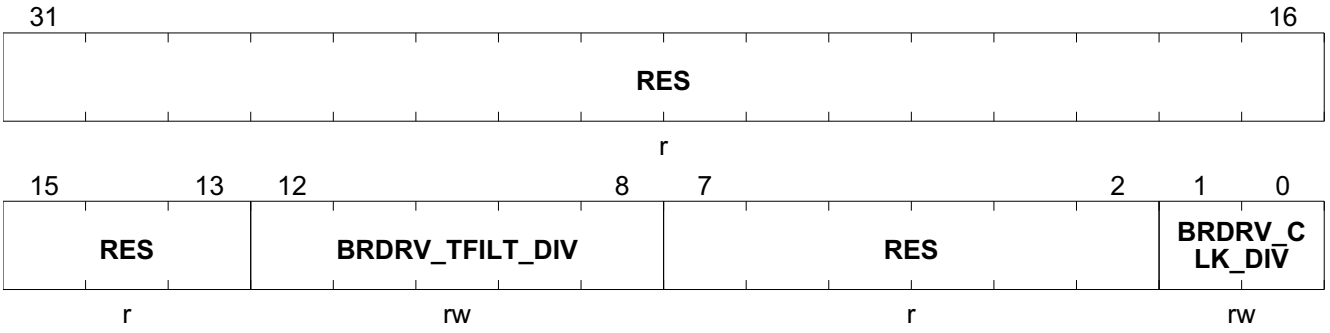
Field	Bits	Type	Description
RES	31:10	r	**Reserved** Returns 0 if read; should be written with 0.
DPP1_CLK_DIV	9:8	rw	**ADC1 Post processing clock divider** This bit field defines the factor by which the system clock is divided for the post processing of ADC1. 00_B **div 1**, Divide by 1 01_B **div 2**, Divide by 2 10_B **div 3**, Divide by 3 11_B **div 4**, Divide by 4
RES	7:4	r	**Reserved** Returns 0 if read; should be written with 0.
ADC1_CLK_DIV	3:0	rw	**ADC1 Clock divider** This bit field defines the factor by which the divided system clock from DPP1_CLK_DIV is divided additionally for ADC1 core clock. There are 16 possible clock divider values according to the following examples: 0000_B **div 1**, Divide by 1 0001_B **div 2**, Divide by 2 0010_B **div 3**, Divide by 3 0011_B **div 4**, Divide by 4 0100_B **div 5**, Divide by 5 0101_B **div 6**, Divide by 6 0110_B **div 7**, Divide by 7 0111_B **div 8**, Divide by 8, ... 1110_B **div 15**, Divide by 15 1111_B **div 16**, Divide by 16

Table 49 RESET of SCU_ADC1_CLK

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		
TRIM_100_TP	00000000_H	RESET		

BRDRV Peripheral Clock Register

SCU_BRDRV_CLK	Offset	Reset Value
BDrv Peripheral Clock Register	150_H	see Table 50



Field	Bits	Type	Description
RES	31:13	r	**Reserved** Returns 0 if read; should be written with 0.

Field	Bits	Type	Description
BRDRV_TFILT_DIV	12:8	rw	**Slow Down Clock Divider for TFILT_CLK Generation** This setting is effective only when the APCLK_SET = 1. There are 32 possible clock divider values according to the following examples: 00000_B **div 1**, f_{sys} 00001_B **div 2**, $f_{sys}/2$ 00010_B **div 3**, $f_{sys}/3$ 00011_B **div 4**, $f_{sys}/4$ 00100_B **div 5**, $f_{sys}/5$ 00101_B **div 6**, $f_{sys}/6$ 00110_B **div 7**, $f_{sys}/7$ 00111_B **div 8**, $f_{sys}/8$ 01000_B **div 9**, $f_{sys}/9$ 01001_B **div 10**, $f_{sys}/10$ 01010_B **div 11**, $f_{sys}/11$ 01011_B **div 12**, $f_{sys}/12$, ... 11110_B **div 31**, $f_{sys}/31$ 11111_B **div 32**, $f_{sys}/32$ *Note: The setting is only overtaken if BRDRV_TFILT_DIV setting is greater or equal than BRDRV_CLK_DIV* **Notes** 1. *If SYSCLKSEL[1:0] = "10" (LP_CLK) the default BRDRV_TFILT_DIV = 4 is taken* 2. *if SYSCLKSEL[1:0] = "11" (pll_40m_clk_i from 80 MHz oscillator) and OSC80MDIV[1] = 0 the value BRDRV_TFILT_DIV = 9 is taken* 3. *if SYCLKSEL[1:0] = "11" (pll_40m_clk_i from 80 MHz oscillator) and OSC80MDIV[1] = 1 the value BRDRV_TFILT_DIV = 4 is taken* 4. *f_{SYS} is further divided by the BRDRV_TFILT_DIV factor to generate TFILT_CLK. The clock should be always configured to 4 MHz.*
RES	7:2	r	**Reserved** Returns 0 if read; should be written with 0.

Field	Bits	Type	Description
BRDRV_CLK_DIV	1:0	rw	**Bridge Driver Module Clock Factor** This bit field defines the factor by which the system clock is divided down, with respect to the synchronous MI_CLK clock. *Note: The setting is only overtaken if BRDRV_TFILT_DIV setting is greater or equal than BRDRV_CLK_DIV* 00_B **div 1**, Divide by 1 01_B **div 2**, Divide by 2 10_B **div 3**, Divide by 3 11_B **div 4**, Divide by 4 The APCLKFAC bit is not a protected bit. This setting is only effective when APCLK_SET = 1. Note: If SYSCLKSEL[1:0] = “10”(LP_CLK) the default BRDRV_CLK_DIV = “00” is taken (divide by 1) if SYCLKSEL[1:0] = “11” and OSC80MDIV = 0 the value BRDRV_CLK_DIV = “01” is taken if SYSCLKSEL[1:0] = “11” and OSC80MDIV = 1 the value BRDRV_CLK_DIV = “00” is taken

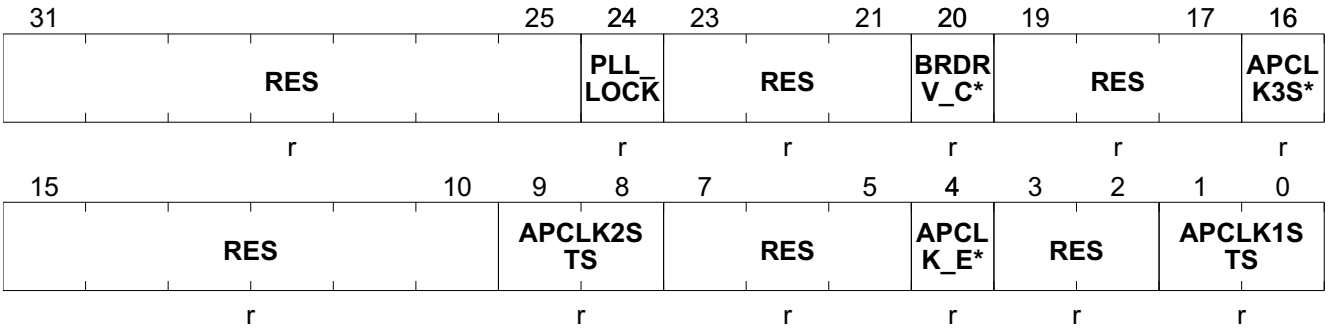
Table 50 RESET of SCU_BRDRV_CLK

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		
VARIANT	00000000_H	VARIANT		

Analog Peripheral Clock Status Register

The clock source for the analog modules is selected via register APCLK.

SCU_APCLK_STS	Offset	Reset Value
Analog Peripheral Clock Status Register	$05C_H$	see Table 51



Field	Bits	Type	Description
RES	31:25	r	**Reserved** Returns 0 if read; should be written with 0.
PLL_LOCK	24	r	**PLL LOCK Status** This bit field indicates the PLL lock status. 0_B **no lock**, PLL has not locked 1_B **lock**, PLL has locked
RES	23:21	r	**Reserved** Returns 0 if read; should be written with 0.
BRDRV_CLK_ERR_STS	20	r	**BRDRV CLK Error Status** This bit field reflects that the clock settings for SCU_BRDRV_CLK.BRDRV_CLK_DIV and SCU_BRDRV_CLK.BRDRV_TFILT_DIV were blocked for being written. 0_B **no Error**, no Error writing was not blocked 1_B **Error**, Error writing was blocked
RES	19:17	r	**Reserved** Returns 0 if read; should be written with 0.
APCLK3STS	16	r	**Loss of Clock Status** This bit field indicate the loss of clock status. 0_B **no loss**, No loss of clock 1_B **loss**, Loss of clock occurred
RES	15:10	r	**Reserved** Returns 0 if read; should be written with 0.

Field	Bits	Type	Description
APCLK2STS	9:8	r	**Analog Peripherals Clock Status** This bit field reflects the analog peripheral clock source status that is used as system clock for the analog module operation. The implemented clock watchdog (**see Chapter SCU_PM**) is monitoring the frequency of the analog subsystem. If the clock is not inside the required range, a system reset will be issued. *Note: The functionality of the analog modules can only be guaranteed, when their clock is in the required range.* 00_B **OK**, The TFILT_CLK clock is in the required range 01_B **Too high**, The TFILT_CLK clock exceeds the higher limit 10_B **Too low**, The TFILT_CLK clock exceeds the lower limit 11_B **Out of Limit**, The TFILT_CLK clock is not inside the specified limit.
RES	7:5	r	**Reserved** Returns 0 if read; should be written with 0.
APCLK_ERR_STS	4	r	**APCLK Error Status** This bit field reflects that the clock settings for SCU_APCLK_CTRL.APCLK1FAC and SCU_APCLK_CTRL.APCLK2FAC were blocked for being written. 0_B **no Error**, no Error writing was not blocked 1_B **Error**, Error writing was blocked
RES	3:2	r	**Reserved** Returns 0 if read; should be written with 0.

Field	Bits	Type	Description
APCLK1STS	1:0	r	**Analog Peripherals Clock Status** This bit field reflects the analog peripheral clock source status that is used as system clock for the analog module operation. The implemented clock watchdog (**see Chapter SCU_PM**) is monitoring the frequency of the analog subsystem. If the clock is not inside the required range, a system reset will be issued. *Note: The functionality of the analog modules can only be guaranteed, when their clock is in the required range.* 00_B **OK**, The MI_CLK clock is in the required range 01_B **too high**, The MI_CLK clock exceeds the higher limit 10_B **too low**, The MI_CLK clock exceeds the lower limit 11_B **out of limit**, The MI_CLK clock is not inside the specified limit.

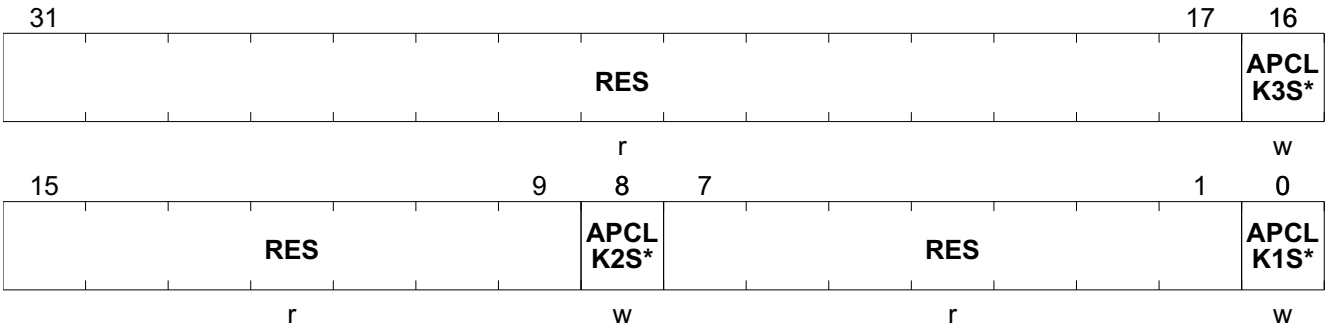
Table 51 RESET of SCU_APCLK_STS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		

Analog Peripheral Clock Status Clear Register

The clock source for the analog modules is selected via register APCLK.

SCU_APCLK_SCLR	Offset	Reset Value
Analog Peripheral Clock Status Clear Register	064_H	see Table 52



Field	Bits	Type	Description
RES	31:17	r	**Reserved** Returns 0 if read; should be written with 0.
APCLK3SCLR	16	w	**Analog Peripherals Clock 3 Status Clear** This bit field is used for APCLK3 Status Clear.
RES	15:9	r	**Reserved** Returns 0 if read; should be written with 0.
APCLK2SCLR	8	w	**Analog Peripherals Clock Status Clear** This bit field is used for APCLK2 Status Clear.
RES	7:1	r	**Reserved** Returns 0 if read; should be written with 0.
APCLK1SCLR	0	w	**Analog Peripherals Clock Status Clear** This bit field is used for APCLK1 Status Clear.

Table 52 RESET of SCU_APCLK_SCLR

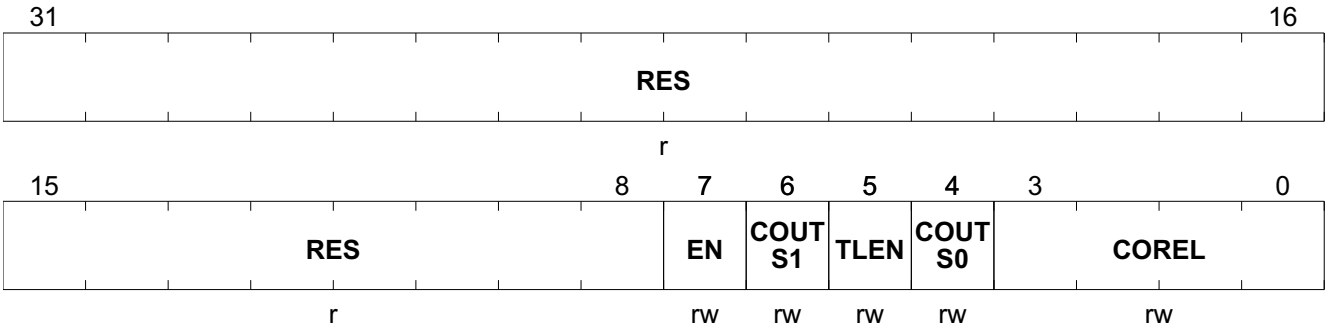
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		

7.3.6.5 External Clock Control Register

This register controls the setting of external clock for CLKOUT.

Clock Output Control Register

SCU_COCON	Offset	Reset Value
Clock Output Control Register	$0B4_H$	see Table 53



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
EN	7	rw	**CLKOUT Enable** 0_B **disable**, No external clock signal is provided 1_B **enable**, The configured external clock signal is provided
COUTS1	6	rw	**Clock Out Source Select Bit 1** 0_B **f_cclk**, f_{CCLK} is selected. 1_B **COUTS0**, Based on setting of COUTS0.
TLEN	5	rw	**Toggle Latch Enable** Enable this bit if 50% duty cycle is desired on CLKOUT. This bit is only applicable when both COUTS1 and COUTS0 are set to 1. 0_B **disable**, Toggle Latch is disabled. Clock output frequency is chosen by the bit field COREL. 1_B **enable**, Toggle Latch is enabled. Clock output frequency is half of the frequency that is chosen by the bit field COREL. The resulting output frequency has 50% duty cycle.
COUTS0	4	rw	**Clock Out Source Select Bit 0** This bit is effective only if COUTS1 is set to 1. 0_B **Osc**, Oscillator output frequency is selected. 1_B **COREL**, Clock output frequency is chosen by the bit field COREL.

Field	Bits	Type	Description
COREL	3:0	rw	**Clock Output Divider** 0000_B **div 1**, f_{sys} 0001_B **div 2**, $f_{sys}/2$ 0010_B **div 3**, $f_{sys}/3$ 0011_B **div 4**, $f_{sys}/4$ 0100_B **div 6**, $f_{sys}/6$ 0101_B **div 8**, $f_{sys}/8$ 0110_B **div 10**, $f_{sys}/10$ 0111_B **div 12**, $f_{sys}/12$ 1000_B **div 14**, $f_{sys}/14$ 1001_B **div 16**, $f_{sys}/16$ 1010_B **div 18**, $f_{sys}/18$ 1011_B **div 20**, $f_{sys}/20$ 1100_B **div 24**, $f_{sys}/24$ 1101_B **div 32**, $f_{sys}/32$ 1110_B **div 36**, $f_{sys}/36$ 1111_B **div 40**, $f_{sys}/40$

Table 53 RESET of SCU_COCON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		

7.4 Reset Control

This section describes the types of reset and the effects of each reset on the TLE985xQX.

7.4.1 Types of Reset

The following reset types are recognized by the TLE985xQX.

- Power-on reset
  - Requested asynchronously and released by supply voltage V_S reaching the upper threshold. Indication is a direct analysis of V_S undervoltage.
- Brown-out reset
  - Is not differentiated by system with power-on reset.
- Wake-up reset
  - Requested asynchronously by wake-up event during power save mode.
- Hardware reset
  - Requested asynchronously by event on external reset input (pin) .
- WDT1 reset
  - Activated asynchronously by external WDT1 reset event .
- SCU Watchdog Timer (WDT) reset
  - Requested by WDT reset event.
- Soft reset
  - Requested synchronously by soft reset event.

7.4.2 Overview

When the TLE985xQX is first powered up or with brown-out condition triggered by supply voltage input(s) going below the threshold, proper voltage thresholds must be reached before the MCU system starts operation with the release of the MCU, CPU and NVM resets. With all resets (except soft and SCU watchdog timer resets), the boot configuration is latched. The CPU starts to execute from the Boot ROM firmware with the release of MCU reset.

If the system is in power save mode, it is possible to wake-up with reset. Wake-up reset is basically equivalent to power-on reset except that it is a ‘warm’ reset and certain settings or configuration of the system are maintained across the reset . A wake-up via hard reset pin while in power save mode is effected as wake-up reset.

The hardware reset function via pin can be used anytime to restart the system.

The external watchdog timer (WDT1) can trigger a WDT1 reset on the system, if the timer is not refreshed before it overflows.

Likewise, the SCU watchdog timer (WDT) can trigger a watchdog timer reset on the system if the timer is not refreshed before it overflows.

Soft reset can be triggered by application software where applicable.

Note that the boot configuration is only latched with the power-on, brown-out, WDT1, wake-up and hardware resets.

7.4.3 Module Reset Behavior

Table 54 gives an overview on how the various modules or functions of the TLE985xQX are affected with respect to the reset type. A “n” means that the module/function is reset to its default state. Refer to Table 54 for effective reset as priority.

Table 54 Effect of Reset on Modules/Functions

Module/ Function	Power-On/ Brown-Out Reset	Wake-up Reset[1)]	Hardware Reset[1)]	WDT1 Reset[1)]	WDT Reset	Soft Reset
CPU Core	n	n	n	n	n	n
SCU	n except reset indication bit	n except indication bits	n except reset indication bit	n except reset indication bit	n except certain status bits[2)]	n except certain status bits[2)]
Peripherals	n	n	n	n	n	n
Debug System	n	n	n	n	Not affected	Not affected
Port Control	n	n	n	n	n	n
FW Startup Execution	Executes all INIT	Sleep: Executes all INIT	Executes most INIT	Executes most INIT	Skips not required INIT	Skips not required INIT
On-Chip Static RAM	Initialized to 0	Sleep: Initialized to 0; Stop: Not affected	Initialized to 0 [3)]	Initialized to 0[3)]	Not affected	Not affected
NVM	n	n	n	n	n	n
Clock System incl. PLL	n	n	n	n	Not affected[4)]	Not affected[4)]

1) MCU sub-system: Hardware reset, WDT1 reset and wake-up reset (from Stop Mode or Sleep Mode) are generally HW-equivalent to power-on/brown-out reset, any exceptions are mainly due to power-on reset being a ‘cold’ start.
2) These bits include the reset requestor indication bit, the last power-on/brown-out/WDT1/wake-up reset latched boot configuration, and NMI status flags e.g. NMISR.
3) If the reset happens during a write to SRAM, the byte in the targeted write address may be corrupted.
4) All configuration including trim settings.

7.4.4 Functional Description of Reset Types

This section describes the definition and controls depending on the reset source.

7.4.4.1 Power-On / Brown-out Reset

Power-on reset is the highest level reset whereby the whole system is powered up and reset. Brown-out reset occurs when any required voltage drops below its minimum threshold.

In user mode, the system clock is switched to the PLL output at the defined frequency of the device.

7.4.4.2 Wake-up Reset

Wake-up reset occurs due to enabled event on defined functional input pins leading to reset of device while the device was in power-save mode. Wake-up reset from sleep and power-down (stop) mode is differentiated by respective indicator bits In case of wake-up from Sleep Mode, reset is always effected. Note that event on RESET input pin while device was in power-save mode is effectively a hardware reset. In this case, the wake-up indicator bit WKRS is also set.

Wake-up reset has the next highest priority after power-on/brown-out reset.

In user mode, the system clock is switched to the PLL output at the defined frequency of the device.

7.4.4.3 Hardware Reset

Hardware reset is requested asynchronously by event on external RESET (low active) input pin, and has the next highest priority after wake-up reset.

In case of hardware reset is activated while the device is in power-save mode, this is effectively a wake-up reset. Refer **Chapter 7.4.4.2**.

In user mode, the system clock is switched

For details of programming the reset blind time of the external RESET (low active) input pin see the corresponding reset pin blind time register, PMU_CNF_RST_TFB.RST_TFB.

7.4.4.4 WDT1 Reset

WDT1 reset occurs due to WDT1 timer overflow or when servicing in a closed window, and has the next highest priority after hardware reset.

In user mode, the system clock is switched to the PLL output at the defined frequency of the device.

7.4.4.5 WDT / Soft Reset

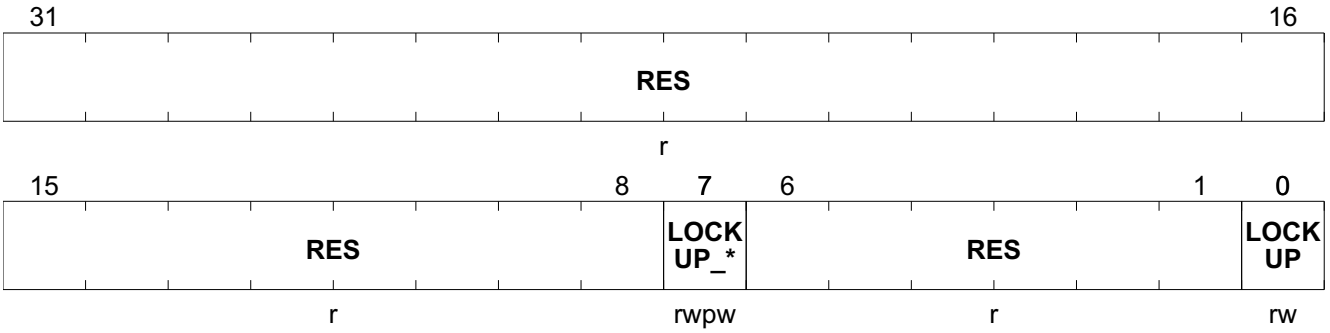
WDT reset occurs due to WDT timer overflow; Soft reset occurs due to software set of the soft reset request bit.

These two resets are at the same priority level (same effect on system) and has the lowest priority level. With these resets, the device continues running on the previous clock system configuration.

7.4.5 Reset Register Description

Reset Control Register

SCU_RSTCON	Offset	Reset Value
Reset Control Register	068_H	see Table 55



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
LOCKUP_EN	7	rwpw	**Lockup Reset Enable Flag** The LOCKUP_EN bit is a protected bit. When the Protection Scheme is activated, this bit cannot be written directly. For more information on Protection Scheme, see **Section 7.15**. *Note: This bit is RESET_TYPE_4* 0_B **disable**, Lockup is disabled. 1_B **enable**, Lockup is enabled.
RES	6:1	r	**Reserved** Returns 0 if read; should be written with 0.
LOCKUP	0	rw	**Lockup Flag** The LOCKUP bit is a protected bit. When the Protection Scheme is activated, this bit cannot be written directly. For more information on Protection Scheme, see **Section 7.15**. *Note: This bit is RESET_TYPE_3* 0_B **disable**, Lockup Status not active. 1_B **enable**, Lockup Status active.

Table 55 RESET of SCU_RSTCON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

7.4.6 Booting Scheme

After any power-on reset, brown-out reset, hardware reset, WDT1 reset or wake-up reset, the pins TMS, P0.0, P0.2 together choose different modes. **Table 56**shows the boot selection options available in the TLE985xQX.

Table 56 TLE985xQX Boot Options

TMS/SWD	P0.0	MODE
0	x	User Mode / BSL Mode
1	1	Debug Mode with Serial Wire (SW) port

7.5 Power Management

This section describes the features and functionality provided for power management of the device.

7.5.1 Overview

The TLE985xQX power-management system allows software to configure the various processing units so that they automatically adjust to draw the minimum necessary power for the application.

There are four power modes: Active Mode, Slow Down Mode, Stop Mode and Sleep Mode, as shown in **Figure 30**. Sleep Mode is a special case which can only be exited with a system reset.

The operation of the system components in each of these states can be configured by software. The power modes provide flexible reduction of power consumption through a combination of techniques, including:

- Stopping the CPU clock
- Stopping the clocks of other system components individually
- Clock-speed reduction of some peripheral components
- Power-down of the entire system with fast restart capability
- Reducing or removing the power supply to power domains

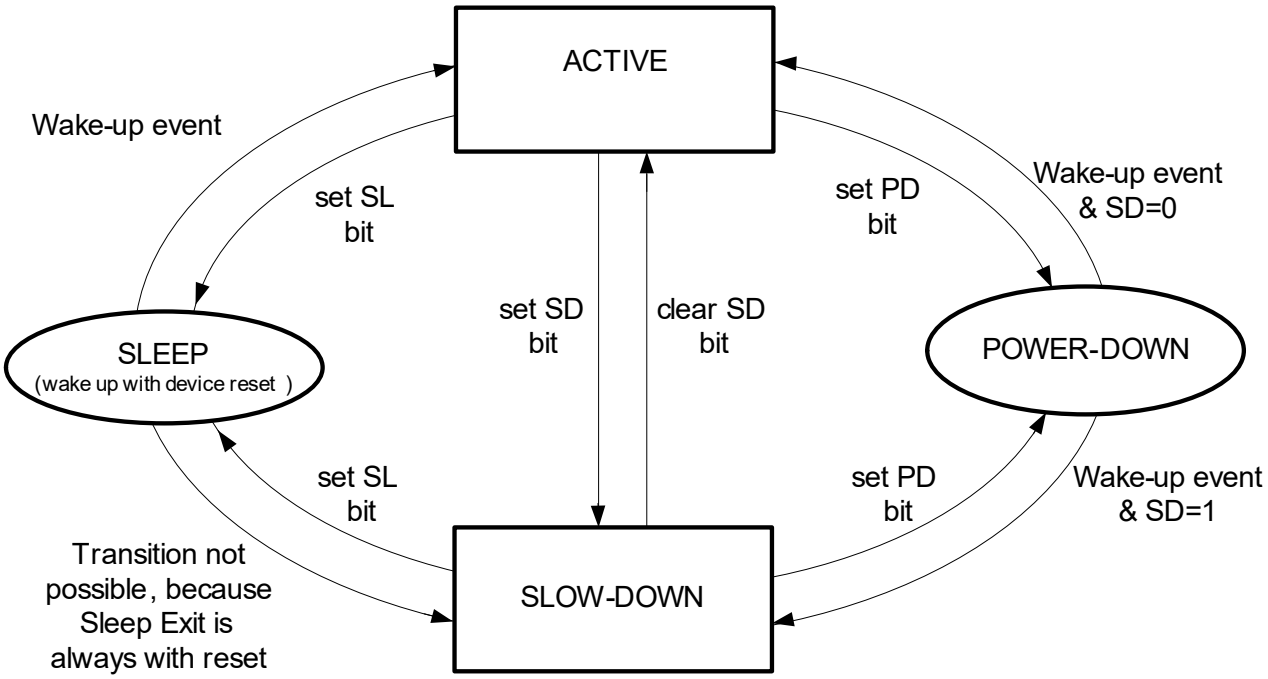


Figure 30 Transition between Various Modes of Operation (without reset)

In Slow Down mode, the clock generation unit is instructed to reduce its clock frequency so that the clock to the system, i.e. Core and peripheral, will be divided by a programmable factor.

In Stop Mode, the clock is turned off. Hence, it cannot be awakened by an interrupt or the Watchdog Timer. It will be awakened only when it receives an external wake-up signal or reset signal. The application must be prepared that the TLE985xQX is served with one of these signals. A wake-up circuit is used to detect enabled wake-up signal(s) and activate the Stop Mode wake-up. During Stop Mode, this circuit remains active.

In Sleep Mode, the power supply to the whole MCU subsystem is removed. On detection of wake-up event, a system reset is generated and the MCU is reset to default configuration then restart operation as initialized.

The priority for entry to the power-save modes starting from the highest is Sleep Mode, Stop Mode, then Slow Down Mode.

7.5.2 Functional Description

This section describes the power-save modes, their operations, and entry and exit. It also describes the respective behavior of TLE985xQX system components.

7.5.2.1 Slow Down Mode

The Slow Down Mode is used to reduce the power consumption by decreasing the internal clock in the device.

The Slow Down Mode is activated by setting the bit SD in SFR PMCON0. The bit field CMCON1.CLKREL is used to select different slow down frequency. The CPU and peripherals are clocked at this lower frequency. The Slow Down Mode is terminated by clearing bit SD.

7.5.2.2 Stop Mode

In the Stop Mode, the NVM is put into NVM shutdown mode (analog and digital except MapRAM shut down). The 5 V (VDDP) power supply to the analog modules ADC and PLL & internal oscillator is not removed. The MCU digital and NVM MapRAM is powered by the 1.5V (VDDC) regulator (0.9 V). All functions of the microcontroller are stopped while the contents of the NVM, on-chip RAM, RAM, and the SFRs are maintained. As for the external ports, all digital pads are still powered.

In Stop Mode, the clock is turned off. Hence, the system cannot be awakened by an interrupt or the Watchdog Timer. It will be awakened only when it receives an external wake-up signal (with or without a following system reset) or with reset by asserting the hard reset pin.

Software requests Stop Mode by setting the bit **PMCON0.PD** to 1. In addition to this Flag the **WFI** or **WFE** instruction has to be executed. When the controller will finish its currently executed interrupt task it will enter the Stop Mode. Figure below shows the required sequence to enter stop mode properly (please make sure not to disturb this sequence by pending wake events or interrupts):

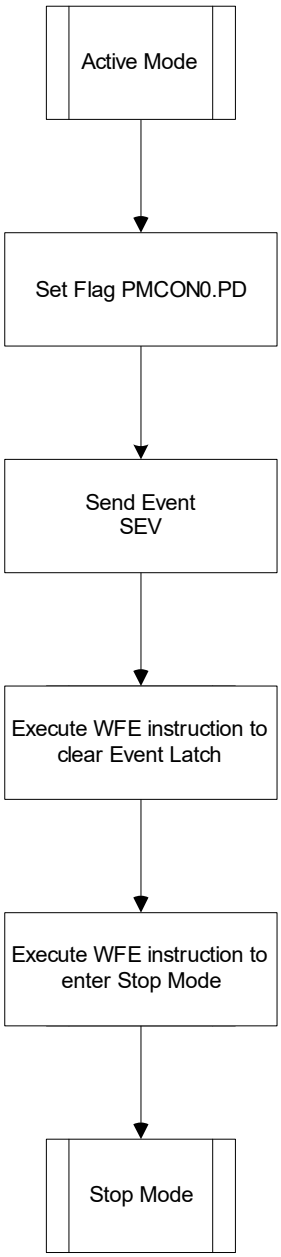


Figure 31 Stop Mode Entry Programming Sequence

Exiting Stop Mode

Stop Mode can be exited by active edge on the enabled wake-up pin(s) or by asserting the hard reset pin.

The wake-up circuitry will perform a sequence of predefined actions such as restore all supply voltages, restore modules to operational mode including the oscillator and PLL. On stable clock per user configuration is restored, peripheral clock gating, CPU clock gating is removed and the CPU starts to run from the instruction following the one that sets the PD bit. It is required by the user code to insert three NOP instructions following the one that sets the PD bit.

Note that if user has selected the PLL output as system clock (typical usage) but lock status of the PLL cannot be achieved, the device cannot wake up and shall hang in this state until a device reset.

7.5.2.2.1 Usage of Arm® Cortex®-M0 Core Low Power Modes for Stop and Sleep Mode

The Arm® Cortex®-M0 Core provides two low power modes, which are Sleep and Deep sleep. For stop mode of the system the Deep sleep will be used. To enable the deep sleep mode the System Control Register at address $E000ED10_H$. When the user wants to enter sleep mode it can be done via two different instructions:

- **WFI**
- **WFE**

When the controller enters stop mode via WFI instruction, it executes the lowest prior pending interrupt and after that enters sleep mode. This feature is not recommended to be used for normal operation using stop mode, because the controller would only operate interrupt triggered.

When the WFE instruction is used, the controller starts to operate triggered by an external event. If CPU will be woken up be this external event, it stays in thread mode and continue to execute the code before it entered stop mode.

This is the recommended procedure to enter stop mode.

7.5.2.3 Sleep Mode

In the Sleep Mode, the supply to the whole MCU subsystem including the ADC, PLL and NVM is removed. The wake-up detection circuitry remains supplied. Only contents of non-volatile memory are retained. As for the external ports, only the wake-up pads are still powered. The supply to ADC pads is removed.

Sleep Mode is always exit with a system reset, which is triggered by active edge on the enabled wake-up pin(s). It is not possible to exit Sleep Mode by asserting the hard reset pin as the digital 5 V pads will not be powered.

Software requests Sleep Mode by setting the bit PMCON0.SL to 1.

Exiting Sleep Mode

Sleep Mode can only be exited with a system reset, triggered by active edge on the enabled wake-up pin(s).

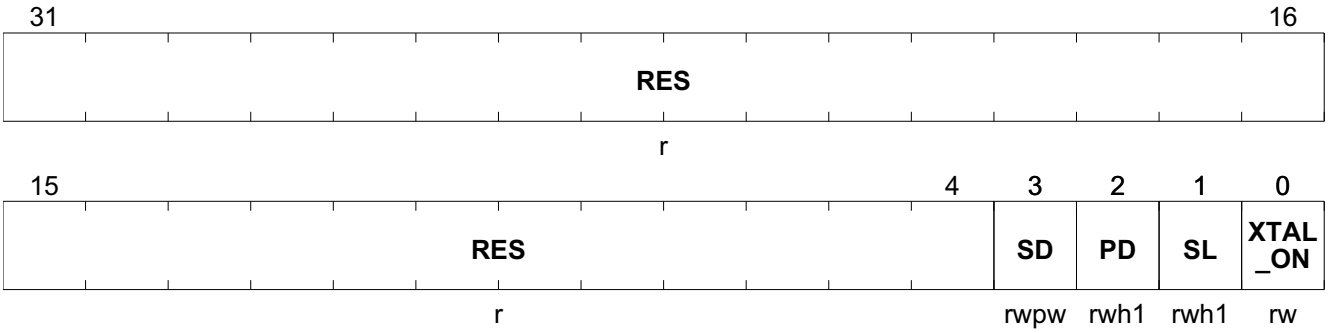
Notes

1. *Ready for first LIN message at > 400 µs (assume 64 Kbyte MapRAM init): start-up boot, NVM pumps ramp up including SFR and MapRAM init.*
2. *To avoid power switching, dedicated VREGs are provided for necessary power domains.*

7.5.3 Register Description

Power Mode Control Register 0

SCU_PMCON0	Offset	Reset Value
Power Mode Control Register 0	040_H	see Table 57



Field	Bits	Type	Description
RES	31:4	r	**Reserved** Returns 0 if read; should be written with 0.
SD	3	rwpw	**Slow Down Mode Enable. Active High.** Setting this bit will cause the chip to go into slow down mode. Reset by user. The SD bit is a protected bit. When the Protection Scheme is activated, this bit cannot be written directly. For more information on Protection Scheme, see **Section 7.15**.
PD	2	rwh1	**Power Down Mode Enable. Active High.** Setting this bit will cause the chip to go into a Power Down mode. Reset by wake-up circuit. The PD bit is a protected bit. When the Protection Scheme is activated, this bit cannot be written directly. For more information on Protection Scheme, see **Section 7.15**.
SL	1	rwh1	**Sleep Mode Enable. Active High.** Setting this bit will cause the chip to go into Sleep Mode. Reset by wake-up circuit. The SL bit is a protected bit. When the Protection Scheme is activated, this bit cannot be written directly. For more information on Protection Scheme, see **Section 7.15**.

Field	Bits	Type	Description
XTAL_ON	0	rw	**OSC_HP Operation in Power Down Mode** This provides user the option for reduced power consumption in the Power Down mode. It must be noted that the startup time of OSC_HP can be in the range of some milliseconds. Alternatively for fast wake-up from Power Down mode while avoiding this power consumption, the user can selectively enable internal oscillator as clock source and disable OSC_HP before enable Power Down mode. 0_B **PD**, OSC_HP (XTAL) will be put to Power Down mode by hardware in power save mode. 1_B **ON**, OSC_HP (XTAL) continues to operate in Power Down mode, if enabled by **SCU_OSC_CON**.XPD.

Table 57 RESET of SCU_PMCON0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

7.6 Interrupt Management

This section describes the management of interrupts by the system control unit.

7.6.1 Overview

The Interrupt Management sub-module in the SCU controls the non-core-generated interrupt requests to the core. The core has one non-maskable interrupt (NMI) node and total 24 maskable interrupt nodes. **Figure 32** shows the block diagram of the Interrupt Management sub-module.

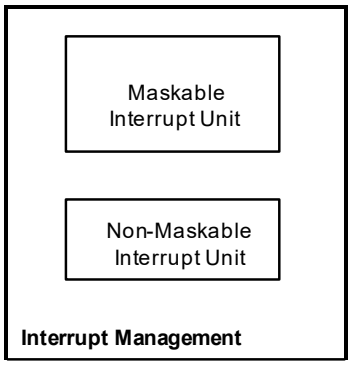


Figure 32 Interrupt Management Block Diagram

The non-maskable interrupt unit controls the NMI requests. Incoming NMI request is not maskable and in this sense, differs from the regular interrupts. In addition, NMI request always has the highest priority to be serviced. In the TLE985xQX, eight different sources can generate an NMI: watchdog timer prewarning, PLL loss-of-lock, oscillator watchdog event, NVM map error, Memory ECC error, Debug Mode user IRAM event and supply prewarning. Some NMI sources can be triggered by one of several events. These NMI sources are ORed to generate an NMI interrupt directly to the core. The triggering NMI sources/events are indicated in the NMI Status Register (NMISR), and in some cases the event flags are located in the peripheral register. The NMI node source control is via the NMI Control Register (NMICON).

There are generally 3 types of maskable inputs into the core: internal, external and extended interrupts. The maskable interrupt unit will generate the respective interrupt node request to the core and will maintain corresponding SCU flags and control. In general, to support all types of peripheral interrupts, an interrupt node of the core may be shared among several interrupt sources.

7.6.1.1 External Interrupts

The generation of interrupt request from an external source by edge detection in SCU is shown in **Figure 33**. External interrupts can be positive, negative or double edge triggered. Register EXICON0 specifies the active edge for the external interrupt.

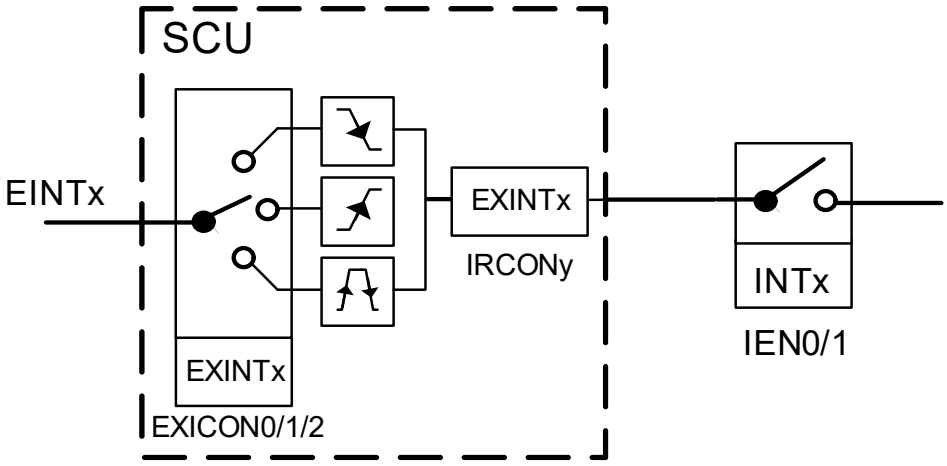


Figure 33 Interrupt Request Generation of External and Peripheral Interrupts

7.6.1.2 Extended Interrupts

Extended interrupts are for non-core on-chip peripherals for core-external trigger of interrupt requests to the core.

Interrupt signals from such on-chip peripherals are pulse triggered and active for two clock cycles. These interrupt signals belonging to the same interrupt node will be latched as one direct interrupt request to the core. IRCONx (where x = 0-1, 3-4) or peripheral registers hold the interrupt event flags for these extended and external interrupt events. Corresponding bits in the Interrupt Enable Registers (IEN) within the core may block or transfer these interrupt requests to the core interrupt controller. An enabled interrupt request is acknowledged when the core vectors to the interrupt routine. The software routine should clear the interrupt flags in the IRCONx registers.

As there are more peripheral interrupts than interrupt nodes supported by the core, some interrupts are multiplexed to the same interrupt node. Where possible and necessary, critical peripheral interrupts (e.g. SC) have their own dedicated interrupt node.

7.6.2 Interrupt Node Assignment

For an overview of the interrupt node assignment of TLE985xQX please refer to**Table 170** and **Table 171**.

7.6.3 Interrupt Registers

Interrupt registers are used for interrupt node enable, external interrupt control, interrupt flags and interrupt priority setting.

Table 58 Register Address Space

Module	Base Address	End Address	Note
SCU	50005000_H	$50005FFF_H$	SCU

Table 59 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Interrupt Registers, Interrupt Node Enable Registers			
SCU_IEN0	Interrupt Enable Register 0	$01C_H$	$0000\ 0000_H$
SCU_VTOR	Vector Table Reallocation Register	020_H	$0000\ 0000_H$
SCU_NMICON	NMI Control Register	024_H	$0000\ 0000_H$
Interrupt Registers, External Interrupt Control Registers			
SCU_EXICON0	External Interrupt Control Register 0	028_H	$0000\ 0030_H$
SCU_EXICON1	External Interrupt Control Register 1	$02C_H$	$0000\ 0000_H$
SCU_WAKECON	Wakeup Interrupt Control Register	$0EC_H$	$0000\ 0000_H$
Interrupt Registers, Interrupt Flag Registers			
SCU_IRCON0	Interrupt Request Register 0	004_H	$0000\ 0000_H$
SCU_IRCON0CLR	Interrupt Request Clear Register 0	178_H	$0000\ 0000_H$
SCU_IRCON1	Interrupt Request Register 1	008_H	$0000\ 0000_H$
SCU_IRCON1CLR	Interrupt Request Clear Register 1	$17C_H$	$0000\ 0000_H$
SCU_IRCON2	Interrupt Request Register 2	$00C_H$	$0000\ 0000_H$
SCU_IRCON2CLR	Interrupt Request Clear Register 2	190_H	$0000\ 0000_H$
SCU_IRCON3	Interrupt Request Register 3	010_H	$0000\ 0000_H$
SCU_IRCON3CLR	Interrupt Request Clear Register 3	194_H	$0000\ 0000_H$
SCU_IRCON4	Interrupt Request Register 4	014_H	$0000\ 0000_H$
SCU_IRCON4CLR	Interrupt Request Clear Register 4	198_H	$0000\ 0000_H$
SCU_IRCON5	Interrupt Request Register 5	$0F0_H$	$0000\ 0000_H$
SCU_IRCON5CLR	Interrupt Request Clear Register 5	$19C_H$	$0000\ 0000_H$
SCU_NMISR	NMI Status Register	018_H	$0000\ 0000_H$
SCU_NMISRCLR	NMI Status Clear Register	000_H	$0000\ 0000_H$
SCU_GPT12IRC	Timer and Counter Control/Status Register	160_H	$0000\ 0000_H$
SCU_GPT12ICLR	Timer and Counter Control/Status Clear Register	180_H	$0000\ 0000_H$

The registers are addressed wordwise.

7.6.3.1 Interrupt Node Enable Registers

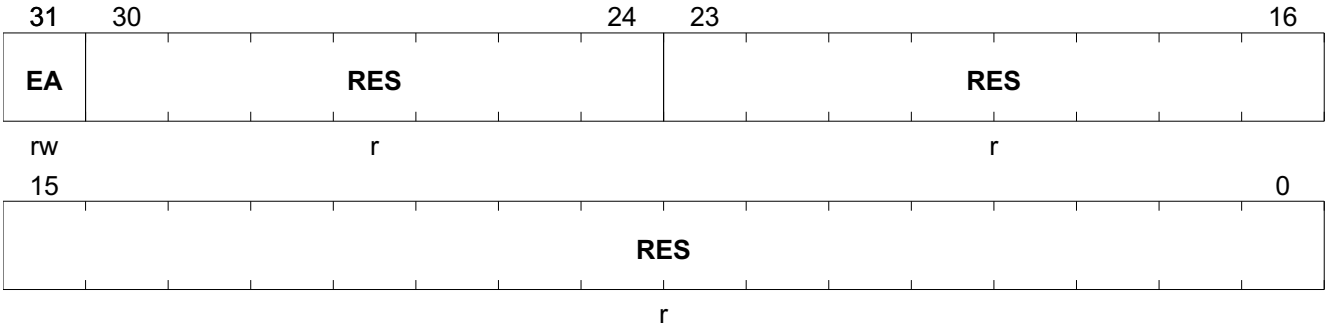
Register IEN0 contains the global interrupt masking bit (EA), which can be cleared to block all pending interrupt requests at once.

The NMI interrupt vector is shared by a number of sources, each of which can be enabled or disabled individually via register NMICON.

After reset, the enable bits in IEN0 and NMICON are cleared to 0. This implies that all interrupt nodes are disabled by default.

Interrupt Enable Register 0

SCU_IEN0	Offset	Reset Value
Interrupt Enable Register 0	$01C_H$	see Table 60



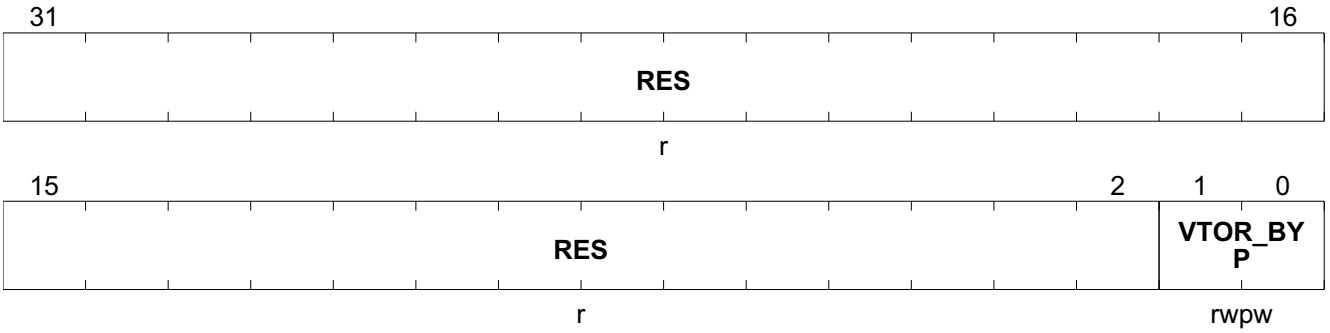
Field	Bits	Type	Description
EA	31	rw	**Global Interrupt Mask** 0_B **disable**, All pending interrupt requests (except NMI) are blocked from the core. 1_B **enable**, Pending interrupt requests are not blocked from the core.
RES	30:24	r	**Reserved** Returns 0 if read; should be written with 0.
RES	23:0	r	**Reserved** Returns 0 if read; should be written with 0.

Table 60 RESET of SCU_IEN0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		
VARIANT	00000000_H	VARIANT		

Vector Table Reallocation Register

SCU_VTOR	Offset	Reset Value
Vector Table Reallocation Register	020_H	see Table 61



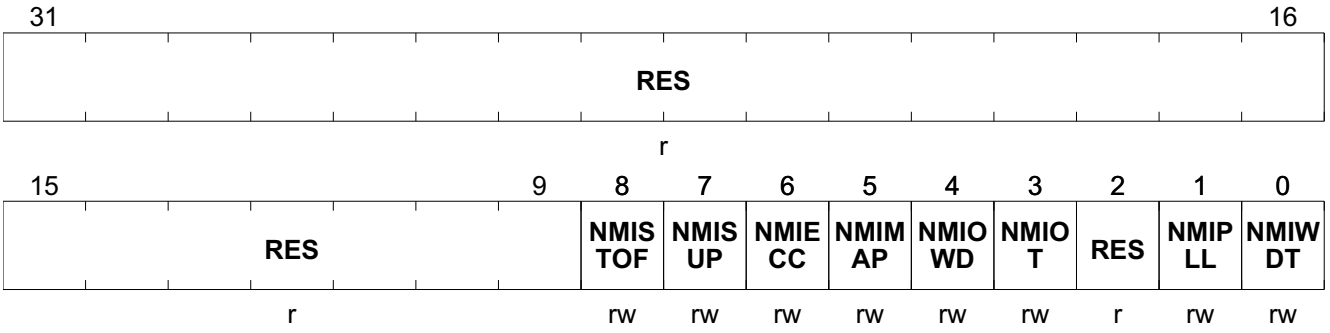
Field	Bits	Type	Description
RES	31:2	r	**Reserved** Returns 0 if read; should be written with 0.
VTOR_BYP	1:0	rwpw	**Vector Table Bypass Mode** 00_B **ROM**, VTOR is not remapped (ROM) 01_B **RAM**, VTOR is remapped to RAM 10_B **NVM_BSL**, VTOR is remapped to NVM 11_B **NVM_LIN**, VTOR is remapped to NVM

Table 61 RESET of SCU_VTOR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

NMI Control Register

SCU_NMICON	Offset	Reset Value
NMI Control Register	024_H	see Table 62



Field	Bits	Type	Description
RES	31:9	r	**Reserved** Returns 0 if read; should be written with 0.
NMISTOF	8	rw	**Stack Overflow NMI Enable** 0_B **disable**, Stack overflow NMI is disabled. 1_B **enable**, Stack overflow NMI is enabled.
NMISUP	7	rw	**Supply Prewarning NMI Enable** 0_B **disable**, Supply NMI is disabled. 1_B **enable**, Supply NMI is enabled.
NMIECC	6	rw	**ECC Error NMI Enable** 0_B **disable**, ECC Error NMI is disabled. 1_B **enable**, ECC Error NMI is enabled.
NMIMAP	5	rw	**NVM Map Error NMI Enable** 0_B **disable**, NVM Map Error NMI is disabled. 1_B **enable**, NVM Map Error NMI is enabled.
NMIOWD	4	rw	**Oscillator Watchdog NMI Enable** 0_B **disable**, Oscillator watchdog NMI is disabled. 1_B **enable**, Oscillator watchdog NMI is enabled.
NMIOT	3	rw	**NMI OT Enable** 0_B **disable**, NMI OT is disabled. 1_B **enable**, NMI OT is enabled.
RES	2	r	**Reserved** Returns 0 if read; should be written with 0.
NMIPLL	1	rw	**PLL Loss of Lock NMI Enable** 0_B **disable**, PLL Loss of Lock NMI is disabled. 1_B **enable**, PLL Loss of Lock NMI is enabled.
NMIWDT	0	rw	**Watchdog Timer NMI Enable** 0_B **disable**, WDT NMI is disabled. 1_B **enable**, WDT NMI is enabled.

Table 62 RESET of SCU_NMICON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

7.6.3.2 External Interrupt Control Registers

The external interrupts, EXT_INT[2:0], are driven into SCU from the ports. External interrupts can be positive, negative or double edge triggered. Register EXICON0 specifies the active edge for the external interrupt.

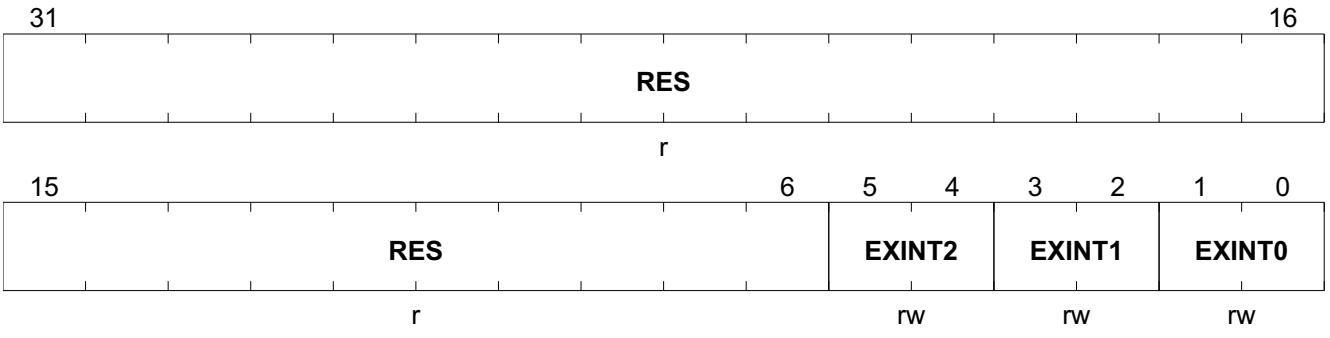
If the external interrupt is positive (negative) edge triggered, the external source must hold the request pin low (high) for at least one CCLK cycle, and then hold it high (low) for at least one CCLK cycle to ensure that the transition is recognized. If edge detection is bypassed for external interrupt 0 and external interrupt 1, the external source must hold the request pin “high” or “low” for at least two CCLK cycles.

External interrupt 2 share the interrupt node with other interrupt sources. Therefore in addition to the corresponding interrupt node enable, external interrupt 2 may be disabled individually, and is disabled by default after reset.

Note: Several external interrupts support alternative input pin, selected via MODPISEL register in the SCU. When switching inputs, the active edge/level trigger select and the level on the associated pins should be considered to prevent unintentional interrupt generation.

External Interrupt Control Register 0

SCU_EXICON0	Offset	Reset Value
External Interrupt Control Register 0	**028_H**	**see Table 63**



Field	Bits	Type	Description
RES	31:6	r	**Reserved** Returns 0 if read; should be written with 0.
EXINT2	5:4	rw	**External Interrupt 2 Trigger Select** 00_B **disable**, Interrupt disabled. 01_B **rising**, Interrupt on rising edge. 10_B **falling**, Interrupt on falling edge. 11_B **both**, Interrupt on both rising and falling edge.

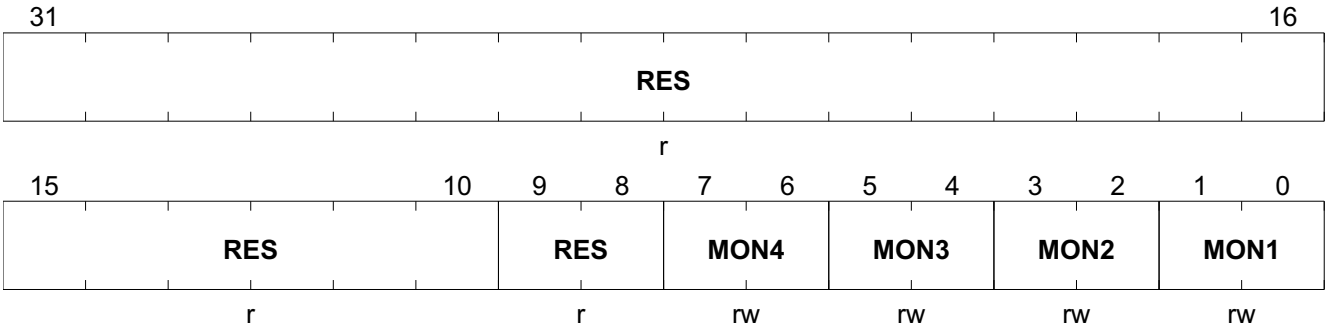
Field	Bits	Type	Description
EXINT1	3:2	rw	**External Interrupt 1 Trigger Select** 00_B **disable**, Interrupt disabled. 01_B **rising**, Interrupt on rising edge. 10_B **falling**, Interrupt on falling edge. 11_B **both**, Interrupt on both rising and falling edge.
EXINT0	1:0	rw	**External Interrupt 0 Trigger Select** 00_B **enable**, Interrupt disabled. 01_B **rising**, Interrupt on rising edge. 10_B **falling**, Interrupt on falling edge. 11_B **both**, Interrupt on both rising and falling edge.

Table 63 RESET of SCU_EXICON0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0030_H	RESET_TYPE_3		

External Interrupt Control Register 1

SCU_EXICON1	Offset	Reset Value
External Interrupt Control Register 1	**$02C_H$**	**see Table 64**



Field	Bits	Type	Description
RES	31:10	r	**Reserved** Returns 0 if read; should be written with 0.
RES	9:8	r	**Reserved** Returns 0 if read; should be written with 0.
MON4	7:6	rw	**MON4 Input Trigger Select** 00_B **disable**, external interrupt MON is disabled. 01_B **rising**, Interrupt on rising edge. 10_B **falling**, Interrupt on falling edge. 11_B **both**, Interrupt on both rising and falling edge.

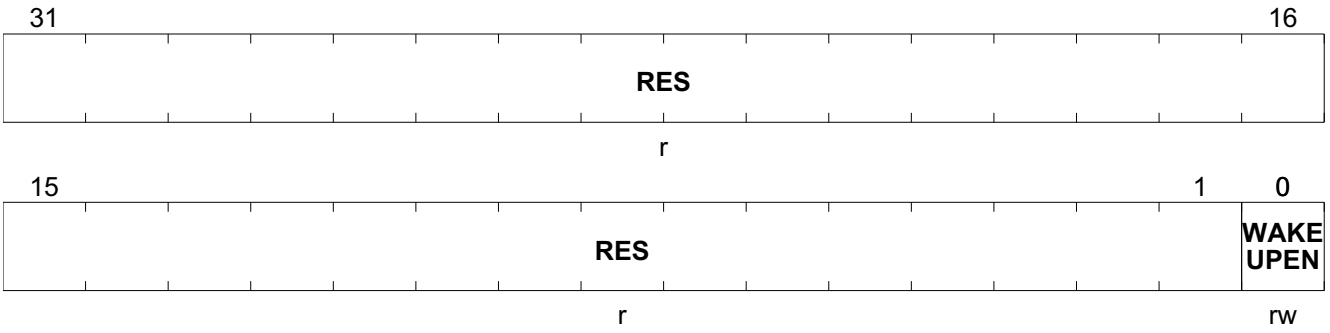
Field	Bits	Type	Description
MON3	5:4	rw	**MON3 Input Trigger Select** 00_B **disable**, external interrupt MON is disabled. 01_B **rising**, Interrupt on rising edge. 10_B **falling**, Interrupt on falling edge. 11_B **both**, Interrupt on both rising and falling edge.
MON2	3:2	rw	**MON2 Input Trigger Select** 00_B **disable**, external interrupt MON is disabled. 01_B **rising**, Interrupt on rising edge. 10_B **falling**, Interrupt on falling edge. 11_B **both**, Interrupt on both rising and falling edge.
MON1	1:0	rw	**MON1 Input Trigger Select** 00_B **disable**, external interrupt MON is disabled. 01_B **rising**, Interrupt on rising edge. 10_B **falling**, Interrupt on falling edge. 11_B **both**, Interrupt on both rising and falling edge.

Table 64 RESET of SCU_EXICON1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Wakeup Interrupt Control Register

SCU_WAKECON	Offset	Reset Value
Wakeup Interrupt Control Register	**$0EC_H$**	**see Table 65**



Field	Bits	Type	Description
RES	31:1	r	**Reserved** Returns 0 if read; should be written with 0.
WAKEUPEN	0	rw	**Wakeup Interrupt Enable** 0_B **disable**, wakeup interrupt is disabled. 1_B **enable**, wakeup interrupt is enabled.

Table 65 RESET of SCU_WAKECON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

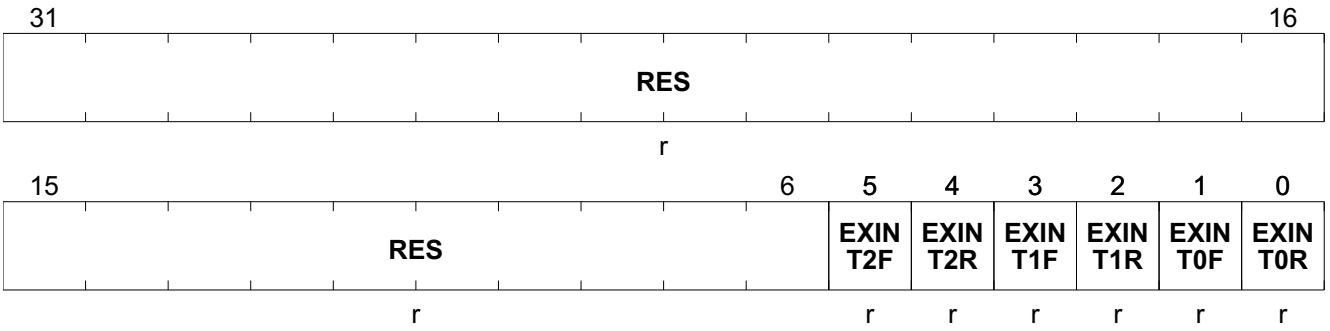
7.6.3.3 Interrupt Flag Registers

The interrupt flags for the different interrupt sources are located in several special function registers. This section describes the interrupt flags located in system registers or external interrupts belonging to system. Other interrupt flags located in respective module registers are described in the specific module chapter. For a complete listing of the interrupt flags and their assignment to SFRs, refer to **Table 172**.

In case of software and hardware access to a flag bit at the same time, hardware will have higher priority.

Interrupt Request Register 0

SCU_IRCON0	Offset	Reset Value
Interrupt Request Register 0	004_H	see Table 66



Field	Bits	Type	Description
RES	31:6	r	**Reserved** Returns 0 if read; should be written with 0.
EXINT2F	5	r	**Interrupt Flag for External Interrupt 2x on falling edge** This bit is set by hardware and can only be cleared by software. 0_B **Int**, Interrupt on falling edge event has not occurred. 1_B **no Int**, Interrupt on falling edge event has occurred.
EXINT2R	4	r	**Interrupt Flag for External Interrupt 2x on rising edge** This bit is set by hardware and can only be cleared by software. 0_B **Int**, Interrupt on rising edge event has not occurred. 1_B **no Int**, Interrupt on rising edge event has occurred.
EXINT1F	3	r	**Interrupt Flag for External Interrupt 1x on falling edge** This bit is set by hardware and can only be cleared by software. 0_B **Int**, Interrupt on falling edge event has not occurred. 1_B **no Int**, Interrupt on falling edge event has occurred.
EXINT1R	2	r	**Interrupt Flag for External Interrupt 1x on rising edge** This bit is set by hardware and can only be cleared by software. 0_B **Int**, Interrupt on rising edge event has not occurred. 1_B **no Int**, Interrupt on rising edge event has occurred.

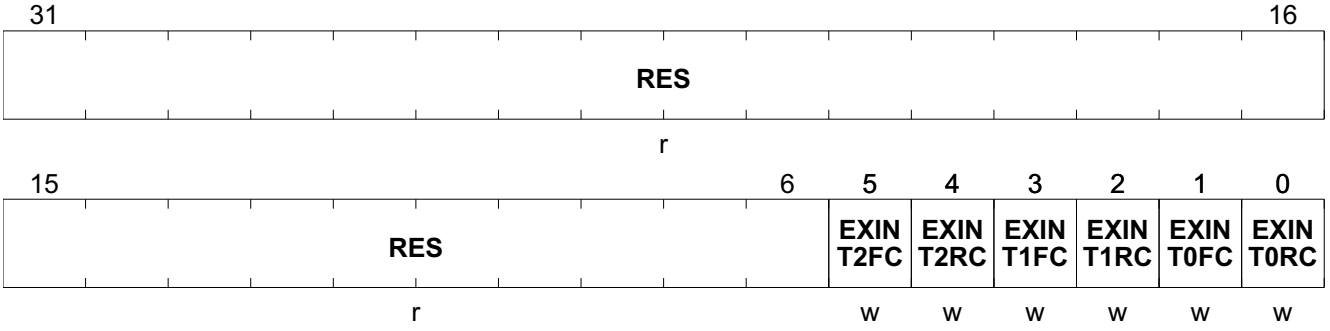
Field	Bits	Type	Description
EXINT0F	1	r	**Interrupt Flag for External Interrupt 0x on falling edge** This bit is set by hardware and can only be cleared by software. 0_B **Int**, Interrupt on falling edge event has not occurred. 1_B **no Int**, Interrupt on falling edge event has occurred.
EXINT0R	0	r	**Interrupt Flag for External Interrupt 0x on rising edge** This bit is set by hardware and can only be cleared by software. 0_B **Int**, Interrupt on rising edge event has not occurred. 1_B **no Int**, Interrupt on rising edge event has occurred.

Table 66 RESET of SCU_IRCON0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000_H	RESET_TYPE_3		

Interrupt Request Register 0 Clear

SCU_IRCON0CLR	Offset	Reset Value
Interrupt Request 0 Clear Register	178_H	see Table 67



Field	Bits	Type	Description
RES	31:6	r	**Reserved** Returns 0 if read; should be written with 0.
EXINT2FC	5	w	**Interrupt Flag for External Interrupt 2x on falling edge** 0_B **not cleared**, Interrupt event is not cleared. 1_B **cleared**, Interrupt event is cleared
EXINT2RC	4	w	**Interrupt Flag for External Interrupt 2x on rising edge** 0_B **not cleared**, Interrupt event is not cleared. 1_B **cleared**, Interrupt event is cleared
EXINT1FC	3	w	**Interrupt Flag for External Interrupt 1x on falling edge** 0_B **not cleared**, Interrupt event is not cleared. 1_B **cleared**, Interrupt event is cleared

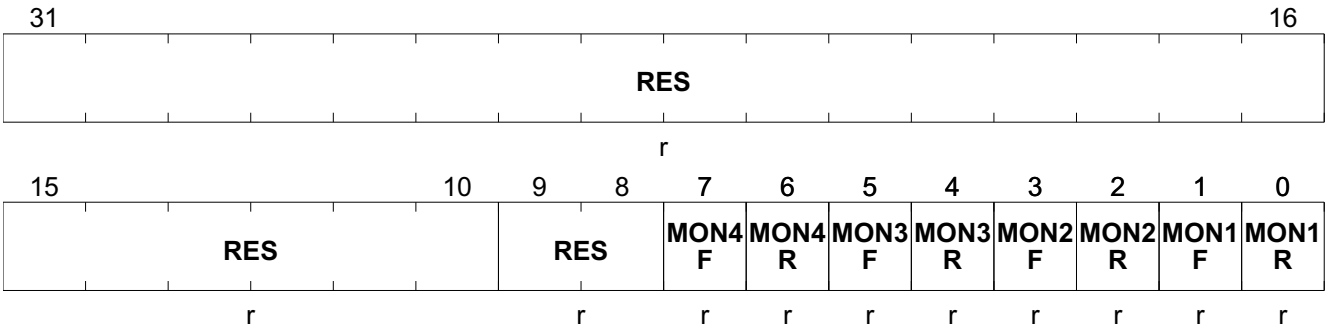
Field	Bits	Type	Description
EXINT1RC	2	w	**Interrupt Flag for External Interrupt 1x on rising edge** 0_B **not cleared**, Interrupt event is not cleared. 1_B **cleared**, Interrupt event is cleared
EXINT0FC	1	w	**Interrupt Flag for External Interrupt 0x on falling edge** 0_B **not cleared**, Interrupt event is not cleared. 1_B **cleared**, Interrupt event is cleared
EXINT0RC	0	w	**Interrupt Flag for External Interrupt 0x on rising edge** 0_B **not cleared**, Interrupt event is not cleared. 1_B **cleared**, Interrupt event is cleared

Table 67 RESET of SCU_IRCON0CLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000_H	RESET_TYPE_3		

Interrupt Request Register 1

SCU_IRCON1	Offset	Reset Value
Interrupt Request Register 1	008_H	see Table 68



Field	Bits	Type	Description
RES	31:10	r	**Reserved** Returns 0 if read; should be written with 0.
RES	9:8	r	**Reserved** Returns 0 if read; should be written with 0.
MON4F	7	r	**Interrupt Flag for MON4x on falling edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on falling edge event has not occurred. 1_B **Int**, Interrupt on falling edge event has occurred.

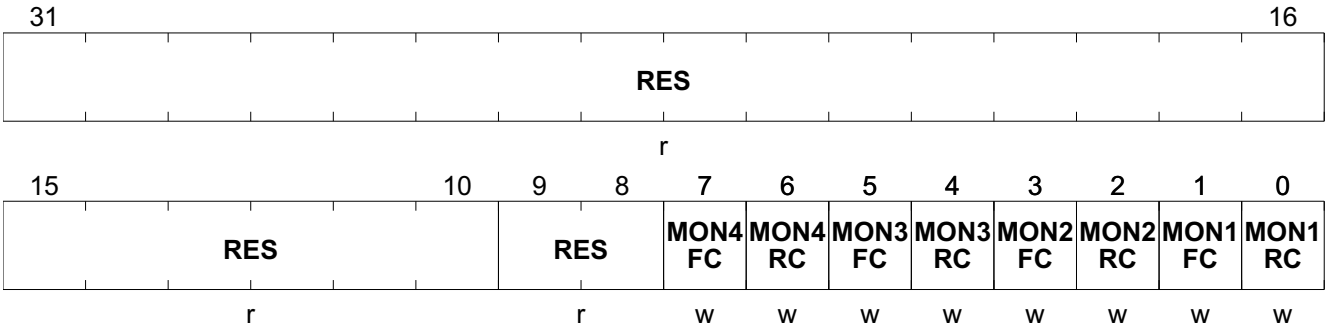
Field	Bits	Type	Description
MON4R	6	r	**Interrupt Flag for MON4x on rising edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on rising edge event has not occurred. 1_B **Int**, Interrupt on rising edge event has occurred.
MON3F	5	r	**Interrupt Flag for MON3x on falling edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on falling edge event has not occurred. 1_B **Int**, Interrupt on falling edge event has occurred.
MON3R	4	r	**Interrupt Flag for MON3x on rising edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on rising edge event has not occurred. 1_B **Int**, Interrupt on rising edge event has occurred.
MON2F	3	r	**Interrupt Flag for MON2x on falling edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on falling edge event has not occurred. 1_B **Int**, Interrupt on falling edge event has occurred.
MON2R	2	r	**Interrupt Flag for MON2x on rising edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on rising edge event has not occurred. 1_B **Int**, Interrupt on rising edge event has occurred.
MON1F	1	r	**Interrupt Flag for MON1x on falling edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on falling edge event has not occurred. 1_B **Int**, Interrupt on falling edge event has occurred.
MON1R	0	r	**Interrupt Flag for MON1x on rising edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on rising edge event has not occurred. 1_B **Int**, Interrupt on rising edge event has occurred.

Table 68 RESET of SCU_IRCON1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000_H	RESET_TYPE_3		

Interrupt Request Register 1 Clear

SCU_IRCON1CLR	Offset	Reset Value
Interrupt Request 1 Clear Register	$17C_H$	see Table 69



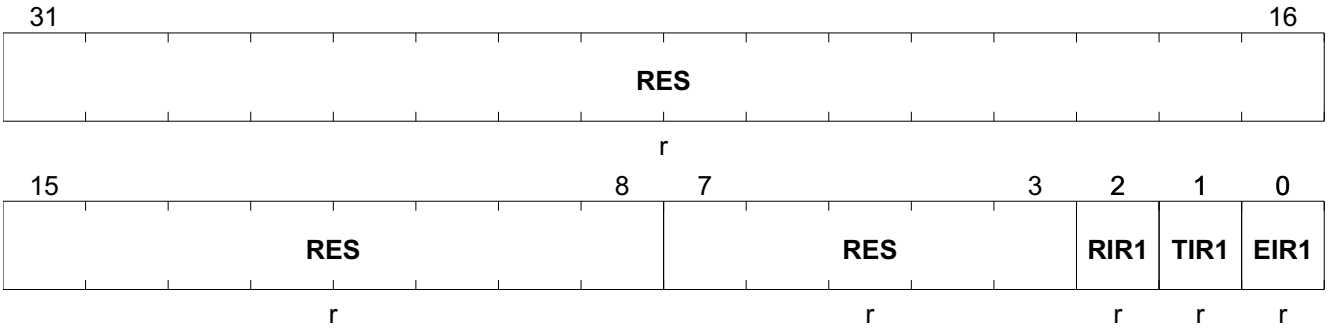
Field	Bits	Type	Description
RES	31:10	r	**Reserved** Returns 0 if read; should be written with 0.
RES	9:8	r	**Reserved** Returns 0 if read; should be written with 0.
MON4FC	7	w	**Interrupt Flag for MON4x on falling edge** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
MON4RC	6	w	**Interrupt Flag for MON4x on rising edge** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
MON3FC	5	w	**Interrupt Flag for MON3x on falling edge** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
MON3RC	4	w	**Interrupt Flag for MON3x on rising edge** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
MON2FC	3	w	**Interrupt Flag for MON2x on falling edge** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
MON2RC	2	w	**Interrupt Flag for MON2x on rising edge** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
MON1FC	1	w	**Interrupt Flag for MON1x on falling edge** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
MON1RC	0	w	**Interrupt Flag for MON1x on rising edge** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared

Table 69 RESET of SCU_IRCON1CLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Interrupt Request Register 2

SCU_IRCON2	Offset	Reset Value
Interrupt Request Register 2	$00C_H$	see Table 70



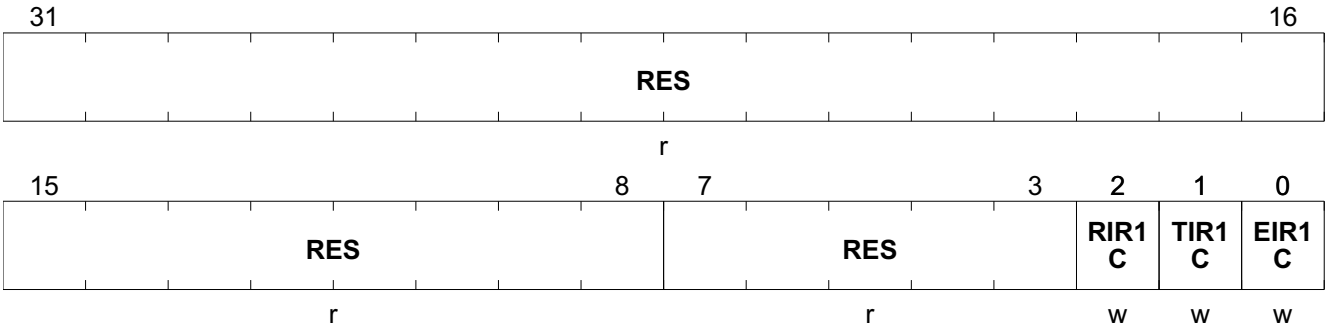
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
RES	7:3	r	**Reserved** Returns 0 if read; should be written with 0.
RIR1	2	r	**Receive Interrupt Flag for SSC1** This bit is set by hardware and can only be cleared by software. 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
TIR1	1	r	**Transmit Interrupt Flag for SSC1** This bit is set by hardware and can only be cleared by software. 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
EIR1	0	r	**Error Interrupt Flag for SSC1** This bit is set by hardware and can only be cleared by software. 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared

Table 70 RESET of SCU_IRCON2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Interrupt Request Register 2 Clear

SCU_IRCON2CLR	Offset	Reset Value
Interrupt Request 2 Clear Register	190_H	see Table 71



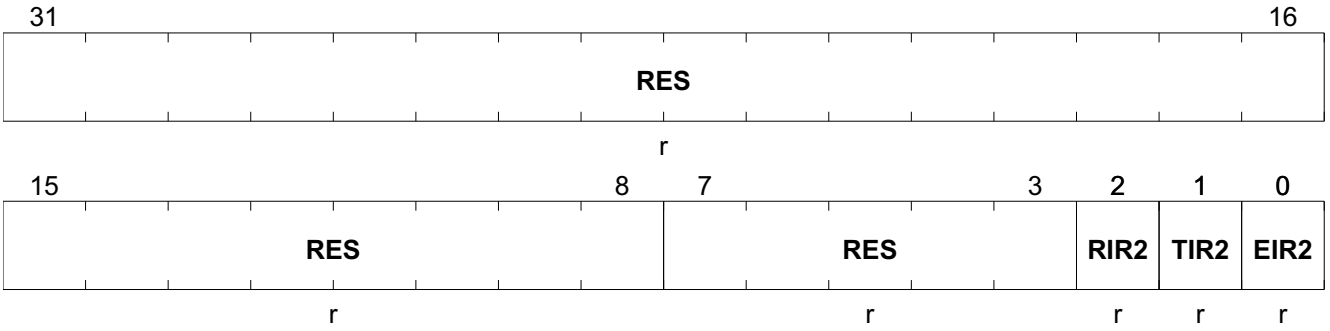
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
RES	7:3	r	**Reserved** Returns 0 if read; should be written with 0.
RIR1C	2	w	**Receive Interrupt Flag for SSC1** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
TIR1C	1	w	**Transmit Interrupt Flag for SSC1** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
EIR1C	0	w	**Error Interrupt Flag for SSC1** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared

Table 71 RESET of SCU_IRCON2CLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Interrupt Request Register 3

SCU_IRCON3	Offset	Reset Value
Interrupt Request Register 3	010_H	see Table 72



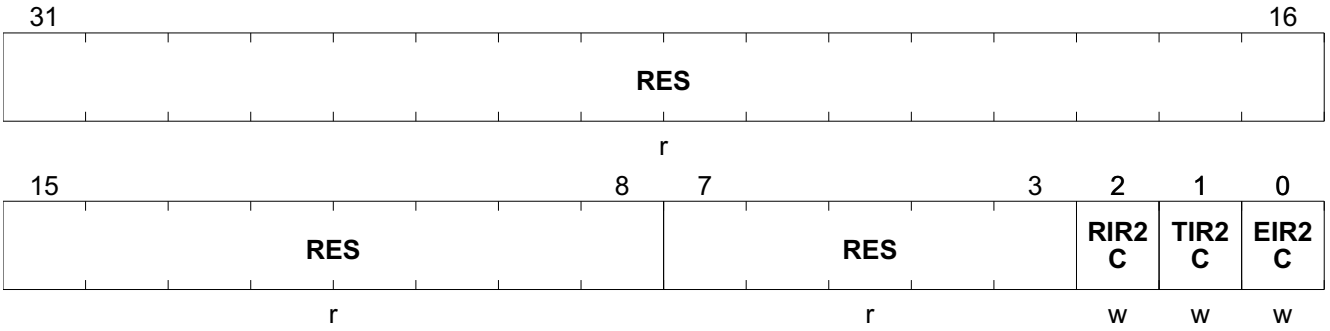
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
RES	7:3	r	**Reserved** Returns 0 if read; should be written with 0.
RIR2	2	r	**Receive Interrupt Flag for SSC2** This bit is set by hardware and can only be cleared by software. 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
TIR2	1	r	**Transmit Interrupt Flag for SSC2** This bit is set by hardware and can only be cleared by software. 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
EIR2	0	r	**Error Interrupt Flag for SSC2** This bit is set by hardware and can only be cleared by software. 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared

Table 72 RESET of SCU_IRCON3

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Interrupt Request Register 3 Clear

SCU_IRCON3CLR	Offset	Reset Value
Interrupt Request 3 Clear Register	194_H	see Table 73



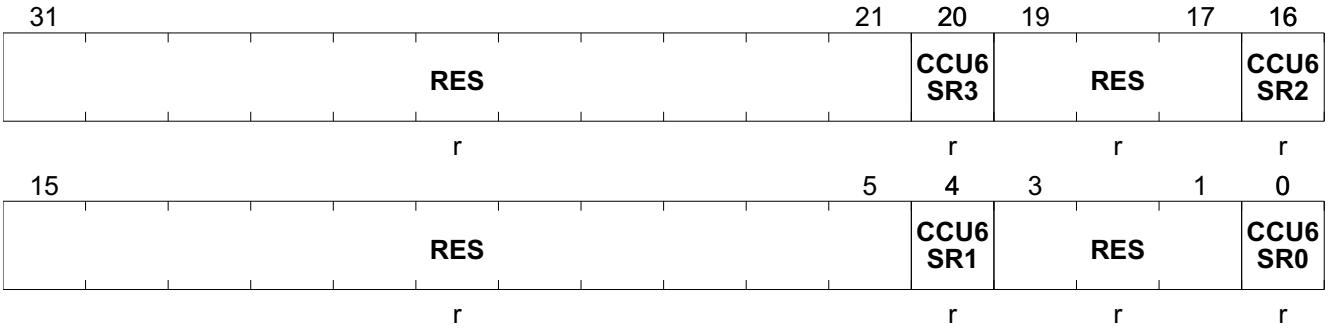
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
RES	7:3	r	**Reserved** Returns 0 if read; should be written with 0.
RIR2C	2	w	**Receive Interrupt Flag for SSC2** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
TIR2C	1	w	**Transmit Interrupt Flag for SSC2** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
EIR2C	0	w	**Error Interrupt Flag for SSC2** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared

Table 73 RESET of SCU_IRCON3CLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Interrupt Request Register 4

SCU_IRCON4	Offset	Reset Value
Interrupt Request Register 4	014_H	see Table 74



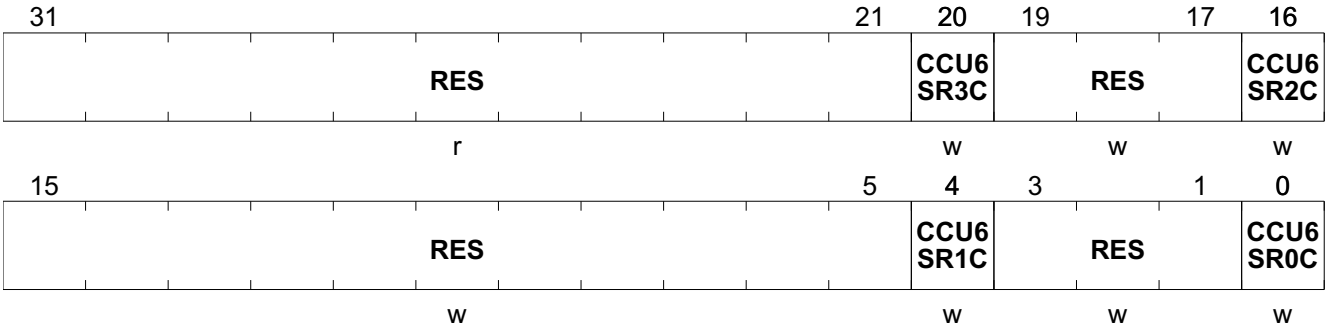
Field	Bits	Type	Description
RES	31:21	r	**Reserved** Returns 0 if read; should be written with 0.
CCU6SR3	20	r	**Interrupt Flag 1 for CCU6** This bit is set by hardware and can only be cleared by software. 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
RES	19:17	r	**Reserved** Returns 0 if read; should be written with 0.
CCU6SR2	16	r	**Interrupt Flag 1 for CCU6** This bit is set by hardware and can only be cleared by software. 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
RES	15:5	r	**Reserved** Returns 0 if read; should be written with 0.
CCU6SR1	4	r	**Interrupt Flag 1 for CCU6** This bit is set by hardware and can only be cleared by software. 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
RES	3:1	r	**Reserved** Returns 0 if read; should be written with 0.
CCU6SR0	0	r	**Interrupt Flag 1 for CCU6** This bit is set by hardware and can only be cleared by software. 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared

Table 74 RESET of SCU_IRCON4

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Interrupt Request Register 4 Clear

SCU_IRCON4CLR	Offset	Reset Value
Interrupt Request 4 Clear Register	198_H	see Table 75



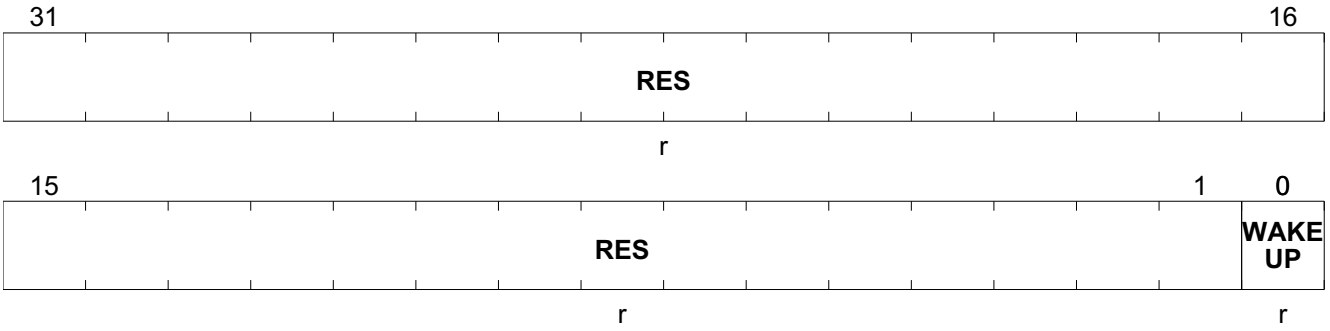
Field	Bits	Type	Description
RES	31:21	r	**Reserved** Returns 0 if read; should be written with 0.
CCU6SR3C	20	w	**Interrupt Flag 1 for CCU6** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
RES	19:17	w	**Reserved** Returns 0 if read; should be written with 0.
CCU6SR2C	16	w	**Interrupt Flag 1 for CCU6** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
RES	15:5	w	**Reserved** Returns 0 if read; should be written with 0.
CCU6SR1C	4	w	**Interrupt Flag 1 for CCU6** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
RES	3:1	w	**Reserved** Returns 0 if read; should be written with 0.
CCU6SR0C	0	w	**Interrupt Flag 1 for CCU6** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared

Table 75 RESET of SCU_IRCON4CLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Interrupt Request Register 5

SCU_IRCON5	Offset	Reset Value
Interrupt Request Register 5	$0F0_H$	see Table 76



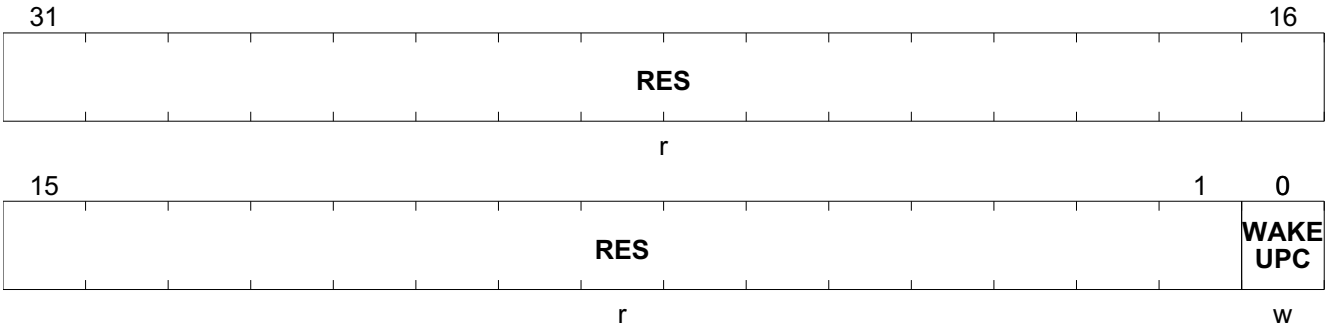
Field	Bits	Type	Description
RES	31:1	r	**Reserved** Returns 0 if read; should be written with 0.
WAKEUP	0	r	**Interrupt Flag for Wakeup** This bit is set by hardware and can only be cleared by software. 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared

Table 76 RESET of SCU_IRCON5

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Interrupt Request Register 5 Clear

SCU_IRCON5CLR	Offset	Reset Value
Interrupt Request 5 Clear Register	$19C_H$	see Table 77



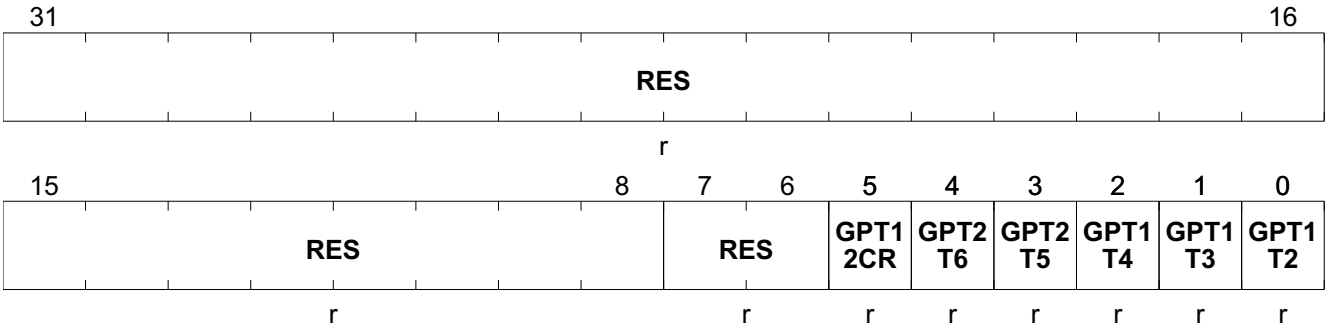
Field	Bits	Type	Description
RES	31:1	r	**Reserved** Returns 0 if read; should be written with 0.
WAKEUPC	0	w	**Clear Flag for Wakeup Interrupt** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared

Table 77 RESET of SCU_IRCON5CLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Timer and Counter Control/Status Register

SCU_GPT12IRC	Offset	Reset Value
Timer and Counter Control/Status Register	160_H	see Table 78



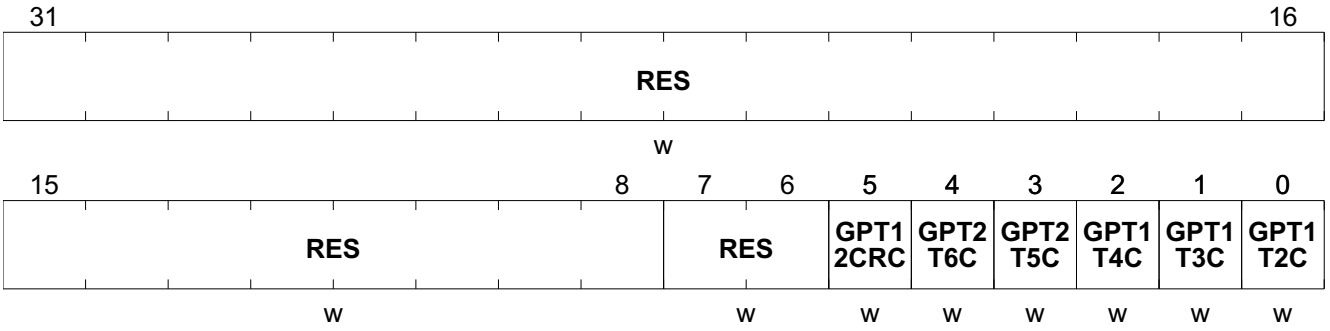
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
RES	7:6	r	**Reserved** Returns 0 if read; should be written with 0.
GPT12CR	5	r	**GPT Module 1 Capture Reload Interrupt Status** Capture Reload Event of GPT1 Module Interrupt Status 0_B **No Int**, No Capture Reload Interrupt has occurred. 1_B **Int**, Capture Reload Interrupt has occurred.
GPT2T6	4	r	**GPT Module 2Timer6 Interrupt Status** Timer 6 of GPT Module Interrupt Status 0_B **No Int**, No Timer 6 Interrupt has occurred. 1_B **Int**, Timer 6 Interrupt has occurred.
GPT2T5	3	r	**GPT Module 2 Timer5 Interrupt Status** Timer 5 of GPT2 Module Interrupt Status 0_B **No Int**, No Timer 5 Interrupt has occurred. 1_B **Int**, Timer 5 Interrupt has occurred.
GPT1T4	2	r	**GPT Module 1 Timer4 Interrupt Status** Timer 4 of GPT1 Module Interrupt Status 0_B **No Int**, No Timer 4 Interrupt has occurred. 1_B **Int**, Timer 4 Interrupt has occurred.
GPT1T3	1	r	**GPT Module 1 Timer3 Interrupt Status** Timer 3 of GPT1 Module Interrupt Status 0_B **No Int**, No Timer 3 Interrupt has occurred. 1_B **Int**, Timer 3 Interrupt has occurred.
GPT1T2	0	r	**GPT Module 1 Timer 2 Interrupt Status** Timer 2 of GPT1 Module Interrupt Status 0_B **No Int**, No Timer 2 Interrupt has occurred. 1_B **Int**, Timer 2 Interrupt has occurred.

Table 78 RESET of SCU_GPT12IRC

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Timer and Counter Control/Status Register

SCU_GPT12ICLR	Offset	Reset Value
Timer and Counter Control/Status Clear Register	180_H	see Table 79



Field	Bits	Type	Description
RES	31:8	w	**Reserved** Returns 0 if read; should be written with 0.
RES	7:6	w	**Reserved** Returns 0 if read; should be written with 0.
GPT12CRC	5	w	**GPT Module 1 Capture Reload Interrupt Status** Capture Reload Event of GPT1 Module Interrupt Status 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
GPT2T6C	4	w	**GPT Module 2Timer6 Interrupt Status** Timer 6 of GPT Module Interrupt Status 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
GPT2T5C	3	w	**GPT Module 2 Timer5 Interrupt Status** Timer 5 of GPT2 Module Interrupt Status 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
GPT1T4C	2	w	**GPT Module 1 Timer4 Interrupt Status** Timer 4 of GPT1 Module Interrupt Status 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
GPT1T3C	1	w	**GPT Module 1 Timer3 Interrupt Status** Timer 3 of GPT1 Module Interrupt Status 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
GPT1T2C	0	w	**GPT Module 1 Timer 2 Interrupt Status** Timer 2 of GPT1 Module Interrupt Status 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared

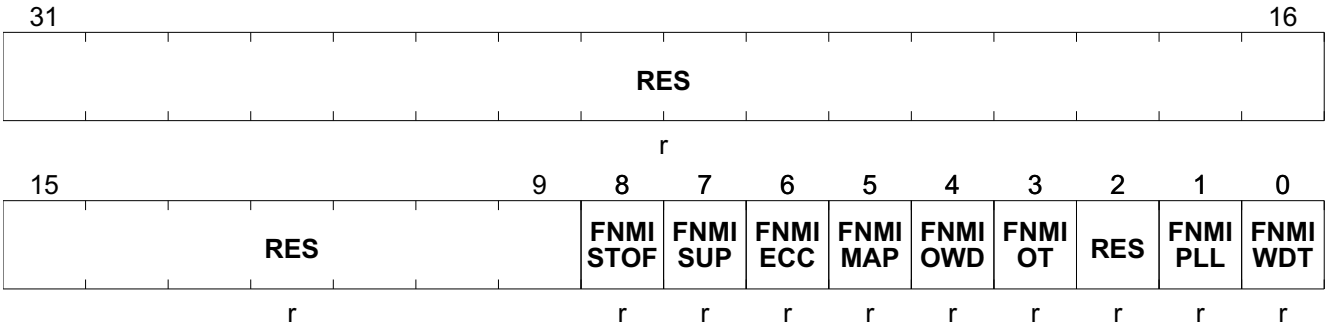
Table 79 RESET of SCU_GPT12ICLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

NMI Status Register

Each NMI event and status flag is retained across these resets: 1) WDT reset, 2) soft reset. These include all the flags of NMISR register: FNMIWDT, FNMIPLL, FNMIOT, FNMIOWD, FNMIMAP, and indirectly, FNMIECC and FNMISUP. In the case of NMIs with shared source i.e. watchdog, ECC or supply prewarning NMI, the respective indicator or event flags not located in NMISR are also retained. Refer to Chapter 1.6.5 for identifying the NMI event.

SCU_NMISR	Offset	Reset Value
NMI Status Register	018_H	see Table 80



Field	Bits	Type	Description
RES	31:9	r	**Reserved** Returns 0 if read; should be written with 0.
FNMISTOF	8	r	**Stack Overflow NMI Flag** This flag is cleared automatically by hardware when the corresponding event flags are cleared. 0_B **no Int**, No supply prewarning NMI has occurred. 1_B **Int**, Supply prewarning has occurred.
FNMISUP	7	r	**Supply Prewarning NMI Flag** This flag is cleared automatically by hardware when the corresponding event flags are cleared. 0_B **no Int**, No supply prewarning NMI has occurred. 1_B **Int**, Supply prewarning has occurred.
FNMIECC	6	r	**ECC Error NMI Flag** This flag is cleared automatically by hardware when the corresponding enabled event flags are cleared. 0_B **no Int**, No uncorrectable ECC error has occurred on NVM, XRAM. 1_B **Int**, Uncorrectable ECC error has occurred on NVM, RAM.
FNMIMAP	5	r	**NVM Map Error NMI Flag** This bit is set by hardware and can only be cleared by software. 0_B **no Int**, No NVM Map Error NMI has occurred. 1_B **Int**, NVM Map Error has occurred.

Field	Bits	Type	Description
FNMIOWD	4	r	**Oscillator Watchdog NMI Flag** This bit is set by hardware and can only be cleared by software. 0_B **no Int**, No oscillator watchdog NMI has occurred. 1_B **Int**, Oscillator watchdog event has occurred.
FNMIOT	3	r	**Overtemperature NMI Flag** This bit is set by hardware and can only be cleared by software. As this is a shared NMI source, this flag should be cleared after checking and clearing the corresponding event flags. 0_B **no Int**, No OT NMI has occurred. 1_B **Int**, OT NMI event has occurred.
RES	2	r	**Reserved** Returns 0 if read; should be written with 0.
FNMIPLL	1	r	**PLL NMI Flag** This bit is set by hardware and can only be cleared by software. 0_B **no Int**, No PLL NMI has occurred. 1_B **Int**, PLL loss-of-lock to the external crystal has occurred.
FNMIWDT	0	r	**Watchdog Timer NMI Flag** This bit is set by hardware and can only be cleared by software. As this is a shared NMI source, this flag should be cleared after checking and clearing the corresponding event flags. 0_B **no Int**, No watchdog NMI has occurred. 1_B **Int**, WDT prewarning has occurred.

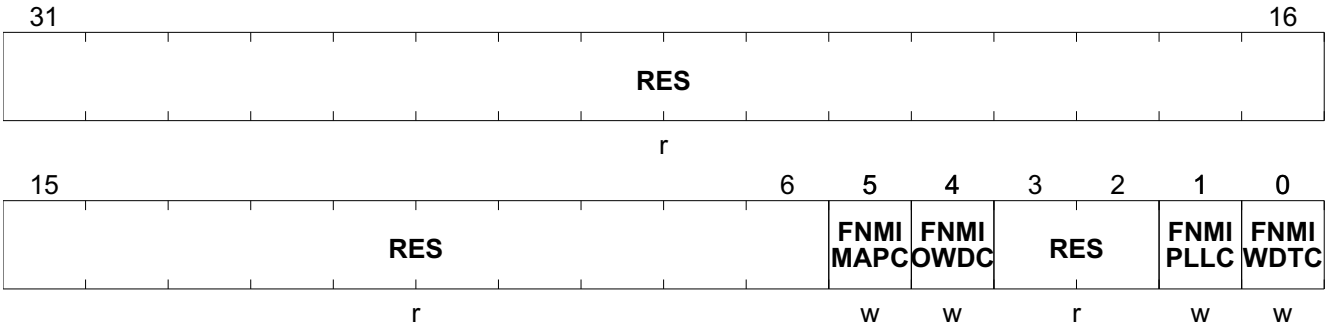
Table 80 RESET of SCU_NMISR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		

NMI Status Register

Each NMI event and status flag is retained across these resets: 1) WDT reset, 2) soft reset. These include all the flags of NMISR register: FNMIWDT, FNMIPLL, FNMIOT, FNMIOWD, FNMIMAP, and indirectly, FNMIECC and FNMISUP. In the case of NMIs with shared source i.e. watchdog, ECC or supply prewarning NMI, the respective indicator or event flags not located in NMISR are also retained. Refer to Chapter 1.6.5 for identifying the NMI event.

SCU_NMISRCLR	Offset	Reset Value
NMI Status Clear Register	000_H	see Table 81



Field	Bits	Type	Description
RES	31:6	r	**Reserved** Returns 0 if read; should be written with 0.
FNMIMAPC	5	w	**NVM Map Error NMI Flag** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
FNMIOWDC	4	w	**Oscillator Watchdog NMI Flag** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
RES	3:2	r	**Reserved** Returns 0 if read; should be written with 0.
FNMIPLLC	1	w	**PLL NMI Flag** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
FNMIWDTC	0	w	**Watchdog Timer NMI Flag** As this is a shared NMI source, this flag should be cleared after checking and clearing the corresponding event flags. 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared

Table 81 RESET of SCU_NMISRCLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000_H	RESET_TYPE_3		

7.6.4 Interrupt Related Registers

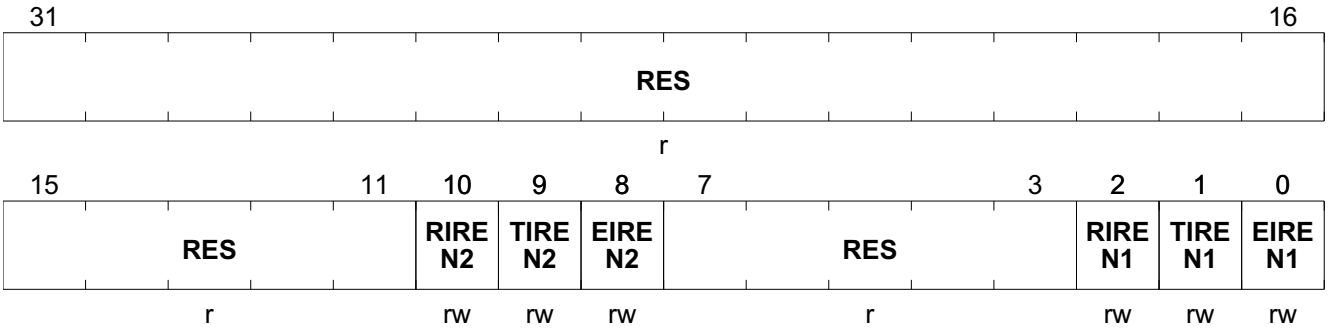
Several interrupt related registers are located in the SCU.

7.6.4.1 Interrupt Event Enable Control

The following registers collect the interrupt enable bits for all interrupt events which do not have enable bits on their respective module level.

Peripheral Interrupt Enable Register 1

SCU_MODIEN1	Offset	Reset Value
Peripheral Interrupt Enable Register 1	**030_H**	**see Table 82**



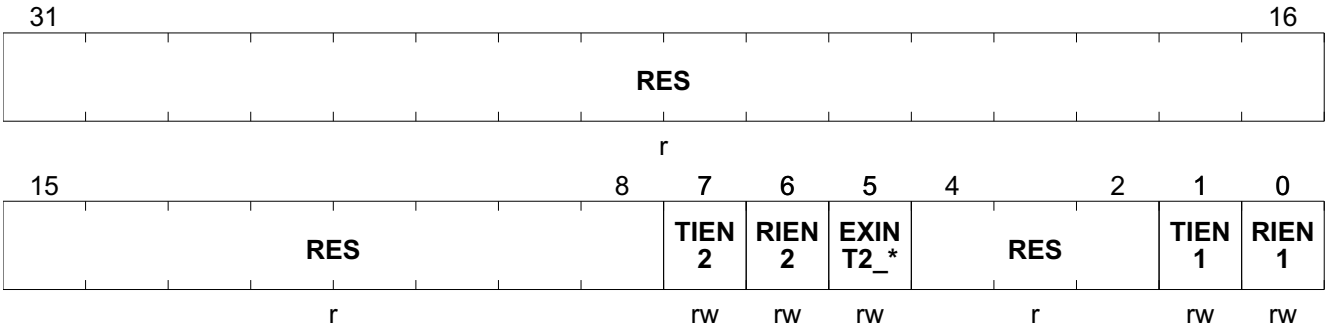
Field	Bits	Type	Description
RES	31:11	r	**Reserved** Returns 0 if read; should be written with 0.
RIREN2	10	rw	**SSC 2 Receive Interrupt Enable** 0_B **Disable**, Receive interrupt is disabled 1_B **Enable**, Receive interrupt is enabled
TIREN2	9	rw	**SSC 2 Transmit Interrupt Enable** 0_B **Disable**, Transmit interrupt is disabled 1_B **Enable**, Transmit interrupt is enabled
EIREN2	8	rw	**SSC 2 Error Interrupt Enable** 0_B **Disable**, Error interrupt is disabled 1_B **Enable**, Error interrupt is enabled
RES	7:3	r	**Reserved** Returns 0 if read; should be written with 0.
RIREN1	2	rw	**SSC 1 Receive Interrupt Enable** 0_B **Disable**, Receive interrupt is disabled 1_B **Enable**, Receive interrupt is enabled
TIREN1	1	rw	**SSC 1 Transmit Interrupt Enable** 0_B **Disable**, Transmit interrupt is disabled 1_B **Enable**, Transmit interrupt is enabled
EIREN1	0	rw	**SSC 1 Error Interrupt Enable** 0_B **Disable**, Error interrupt is disabled 1_B **Enable**, Error interrupt is enabled

Table 82 RESET of SCU_MODIEN1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Peripheral Interrupt Enable Register 2

SCU_MODIEN2	Offset	Reset Value
Peripheral Interrupt Enable Register 2	034_H	see Table 83



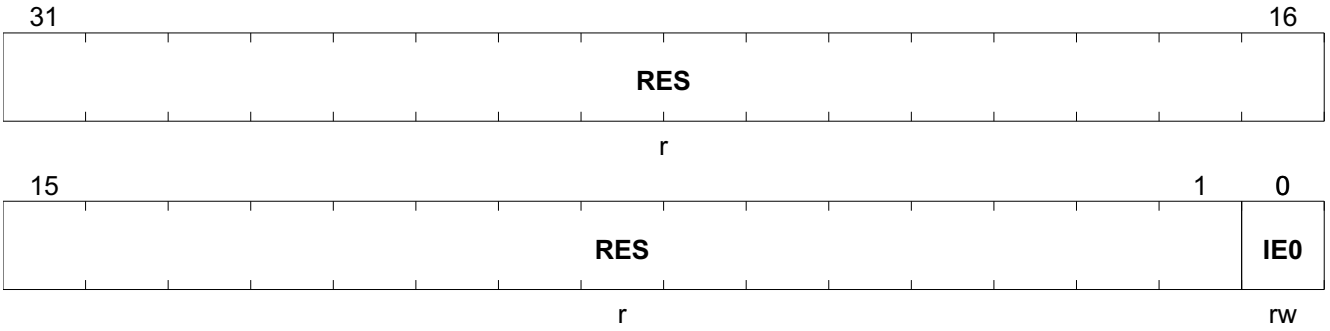
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
TIEN2	7	rw	**UART 2 Transmit Interrupt Enable** 0_B **Disable**, Transmit interrupt is disabled 1_B **Enable**, Transmit interrupt is enabled
RIEN2	6	rw	**UART 2 Receive Interrupt Enable** 0_B **Disable**, Receive interrupt is disabled 1_B **Enable**, Receive interrupt is enabled
EXINT2_EN	5	rw	**External Interrupt 2 Enable** 0_B **Disable**, External interrupt is disabled 1_B **Enable**, External interrupt is enabled
RES	4:2	r	**Reserved** Returns 0 if read; should be written with 0.
TIEN1	1	rw	**UART 1 Transmit Interrupt Enable** 0_B **Disable**, Transmit interrupt is disabled 1_B **Enable**, Transmit interrupt is enabled
RIEN1	0	rw	**UART 1 Receive Interrupt Enable** 0_B **Disable**, Receive interrupt is disabled 1_B **Enable**, Receive interrupt is enabled

Table 83 RESET of SCU_MODIEN2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Peripheral Interrupt Enable Register 3

SCU_MODIEN3	Offset	Reset Value
Peripheral Interrupt Enable Register 3	038_H	see Table 84



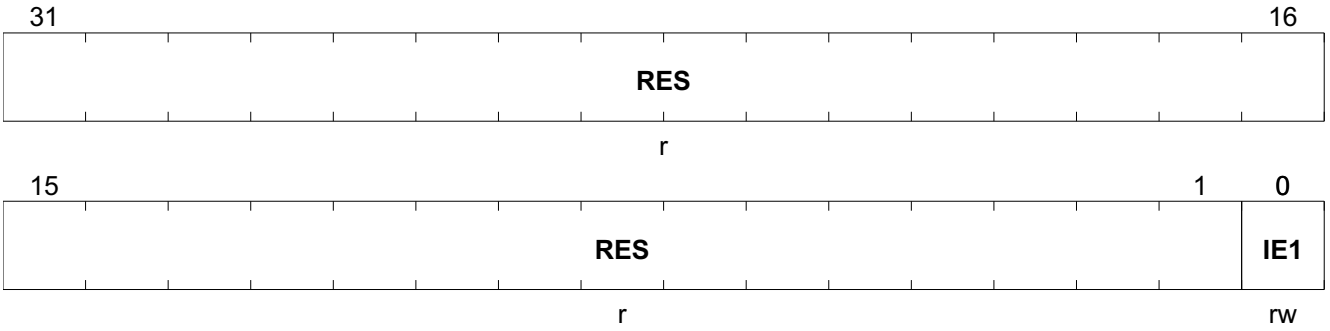
Field	Bits	Type	Description
RES	31:1	r	**Reserved** Returns 0 if read; should be written with 0.
IE0	0	rw	**External Interrupt Enable** 0_B **Disable**, disabled 1_B **Enable**, enabled

Table 84 RESET of SCU_MODIEN3

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Peripheral Interrupt Enable Register 4

SCU_MODIEN4	Offset	Reset Value
Peripheral Interrupt Enable Register 4	$03C_H$	see Table 85



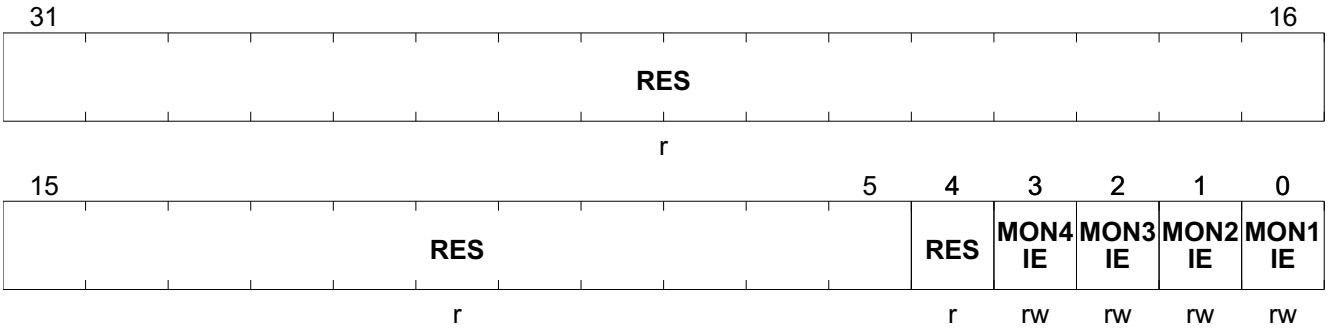
Field	Bits	Type	Description
RES	31:1	r	**Reserved** Returns 0 if read; should be written with 0.
IE1	0	rw	**External Interrupt Enable** 0_B **Disable**, disabled 1_B **Enable**, enabled

Table 85 RESET of SCU_MODIEN4

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Monitoring Input Interrupt Enable Register

SCU_MONIEN	Offset	Reset Value
Monitoring Input Interrupt Enable Register	$18C_H$	see Table 86



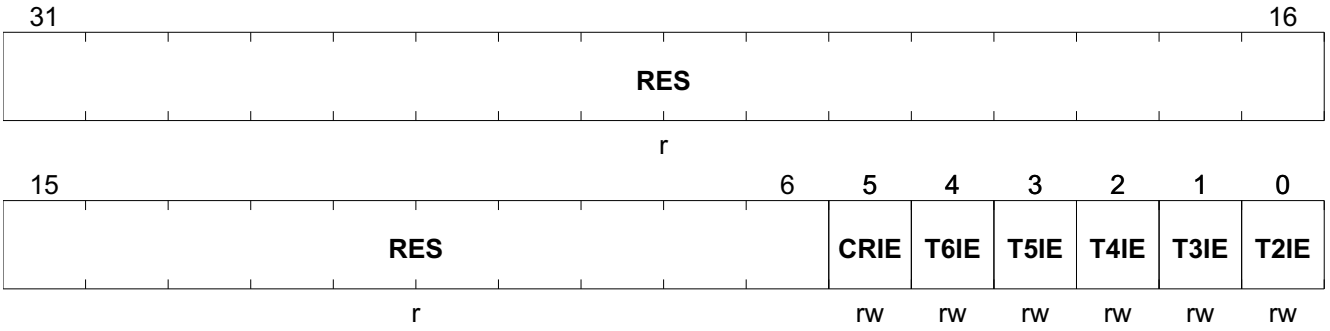
Field	Bits	Type	Description
RES	31:5	r	**Reserved** Returns 0 if read; should be written with 0.
RES	4	r	**Reserved** Returns 0 if read; should be written with 0.
MON4IE	3	rw	**MON 4 Interrupt Enable** 0_B **Disable**, disabled 1_B **Enable**, enabled
MON3IE	2	rw	**MON 3 Interrupt Enable** 0_B **Disable**, disabled 1_B **Enable**, enabled
MON2IE	1	rw	**MON 2 Interrupt Enable** 0_B **Disable**, disabled 1_B **Enable**, enabled
MON1IE	0	rw	**MON 1 Interrupt Enable** 0_B **Disable**, disabled 1_B **Enable**, enabled

Table 86 RESET of SCU_MONIEN

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

General Purpose Timer 12 Interrupt Enable Register

SCU_GPT12IEN	Offset	Reset Value
General Purpose Timer 12 Interrupt Enable Register	$15C_H$	see Table 87



Field	Bits	Type	Description
RES	31:6	r	**Reserved** Returns 0 if read; should be written with 0.
CRIE	5	rw	**General Purpose Timer 12 Capture and Reload Interrupt Enable** 0_B **Disable**, disabled 1_B **Enable**, enabled
T6IE	4	rw	**General Purpose Timer 12 T6 Interrupt Enable** 0_B **Disable**, disabled 1_B **Enable**, enabled
T5IE	3	rw	**General Purpose Timer 12 T5 Interrupt Enable** 0_B **Disable**, disabled 1_B **Enable**, enabled
T4IE	2	rw	**General Purpose Timer 12 T4 Interrupt Enable** 0_B **Disable**, disabled 1_B **Enable**, enabled
T3IE	1	rw	**General Purpose Timer 12 T3 Interrupt Enable** 0_B **Disable**, disabled 1_B **Enable**, enabled
T2IE	0	rw	**General Purpose Timer 12 T2 Interrupt Enable** 0_B **Disable**, disabled 1_B **Enable**, enabled

Table 87 RESET of SCU_GPT12IEN

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Other Interrupt Related Registers

The following interrupt related registers are located in the SCU:

- NMICON
- NMISR
- IRCON0, IRCON1, IRCON3, IRCON4
- EXICON0
- MODIEN1, MODIEN2

All registers, except MODIENx, are described in the Interrupt System **Chapter 12.9**.

7.6.5 NMI Event Flags Handling

Each NMI event and status flag is retained across these resets: 1) WDT reset, 2) soft reset. Specifically, these include all the flags of NMISR register: FNMIWDT, FNMIPLL, FNMIOT, FNMIOWD, FNMIMAP and indirectly, FNMIECC and FNMISUP. In the case of watchdog resets, the requestor can be identified via the reset indicator bits WDT1RST and WDTRST. The ECC NMI is indicated by the respective event flags of SFR EDCSTAT.IRDBE, XRDBE and NVMDBE. Likewise, the supply prewarning NMI and MI_CLK WDT NMI is indicated by the respective event flags located in **SCU_PM** chapter.These NMI event and status flags are otherwise reset to default value with all other resets i.e. power-on, brown-out, hardware, WDT1 (except WDT1RST) and wakeup reset.

7.7 General Port Control

The SCU contains control registers for the selection of:

- alternate input functions of UART, Timers and External Interrupts (**Section 7.7.2**)
- port output driver strength and temperature compensation (**Section 7.7.3**)

For functional description of GPIO ports, refer to **Chapter 15**.

7.7.1 Timer 2/Timer 21 Input Selection Configuration

Table 88 Timer2 Input Signal Selection

MODPISEL1. T2EXCON Bit Field	MODPISEL2. T2EXISCNF Bit Field	MODPISEL2. T2EXIS Bit Field	Description
0_B	00_B	00_B	T2EX_0
0_B	00_B	01_B	T2EX_1
0_B	00_B	10_B	T2EX_2
0_B	00_B	11_B	T2EX_3
0_B	01_B	00_B	MON1
0_B	01_B	01_B	MON2
0_B	01_B	10_B	MON3
0_B	01_B	11_B	MON4
0_B	10_B	00_B	lin_rxd_i
0_B	10_B	01_B	reserved
0_B	10_B	10_B	cc6_cout60
0_B	10_B	11_B	cc6_cout61
0_B	11_B	00_B	cc6_ch0
0_B	11_B	01_B	cc6_ch1
0_B	11_B	10_B	cc6_ch2
0_B	11_B	11_B	cc6_ch2

Table 89 Timer21 Input Signal Selection

MODPISEL1. T21EXCON Bit Field	MODPISEL2. T21EXISCNF Bit Field	MODPISEL2. T21EXIS Bit Field	Description
0_B	00_B	00_B	T21EX_0
0_B	00_B	01_B	T21EX_1
0_B	00_B	10_B	T21EX_2
0_B	00_B	11_B	T21EX_3
0_B	01_B	00_B	MON1
0_B	01_B	01_B	MON2
0_B	01_B	10_B	MON3

Table 89 Timer21 Input Signal Selection

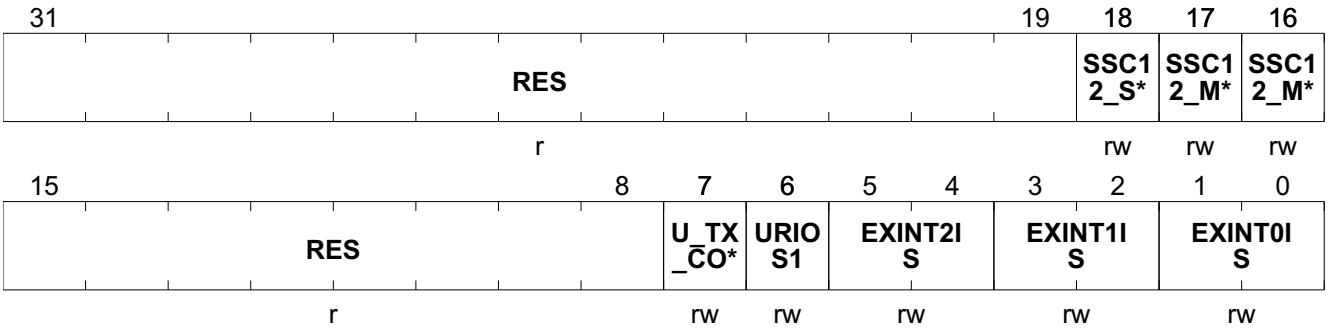
MODPISEL1. T21EXCON Bit Field	MODPISEL2. T21EXISCNF Bit Field	MODPISEL2. T21EXIS Bit Field	Description
0_B	01_B	11_B	MON4
0_B	10_B	00_B	lin_rxd_i
0_B	10_B	01_B	reserved
0_B	10_B	10_B	cc6_ch0
0_B	10_B	11_B	cc6_ch1
0_B	11_B	00_B	cc6_cout60
0_B	11_B	01_B	cc6_cout61
0_B	11_B	10_B	cc6_cout62
0_B	11_B	11_B	cc6_cout63

7.7.2 Input Pin Function Selection

MODPISELx registers control the selection of the input pin functions. For UART, the selection of the RXD line also enables the corresponding TXD line.

Peripheral Input Select Register

SCU_MODPISEL	Offset	Reset Value
Peripheral Input Select Register	$0B8_H$	see Table 90



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Returns 0 if read; should be written with 0.
SSC12_S_MRST_OUTSEL	18	rw	**Output selection for SSC12_S_MRST** See **Chapter 15.4**. 0_B **SSC1**, SSC1_S_MRST 1_B **SSC2**, SSC2_S_MRST
SSC12_M_MTSR_OUTSEL	17	rw	**Output selection for SSC12_M_MTSR** See **Chapter 15.4**. 0_B **SSC1**, SSC1_M_MTSR 1_B **SSC2**, SSC2_M_MTSR

Field	Bits	Type	Description
SSC12_M_SCK_OUTSEL	16	rw	**Output selection for SSC12_M_SCK** See **Chapter 15.4**. 0_B **SSC1**, SSC1_M_SCK 1_B **SSC2**, SSC2_M_SCK
RES	15:8	r	**Reserved** Returns 0 if read; should be written with 0.
U_TX_CONDIS	7	rw	**UART1 TxD Connection Disable** 0_B **Enable**, UART1-TX-Output -LIN Transmitter TX Input Connection available. 1_B **Disable**, UART1-TX-Output -LIN Transmitter TX Input Connection not available (can be stimulated by external port pin).
URIOS1	6	rw	**UART1 Input/Output Select** *Note: To select TXD_0 as the Transmitter output, the Port ALTSELx registers need to be configured additionally.* 0_B **Enable**, UART1 Receiver Input RXD1_0 (Connection to LIN is available). 1_B **Disable**, UART1 Receiver Input RXD1_1 (Connection to LIN is not available).
EXINT2IS	5:4	rw	**External Interrupt 2 Input Select** 00_B **EXINT2_0**, External Interrupt Input EXINT2_0 is selected. 01_B **EXINT2_1**, External Interrupt Input EXINT2_1 is selected. 10_B **EXINT2_2**, External Interrupt Input EXINT2_2 is selected. 11_B **EXINT2_3**, External Interrupt Input EXINT2_3 is selected.
EXINT1IS	3:2	rw	**External Interrupt 1 Input Select** 00_B **EXINT1_0**, External Interrupt Input EXINT1_0 is selected. 01_B **EXINT1_1**, External Interrupt Input EXINT1_1 is selected. 10_B **EXINT1_2**, External Interrupt Input EXINT1_2 is selected. 11_B **EXINT1_3**, External Interrupt Input EXINT1_3 is selected.

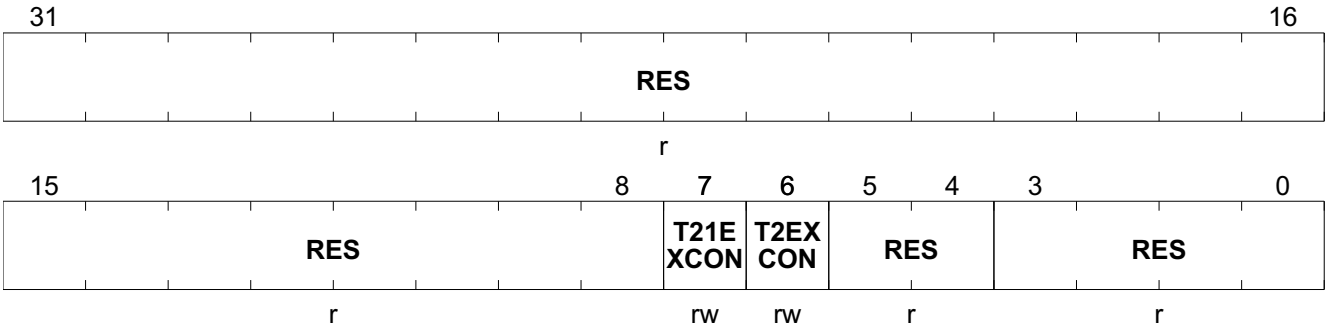
Field	Bits	Type	Description
EXINT0IS	1:0	rw	**External Interrupt 0 Input Select** 00_B **EXINT0_0**, External Interrupt Input EXINT0_0 is selected. 01_B **EXINT0_1**, External Interrupt Input EXINT0_1 is selected. 10_B **EXINT0_2**, External Interrupt Input EXINT0_2 is selected. 11_B **EXINT0_3**, External Interrupt Input EXINT0_3 is selected.

Table 90 RESET of SCU_MODPISEL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Peripheral Input Select Register 1

SCU_MODPISEL1	Offset	Reset Value
Peripheral Input Select Register 1	$0BC_H$	see Table 91



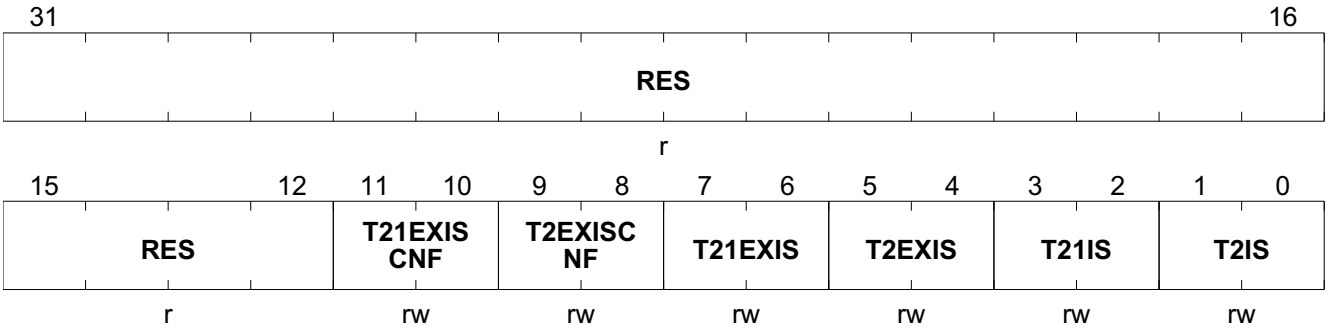
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
T21EXCON	7	rw	**Timer 21 External Input Control** 0_B **MODPISEL**, Timer 21 Input T21EX is selected according to Table 89. 1_B **CCU6**, Timer 21 Input T21EX is connected to signal CCU6.COUT60 (Ch0).
T2EXCON	6	rw	**Timer 2 External Input Control** 0_B **MODPISEL**, Timer 2 Input T2EX is selected according to Table 88. 1_B **CCU6**, Timer 2 Input T2EX is connected to signal CCU6.CC60 (Ch0).
RES	5:4	r	**Reserved** Returns 0 if read; should be written with 0.
RES	3:0	r	**Reserved** Returns 0 if read; should be written with 0.

Table 91 RESET of SCU_MODPISEL1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Peripheral Input Select Register 2

SCU_MODPISEL2	Offset	Reset Value
Peripheral Input Select Register 2	$0C0_H$	see Table 92



Field	Bits	Type	Description
RES	31:12	r	**Reserved** Returns 0 if read; should be written with 0.
T21EXISCNF	11:10	rw	**Timer 21 External Input Select Configuration** See **Table 89**. *Note: This selection takes effect only when SCU_MODPISEL1.T21EXCON = 0.*
T2EXISCNF	9:8	rw	**Timer 2 External Input Select Configuration** See **Table 88**. *Note: This selection takes effect only when SCU_MODPISEL1.T2EXCON = 0.*
T21EXIS	7:6	rw	**Timer 21 External Input Select** See **Table 89**. *Note: This selection takes effect only when SCU_MODPISEL1.T21EXCON = 0.*
T2EXIS	5:4	rw	**Timer 2 External Input Select** See **Table 88**. *Note: This selection takes effect only when SCU_MODPISEL1.T2EXCON = 0.*
T21IS	3:2	rw	**Timer 21 Input Select** 00_B **T21_0**, Timer 21 Input T21_0 is selected. 01_B **T21_1**, Timer 21 Input T21_1 is selected. 10_B **T21_2**, Timer 21 Input T21_2 is selected. 11_B **RES**, Reserved.

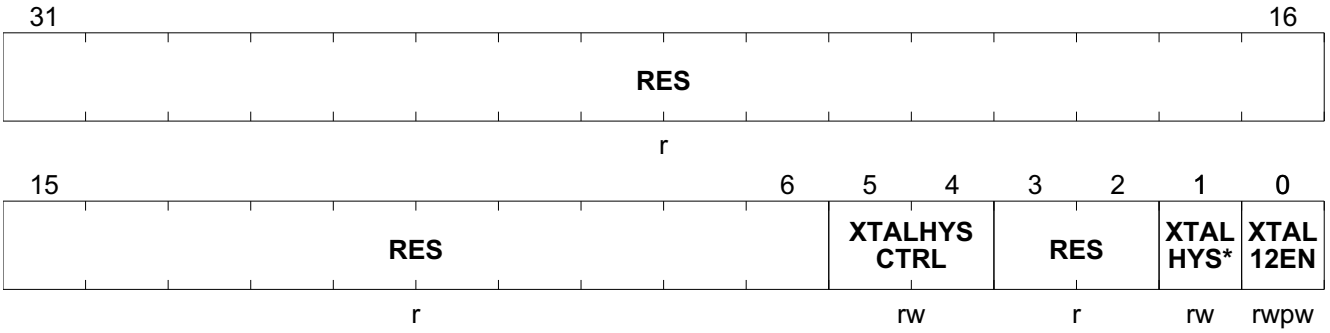
Field	Bits	Type	Description
T2IS	1:0	rw	**Timer 2 Input Select** 00_B **T2_0**, Timer 2 Input T2_0 is selected. 01_B **T2_1**, Timer 2 Input T2_1 is selected. 10_B **T2_2**, Timer 2 Input T2_2 is selected. 11_B **RES**, Reserved.

Table 92 RESET of SCU_MODPISEL2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

XTAL Control Register

SCU_XTAL_CTRL	Offset	Reset Value
Peripheral Input Select Register	100_H	see Table 93



Field	Bits	Type	Description
RES	31:6	r	**Reserved** Returns 0 if read; should be written with 0.
XTALHYSCTRL	5:4	rw	**XTAL Hysteresis Control** 00_B **XHYST_0**, 400mV nom. 01_B **XHYST_1**, 300mV nom. 10_B **XHYST_2**, 200mV nom. 11_B **XHYST_3**, 100mV nom.
RES	3:2	r	**Reserved** Returns 0 if read; should be written with 0.
XTALHYSEN	1	rw	**XTAL Hysteresis Enable** *Note: The Hysteresis must be disabled in external input clock mode for input frequencies greater than 24MHz.* 0_B **Disable**, Hysteresis is disabled 1_B **Enable**, Hysteresis is enabled

Field	Bits	Type	Description
XTAL12EN	0	rwpw	**Pins XTAL1/2 Enable Bit** 0_B **Disable**, Pins XTAL1/2 is not available. This setting overrides the **SCU_OSC_CON**.XPD setting. 1_B **Enable**, Pins XTAL1/2 are available

Table 93 RESET of SCU_XTAL_CTRL

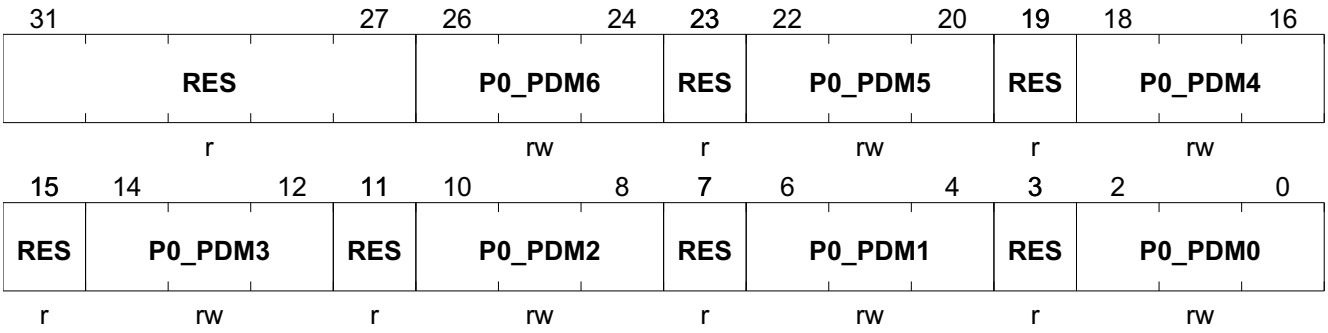
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0032_H$	RESET_TYPE_4		

7.7.3 Port Output Control

Px_POCONy registers controls the output driver strength for each of the bidirectional port pin through the bit field PDMn, where x denotes the port number and n denotes the pin number.

Port Output Control Register

SCU_P0_POCON0	Offset	Reset Value
Port Output Control Register	$0E8_H$	see Table 94



Field	Bits	Type	Description
RES	31:27	r	**Reserved** Returns 0 if read; should be written with 0.
P0_PDM6	26:24	rw	**P0.6 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong-sharp**, Strong driver and sharp edge mode 001_B **Strong-med**, Strong driver and medium edge mode 010_B **Strong-soft**, Strong driver and soft edge mode 011_B **Weak**, Weak driver 100_B **Medium**, Medium driver 101_B **Medium**, Medium driver 110_B **Medium**, Medium driver 111_B **Weak**, Weak driver
RES	23	r	**Reserved** Returns 0 if read; should be written with 0.
P0_PDM5	22:20	rw	**P0.5 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong-sharp**, Strong driver and sharp edge mode 001_B **Strong-med**, Strong driver and medium edge mode 010_B **Strong-soft**, Strong driver and soft edge mode 011_B **Weak**, Weak driver 100_B **Medium**, Medium driver 101_B **Medium**, Medium driver 110_B **Medium**, Medium driver 111_B **Weak**, Weak driver
RES	19	r	**Reserved** Returns 0 if read; should be written with 0.

Field	Bits	Type	Description
P0_PDM4	18:16	rw	**P0.4 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong-sharp**, Strong driver and sharp edge mode 001_B **Strong-med**, Strong driver and medium edge mode 010_B **Strong-soft** , Strong driver and soft edge mode 011_B **Weak**, Weak driver 100_B **Medium**, Medium driver 101_B **Medium**, Medium driver 110_B **Medium**, Medium driver 111_B **Weak**, Weak driver
RES	15	r	**Reserved** Returns 0 if read; should be written with 0.
P0_PDM3	14:12	rw	**P0.3 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong-sharp**, Strong driver and sharp edge mode 001_B **Strong-med**, Strong driver and medium edge mode 010_B **Strong-soft** , Strong driver and soft edge mode 011_B **Weak**, Weak driver 100_B **Medium**, Medium driver 101_B **Medium**, Medium driver 110_B **Medium**, Medium driver 111_B **Weak**, Weak driver
RES	11	r	**Reserved** Returns 0 if read; should be written with 0.
P0_PDM2	10:8	rw	**P0.2 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong-sharp**, Strong driver and sharp edge mode 001_B **Strong-med**, Strong driver and medium edge mode 010_B **Strong-soft** , Strong driver and soft edge mode 011_B **Weak**, Weak driver 100_B **Medium**, Medium driver 101_B **Medium**, Medium driver 110_B **Medium**, Medium driver 111_B **Weak**, Weak driver
RES	7	r	**Reserved** Returns 0 if read; should be written with 0.
P0_PDM1	6:4	rw	**P0.1 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong-sharp**, Strong driver and sharp edge mode 001_B **Strong-med**, Strong driver and medium edge mode 010_B **Strong-soft** , Strong driver and soft edge mode 011_B **Weak**, Weak driver 100_B **Medium**, Medium driver 101_B **Medium**, Medium driver 110_B **Medium**, Medium driver 111_B **Weak**, Weak driver

Field	Bits	Type	Description
RES	3	r	**Reserved** Returns 0 if read; should be written with 0.
P0_PDM0	2:0	rw	**P0.0 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong-sharp**, Strong driver and sharp edge mode 001_B **Strong-med**, Strong driver and medium edge mode 010_B **Strong-soft** , Strong driver and soft edge mode 011_B **Weak**, Weak driver 100_B **Medium**, Medium driver 101_B **Medium**, Medium driver 110_B **Medium**, Medium driver 111_B **Weak**, Weak driver

1) Defines the current the respective driver can deliver to the external circuitry.

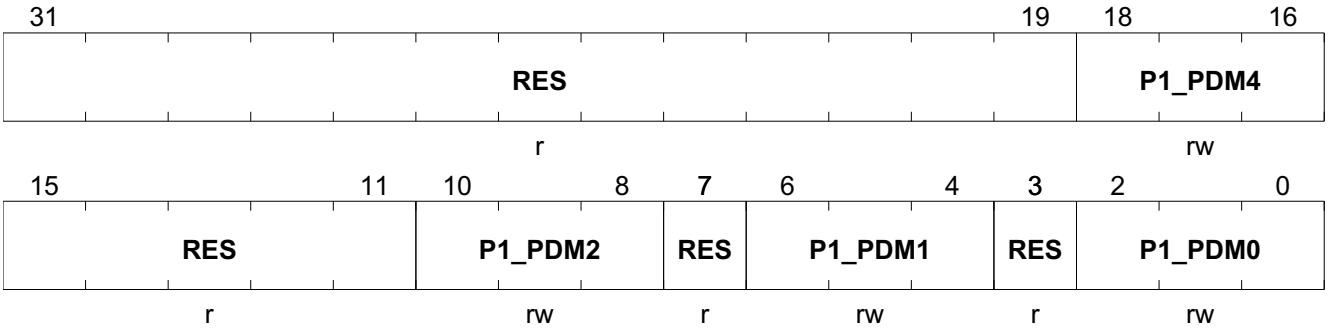
2) Defines the switching characteristics to the respective new output driver. This also influences the peak currents through the driver when producing an edge, i.e. when changing the output level.

Table 94 RESET of SCU_P0_POCON0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Port Output Control Register

SCU_P1_POCON0	Offset	Reset Value
Port Output Control Register	$0F8_H$	see Table 95



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Returns 0 if read; should be written with 0.
P1_PDM4	18:16	rw	**P1.4 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong-sharp**, Strong driver and sharp edge mode 001_B **Strong-med**, Strong driver and medium edge mode 010_B **Strong-soft** , Strong driver and soft edge mode 011_B **Weak**, Weak driver 100_B **Medium**, Medium driver 101_B **Medium**, Medium driver 110_B **Medium**, Medium driver 111_B **Weak**, Weak driver
RES	15:11	r	**Reserved** Returns 0 if read; should be written with 0.
P1_PDM2	10:8	rw	**P1.2 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **not used**, Not used 001_B **not used**, Not used 010_B **not used**, Not Used 011_B **Weak**, Weak driver 100_B **Medium**, Medium driver 101_B **Medium**, Medium driver 110_B **Medium**, Medium driver 111_B **Weak**, Weak driver
RES	7	r	**Reserved** Returns 0 if read; should be written with 0.

Field	Bits	Type	Description
P1_PDM1	6:4	rw	**P1.1 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **not used**, Not used 001_B **not used**, Not used 010_B **not used**, Not Used 011_B **Weak**, Weak driver 100_B **Medium**, Medium driver 101_B **Medium**, Medium driver 110_B **Medium**, Medium driver 111_B **Weak**, Weak driver
RES	3	r	**Reserved** Returns 0 if read; should be written with 0.
P1_PDM0	2:0	rw	**P1.0 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **not used**, Not used 001_B **not used**, Not used 010_B **not used**, Not Used 011_B **Weak**, Weak driver 100_B **Medium**, Medium driver 101_B **Medium**, Medium driver 110_B **Medium**, Medium driver 111_B **Weak**, Weak driver

1) Defines the current the respective driver can deliver to the external circuitry.

2) Defines the switching characteristics to the respective new output driver. This also influences the peak currents through the driver when producing an edge, i.e. when changing the output level.

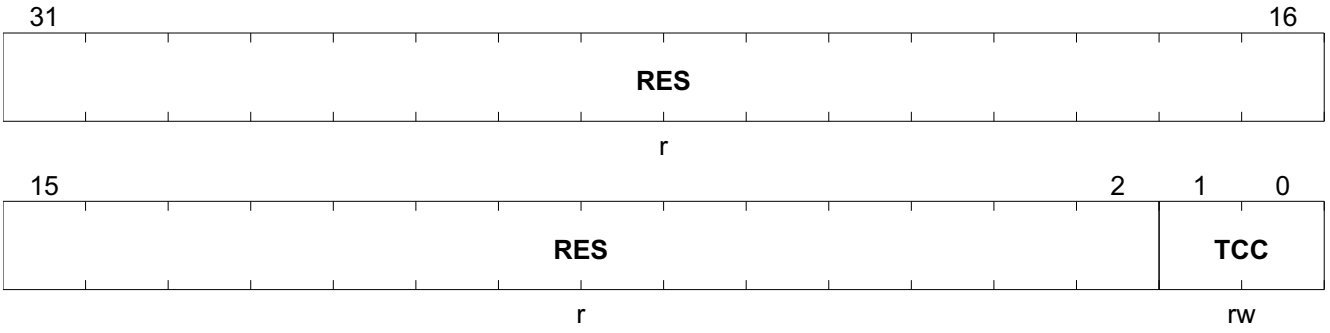
Table 95 RESET of SCU_P1_POCON0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Temperature Compensation Control Register

The TCCR register controls the temperature compensation of all the GPIO output port pins with strong drivers, i.e. on a device level. The TCCR register has no effect on output port pins that operate in the weak and medium driver modes.

SCU_TCCR	Offset	Reset Value
Temperature Compensation Control Register	$0F4_H$	see Table 96



Field	Bits	Type	Description
RES	31:2	r	**Reserved** Returns 0 if read; should be written with 0.
TCC	1:0	rw	**Temperature Compensation Control** The slew rate of the output driver is kept stable over the selected temperature range: 00_B **T1**, T_J: -40 °C to 0 °C 01_B **T2**, T_J: 0 °C to 40 °C 10_B **T3**, T_J: 40 °C to 80 °C 11_B **T4**, T_J: 80 °C to 150 °C

Table 96 RESET of SCU_TCCR

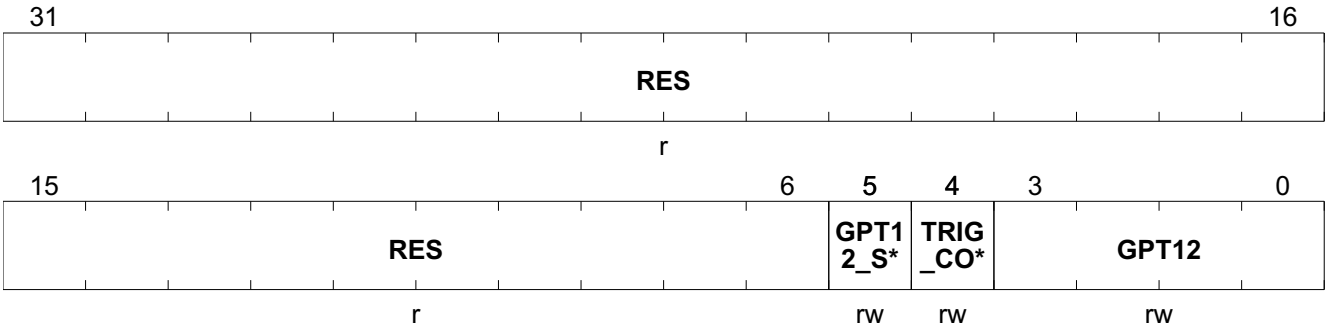
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

7.7.4 GPT12 T3IN/T4IN Input Pin Function Selection

GPT12PISEL register control the selection of the input pin functions of T3INB and T4IND in GPT12.

GPT12 Peripheral Input Select Register

SCU_GPT12PISEL	Offset	Reset Value
GPT12 Peripheral Input Select Register	$0D0_H$	see Table 97



Field	Bits	Type	Description
RES	31:6	r	**Reserved** Returns 0 if read; should be written with 0.
GPT12_SEL	5	rw	**CCU6 Trigger Configuration.** 0_B **T_21**, CCU6_INT is triggered by Timer21 1_B **GPT12**, CCU6_INT is triggered by GPT12PISEL.GPT12
TRIG_CONF	4	rw	**CCU6 Trigger Configuration.** 0_B **Single**, Trigger is just for one measurement (default) 1_B **Edge**, Trigger is present until next input edge (selected by GPT12) - continuous measurement.
GPT12	3:0	rw	**GPT12 TIN3B / TIN4D Input Select** 0000_B**CC60**, CC60 0001_B**CC61**, CC61 0010_B**CC62**, CC62 0011_B**T12 ZM**, T12 ZM 0100_B**T12 PM**, T12 PM 0101_B**T12 CM0**, T12 CM0 0110_B**T12 CM1**, T12 CM1 0111_B**T12 CM2**, T12 CM2 1000_B**T13 PM**, T13 PM 1001_B**T13 ZM**, T13 ZM 1010_B**T13 CM**, T13 CM 1011_B**Edge**, any pos or neg edge on CC60/61/62 1100_B**Res**, RES 1101_B**Res**, RES 1110_B**Res**, RES 1111_B**Res**, RES

Table 97 RESET of SCU_GPT12PISEL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

7.8 Differential Unit Trigger Enable

The Differential Unit inside DPP1 module requires enable signals for telling the processing when to accept and calculate a new result based on an incoming trigger signal. To realize a certain blank timer for the DU Unit to perform the measurements aligned to the dedicated PWM Signal the Timer 13 of CCU6 is used.

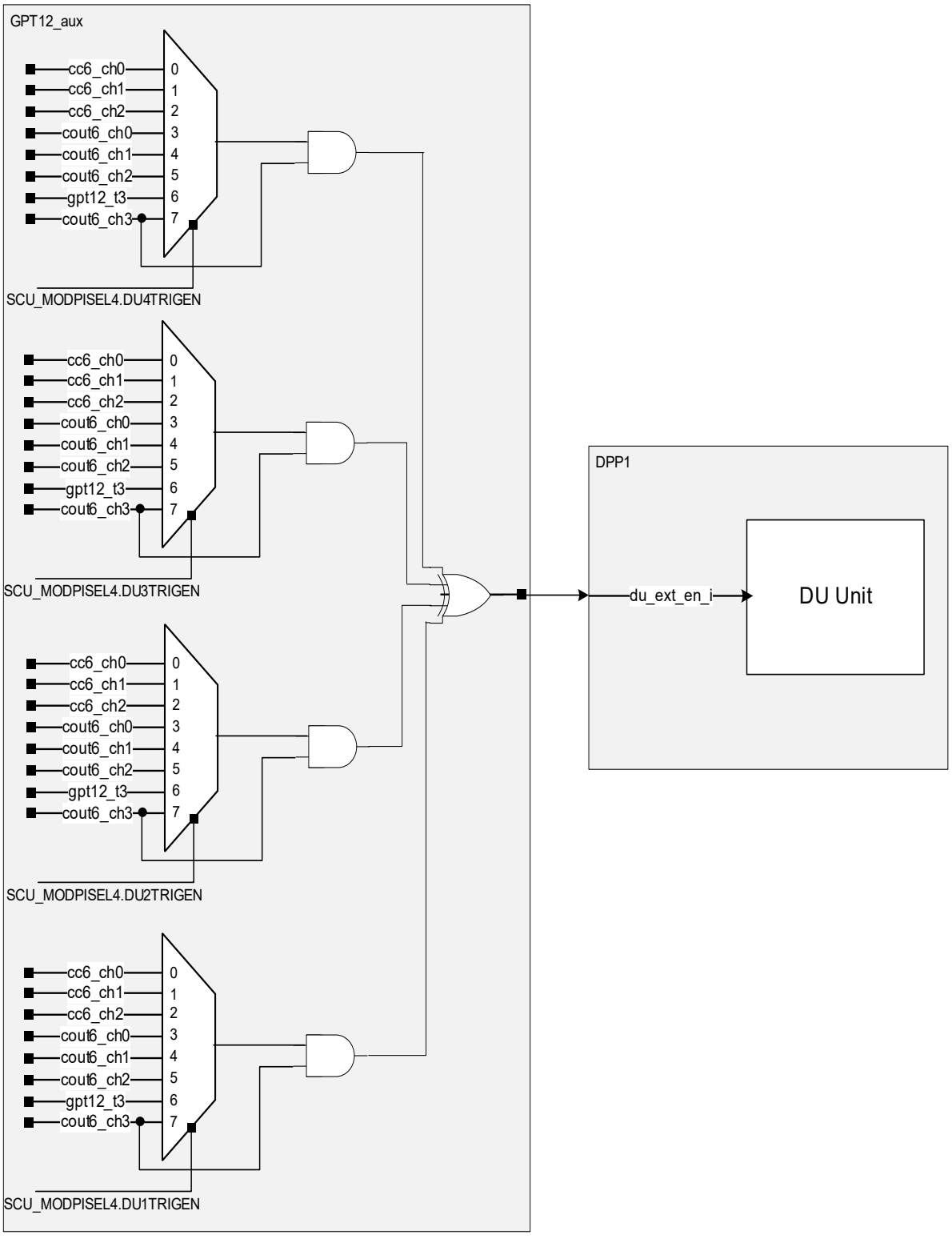


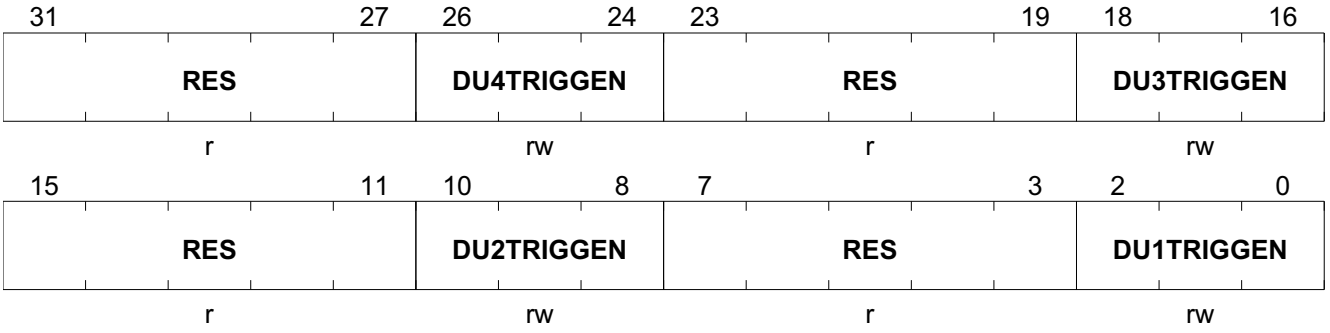
Figure 34 Differential Unit

7.8.1 Differential Unit Trigger

7.8.1.1 Differential Unit Trigger register

Peripheral Input Select Register 4

SCU_MODPISEL4	Offset	Reset Value
Peripheral Input Select Register 4	$0FC_H$	see Table 98



Field	Bits	Type	Description
RES	31:27	r	**Reserved** Returns 0 if read; should be written with 0.
DU4TRIGGEN	26:24	rw	**Differential Unit Trigger Enable** *Note: These bits configure the enable input of the differential unit.* 000_B **CC60**, CC60 is selected. 001_B **CC61**, CC61 is selected. 010_B **CC62**, CC62 is selected. 011_B **COUT60**, COUT60 is selected. 100_B **COUT61**, COUT61 is selected. 101_B **COUT62**, COUT62 is selected. 110_B **T3OUT**, T3OUT is selected. 111_B **COUT63**, COUT63 is selected.
RES	23:19	r	**Reserved** Returns 0 if read; should be written with 0.

Field	Bits	Type	Description
DU3TRIGGEN	18:16	rw	**Differential Unit Trigger Enable** *Note: These bits configure the enable input of the differential unit.* 000_B **CC60**, CC60 is selected. 001_B **CC61**, CC61 is selected. 010_B **CC62**, CC62 is selected. 011_B **COUT60**, COUT60 is selected. 100_B **COUT61**, COUT61 is selected. 101_B **COUT62**, COUT62 is selected. 110_B **T3OUT**, T3OUT is selected. 111_B **COUT63**, COUT63 is selected.
RES	15:11	r	**Reserved** Returns 0 if read; should be written with 0.
DU2TRIGGEN	10:8	rw	**Differential Unit Trigger Enable** *Note: These bits configure the enable input of the differential unit.* 000_B **CC60**, CC60 is selected. 001_B **CC61**, CC61 is selected. 010_B **CC62**, CC62 is selected. 011_B **COUT60**, COUT60 is selected. 100_B **COUT61**, COUT61 is selected. 101_B **COUT62**, COUT62 is selected. 110_B **T3OUT**, T3OUT is selected. 111_B **COUT63**, COUT63 is selected.
RES	7:3	r	**Reserved** Returns 0 if read; should be written with 0.
DU1TRIGGEN	2:0	rw	**Differential Unit Trigger Enable** *Note: These bits configure the enable input of the differential unit.* 000_B **CC60**, CC60 is selected. 001_B **CC61**, CC61 is selected. 010_B **CC62**, CC62 is selected. 011_B **COUT60**, COUT60 is selected. 100_B **COUT61**, COUT61 is selected. 101_B **COUT62**, COUT62 is selected. 110_B **T3OUT**, T3OUT is selected. 111_B **COUT63**, COUT63 is selected.

Table 98 RESET of SCU_MODPISEL4

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0403\ 0100_H$	RESET_TYPE_3		

7.9 Flexible Peripheral Management

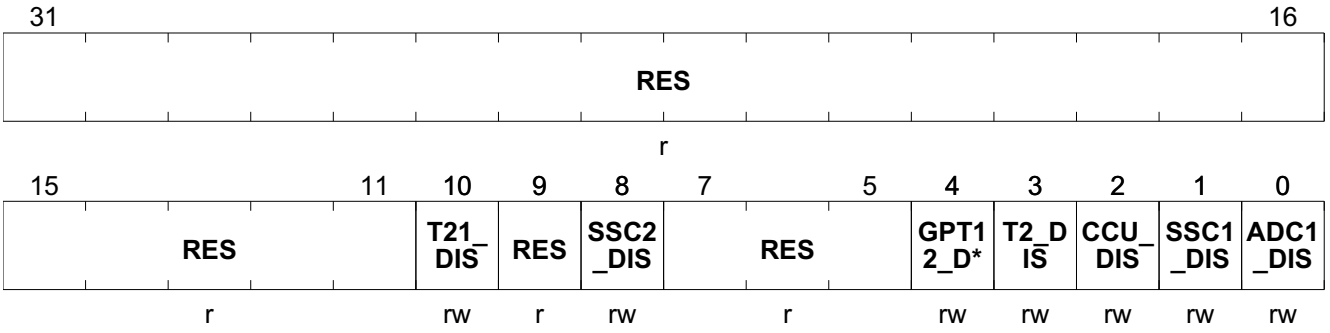
The Flexible Peripheral Management sub-module provides the system designer greater control on the operational status of each individual digital peripheral. Peripherals which are not required for a particular functionality can be disabled by programming the assigned register bits which would gate off the clock inputs. This would further reduce overall power consumption of the microcontroller.

Each register bit controls one peripheral. When this bit is set, the request signal to gate the peripheral clock is activated. The peripheral will then synchronize the gating off of the clock to the peripheral.

7.9.1 Peripheral Management Registers

Peripheral Management Control Register

SCU_PMCON	Offset	Reset Value
Peripheral Management Control Register	060_H	see Table 99



Field	Bits	Type	Description
RES	31:11	r	**Reserved** Returns 0 if read; should be written with 0.
T21_DIS	10	rw	**T21 Disable Request. Active high.** 0_B **Enable**, T21 is in normal operation. (default) 1_B **Disable**, Request to disable the T21.
RES	9	r	**Reserved** Returns 0 if read; should be written with 0.
SSC2_DIS	8	rw	**SSC Disable Request. Active high.** 0_B **Enable**, SSC is in normal operation. (default) 1_B **Disable**, Request to disable the SSC.
RES	7:5	r	**Reserved** Returns 0 if read; should be written with 0.
GPT12_DIS	4	rw	**General Purpose Timer 12 Disable Request. Active high.** 0_B **Enable**, GPT12 is in normal operation. (default) 1_B **Disable**, Request to disable the GPT12.
T2_DIS	3	rw	**T2 Disable Request. Active high.** 0_B **Enable**, T2 is in normal operation. (default) 1_B **Disable**, Request to disable the T2.
CCU_DIS	2	rw	**CCU Disable Request. Active high.** 0_B **Enable**, CCU is in normal operation. (default) 1_B **Disabel**, Request to disable the CCU.
SSC1_DIS	1	rw	**SSC Disable Request. Active high.** 0_B **Enable**, SSC is in normal operation. (default) 1_B **Disable**, Request to disable the SSC.

Field	Bits	Type	Description
ADC1_DIS	0	rw	**ADC1 Disable Request. Active high.** 0_B **Enable**, ADC1 is in normal operation. (default) 1_B **Disable**, Request to disable the ADC.

Table 99 RESET of SCU_PMCON

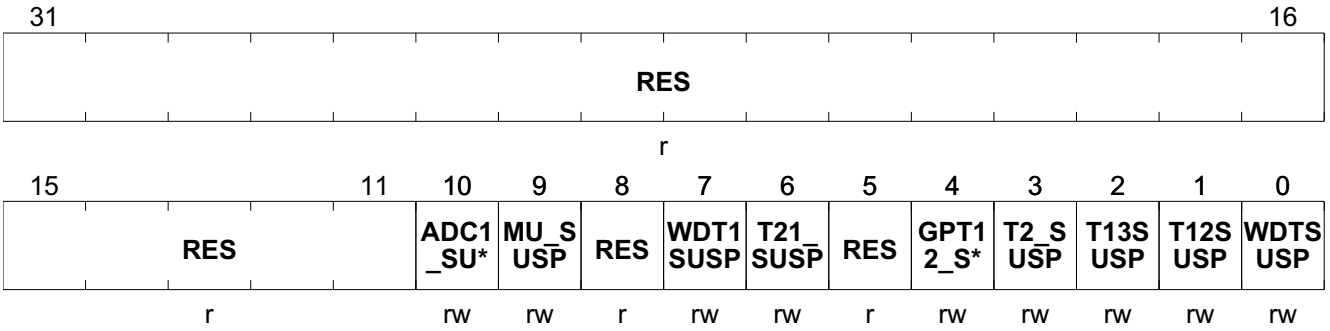
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

7.10 Module Suspend Control

When the On-Chip Debug Support (Debug Mode) is in Monitor Mode (halted_o from Arm® debug), timers in certain modules in TLE985xQX can be suspended based on the settings of their corresponding module suspend bits in register MODSUSP. When suspended, only the timer stops counting as the counter input clock is gated off. The module is still clocked so that module registers are accessible.

Module Suspend Control Register

SCU_MODSUSP	Offset	Reset Value
Module Suspend Control Register	$0C8_H$	see Table 100



Field	Bits	Type	Description
RES	31:11	r	**Reserved** Returns 0 if read; should be written with 0.
ADC1_SUSP	10	rw	**ADC1 Unit Debug Suspend Bit** 0_B **No suspend**, ADC1 will not be suspended. 1_B **Suspend**, ADC1 will be suspended.
MU_SUSP	9	rw	**Measurement Unit Debug Suspend Bit** 0_B **No Suspend**, MU will not be suspended. 1_B **Suspend**, MU will be suspended.
RES	8	r	**Reserved** Returns 0 if read; should be written with 0.
WDT1SUSP	7	rw	**Watchdog Timer 1 Debug Suspend Bit** 0_B **No Suspend**, WDT1 will not be suspended. 1_B **Suspend**, WDT1 will be suspended.
T21_SUSP	6	rw	**Timer21 Debug Suspend Bit** 0_B **No Suspend**, Timer21 will not be suspended. 1_B **Suspend**, Timer21 will be suspended.
RES	5	r	**Reserved** Returns 0 if read; should be written with 0.
GPT12_SUSP	4	rw	**GPT12 Debug Suspend Bit** 0_B **No Suspend**, GPT12 will not be suspended. 1_B **Suspend**, GPT12 will be suspended.

Field	Bits	Type	Description
T2_SUSP	3	rw	**Timer2 Debug Suspend Bit** 0_B **No Suspend**, Timer2 will not be suspended. 1_B **Suspend**, Timer2 will be suspended.
T13SUSP	2	rw	**Timer 13 Debug Suspend Bit** When suspended, additionally the T13 PWM output is set to inactive level. 0_B **No Suspend**, Timer 13 in Capture/Compare Unit will not be suspended. 1_B **Suspend**, Timer 13 in Capture/Compare Unit will be suspended.
T12SUSP	1	rw	**Timer 12 Debug Suspend Bit** When suspended, additionally the T12 PWM outputs are set to inactive level and capture inputs are disabled. 0_B **No Suspend**, Timer 12 in Capture/Compare Unit will not be suspended. 1_B **Suspend**, Timer 12 in Capture/Compare Unit will be suspended.
WDTSUSP	0	rw	**SCU Watchdog Timer Debug Suspend Bit** 0_B **No Suspend**, WDT will not be suspended. 1_B **Suspend**, WDT will be suspended.

Table 100 RESET of SCU_MODSUSP

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0081_H$	RESET_TYPE_3		

7.11 Baud-rate Generator

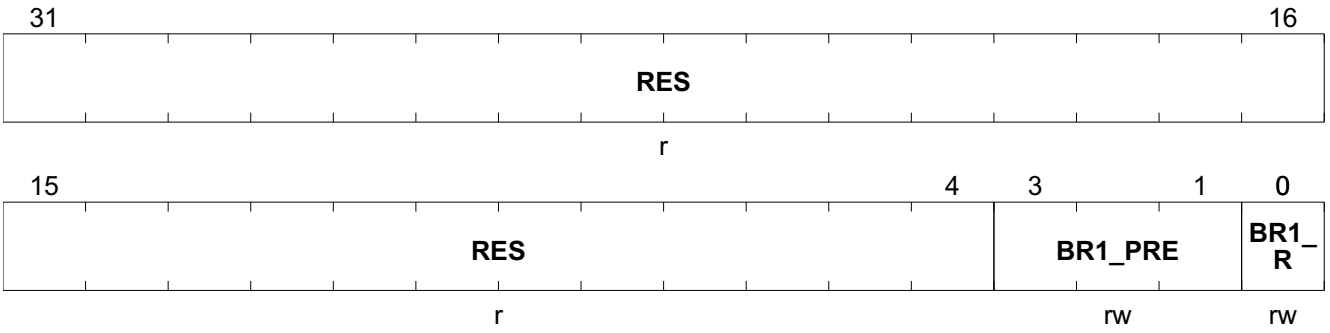
The baud-rate generator in SCU is used to generate the baud rate for the UART module. See **Chapter 19.6** for the functional description. The SCU contains two of this registers. One is dedicated for UART1 and the other for UART2.

7.11.1 Baudrate Generator Registers

7.11.1.1 Baud-rate Generator Control and Status Registers

Baud Rate Control Register 1

SCU_BCON1	Offset	Reset Value
Baud Rate Control Register 1	088_H	see Table 101



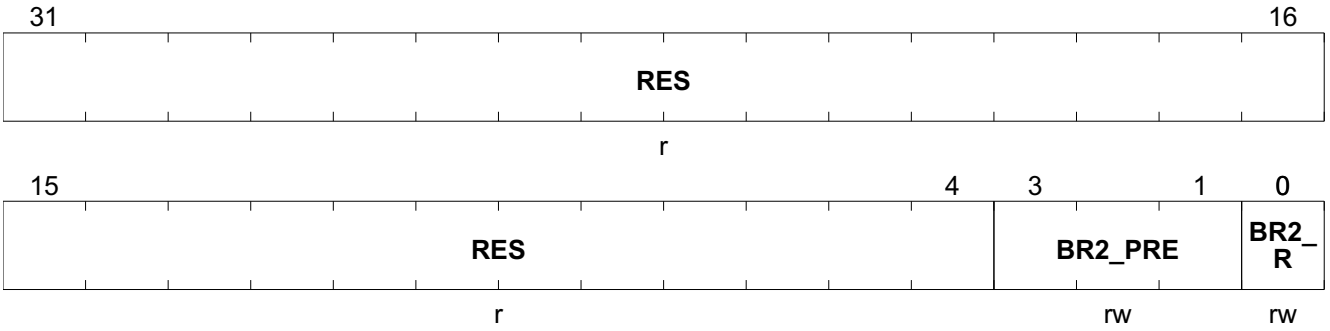
Field	Bits	Type	Description
RES	31:4	r	**Reserved** Returns 0 if read; should be written with 0.
BR1_PRE	3:1	rw	**Prescaler Bit** Selects the input clock for f_{DIV} which is derived from the peripheral clock. Others: reserved 000_B **div 1**, $f_{DIV} = f_{PCLK}$ 001_B **div 2**, $f_{DIV} = f_{PCLK}/2$ 010_B **div 4**, $f_{DIV} = f_{PCLK}/4$ 011_B **div 8**, $f_{DIV} = f_{PCLK}/8$ 100_B **div 16**, $f_{DIV} = f_{PCLK}/16$ 101_B **div 32**, $f_{DIV} = f_{PCLK}/32$
BR1_R	0	rw	**Baud Rate Generator Run Control Bit** *Note: BR_VALUE should only be written if R = 0.* 0_B **Disable**, Baud-rate generator disabled. 1_B **Enable**, Baud-rate generator enabled.

Table 101 RESET of SCU_BCON1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Baud Rate Control Register 2

SCU_BCON2	Offset	Reset Value
Baud Rate Control Register 2	098_H	see Table 102



Field	Bits	Type	Description
RES	31:4	r	**Reserved** Returns 0 if read; should be written with 0.
BR2_PRE	3:1	rw	**Prescaler Bit** Selects the input clock for f_{DIV} which is derived from the peripheral clock. Others: reserved 000_B **div 1**, $f_{DIV} = f_{PCLK}$ 001_B **div 2**, $f_{DIV} = f_{PCLK}/2$ 010_B **div 4**, $f_{DIV} = f_{PCLK}/4$ 011_B **div 8**, $f_{DIV} = f_{PCLK}/8$ 100_B **div 16**, $f_{DIV} = f_{PCLK}/16$ 101_B **div 32**, $f_{DIV} = f_{PCLK}/32$
BR2_R	0	rw	**Baud Rate Generator Run Control Bit** *Note: BR_VALUE should only be written if R = 0.* 0_B **Disable**, Baud-rate generator disabled. 1_B **Enable**, Baud-rate generator enabled.

Table 102 RESET of SCU_BCON2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

7.11.1.2 Baud-rate Generator Timer/Reload Registers

The low and high bytes of the baud rate timer/reload register BG contains the 11-bit reload value for the baud rate timer and the 5-bit fractional divider selection.

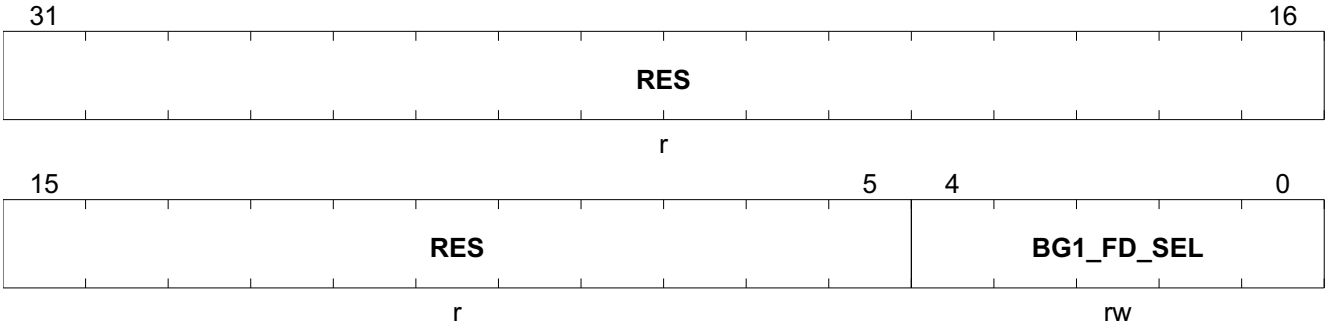
Reading the low byte of register BG returns the content of the lower three bits of the baud rate timer and the FD_SEL setting, while reading the high byte returns the content of the upper 8 bits of the baud rate timer.

Writing to register BG loads the baud rate timer with the reload and fractional divider values from the BG register, the first instruction cycle after BCON.R is set.

BG should only be written if R = 0. Also this register should be present twice. One is for UART1 and the other for UART2.

Baud Rate Timer/Reload Register, Low Byte 1

SCU_BGL1	Offset	Reset Value
Baud Rate Timer/Reload Register, Low Byte 1	$08C_H$	see Table 103



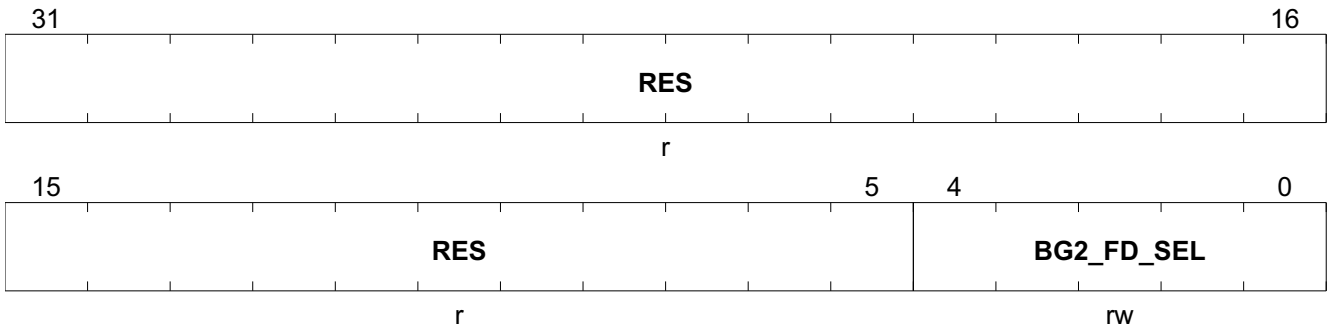
Field	Bits	Type	Description
RES	31:5	r	**Reserved** Returns 0 if read; should be written with 0.
BG1_FD_SEL	4:0	rw	**Fractional Divider Selection** Selects the fractional divider to be n/32, where n is the value of FD_SEL and is in the range of 0 to 31. For example, writing 0001_B to FD_SEL selects the fractional divider to be1/32. *Note: Fractional divider has no effect if BR_VALUE = 000_H .*

Table 103 RESET of SCU_BGL1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Baud Rate Timer/Reload Register, Low Byte 2

SCU_BGL2	Offset	Reset Value
Baud Rate Timer/Reload Register, Low Byte 2	$09C_H$	see Table 104



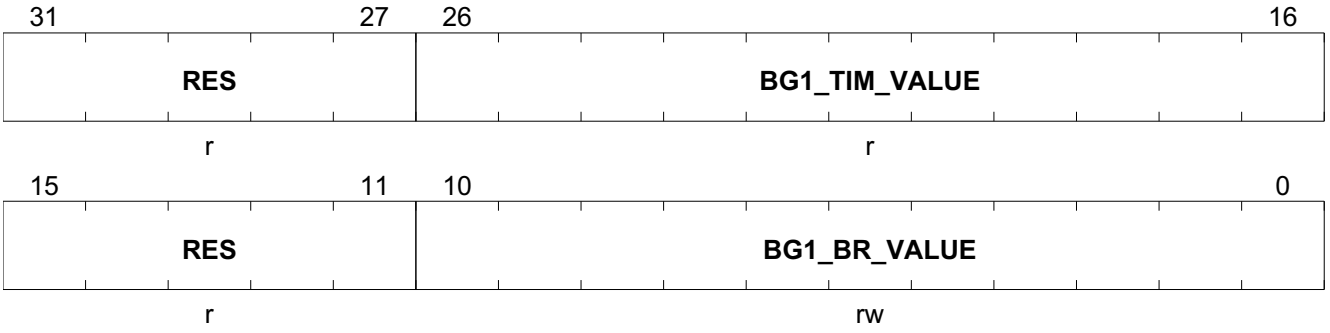
Field	Bits	Type	Description
RES	31:5	r	**Reserved** Returns 0 if read; should be written with 0.
BG2_FD_SEL	4:0	rw	**Fractional Divider Selection** Selects the fractional divider to be n/32, where n is the value of FD_SEL and is in the range of 0 to 31. For example, writing 0001_B to FD_SEL selects the fractional divider to be1/32. *Note: Fractional divider has no effect if BR_VALUE = 000_H.*

Table 104 RESET of SCU_BGL2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Baud Rate Timer/Reload Register

SCU_BG1	Offset	Reset Value
Baud Rate Timer/Reload Register	090_H	see Table 105



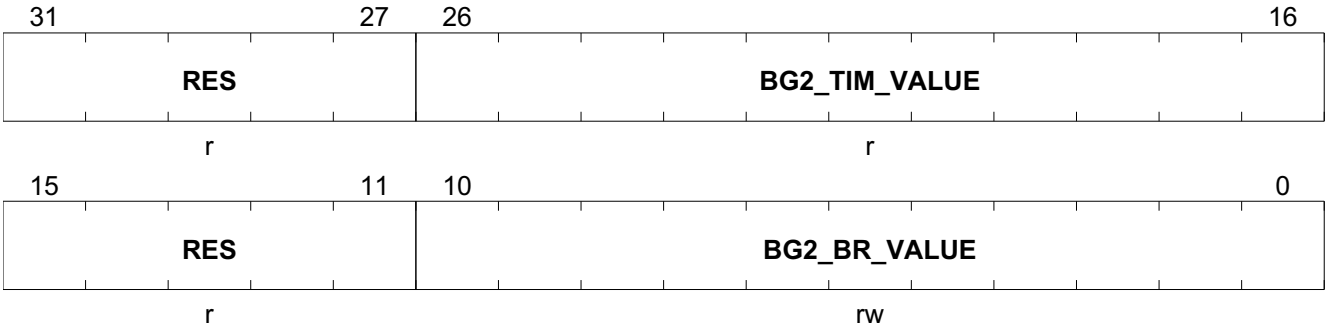
Field	Bits	Type	Description
RES	31:27	r	**Reserved** Returns 0 if read; should be written with 0.
BG1_TIM_VALUE	26:16	r	**Baud Rate Timer Value** 11-bit Baud Rate Timer value. The definition of the 11-bit reload value is as follows: Other combinations equivalent. 000_H **Bypassed**, Baud-rate timer is bypassed. 001_H **1**, 002_H **2**, $7FE_H$ **2046**, $7FF_H$ **2047**,
RES	15:11	r	**Reserved** Returns 0 if read; should be written with 0.
BG1_BR_VALUE	10:0	rw	**Baud Rate Reload Value** 11-bit Baud Rate Timer/Reload value. This Register can be only written if SCU_BCON1.BR1_R = 0 The definition of the 11-bit reload value is as follows: Other combinations equivalent. 000_H **Bypass**, Baud-rate timer is bypassed. 001_H **1**, 002_H **2**, $7FE_H$ **2046**, $7FF_H$ **2047**,

Table 105 RESET of SCU_BG1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Baud Rate Timer/Reload Register

SCU_BG2	Offset	Reset Value
Baud Rate Timer/Reload Register	$0A0_H$	see Table 106



Field	Bits	Type	Description
RES	31:27	r	**Reserved** Returns 0 if read; should be written with 0.
BG2_TIM_VALUE	26:16	r	**Baud Rate Timer Value** 11-bit Baud Rate Timer value. The definition of the 11-bit reload value is as follows: Other combinations equivalent. 000_H **Bypassed**, Baud-rate timer is bypassed. 001_H **1**, 002_H **2**, $7FE_H$ **2046**, $7FF_H$ **2047**,
RES	15:11	r	**Reserved** Returns 0 if read; should be written with 0.
BG2_BR_VALUE	10:0	rw	**Baud Rate Reload Value** 11-bit Baud Rate Reload value. This Register can be only written if SCU_BCON2.BR2_R = 0 The definition of the 11-bit reload value is as follows: Other bit combinations equivalent. 000_H **Bypass**, Baud-rate timer is bypassed. 001_H **1**, 002_H **2**, $7FE_H$ **2046**, $7FF_H$ **2047**,

Table 106 RESET of SCU_BG2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

7.12 LIN Break and Sync Byte Detection

Hardware logic is implemented in the SCU to support LIN Break and Synch Byte detection. See **Chapter 19.7** for the functional description.

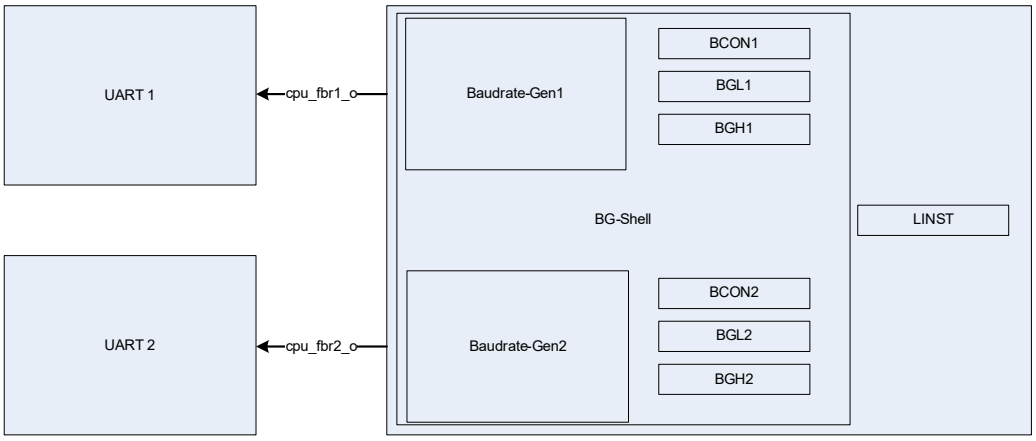


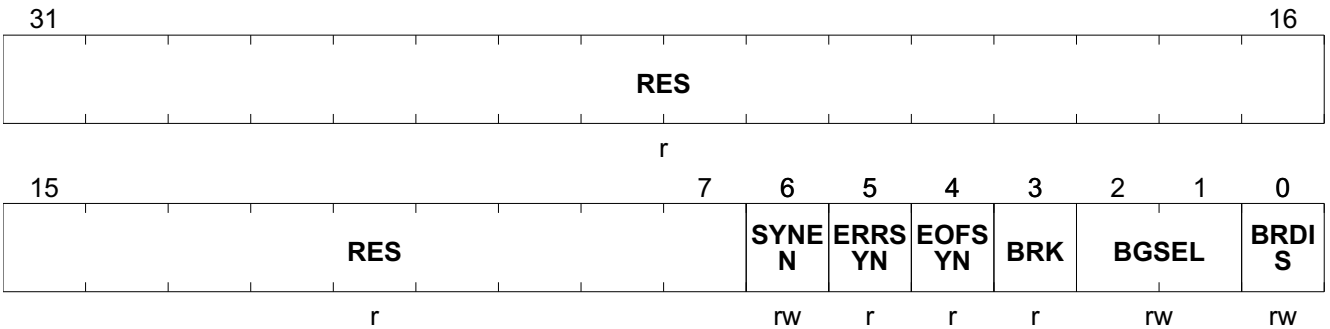
Figure 35 Structure of Baudrate Generator

7.12.1 LIN Break and Sync Byte Detection Control

7.12.1.1 LIN Break and Sync Byte Registers

LIN Status Register

SCU_LINST	Offset	Reset Value
LIN Status Register	094_H	see Table 107



Field	Bits	Type	Description
RES	31:7	r	**Reserved** Returns 0 if read; should be written with 0.

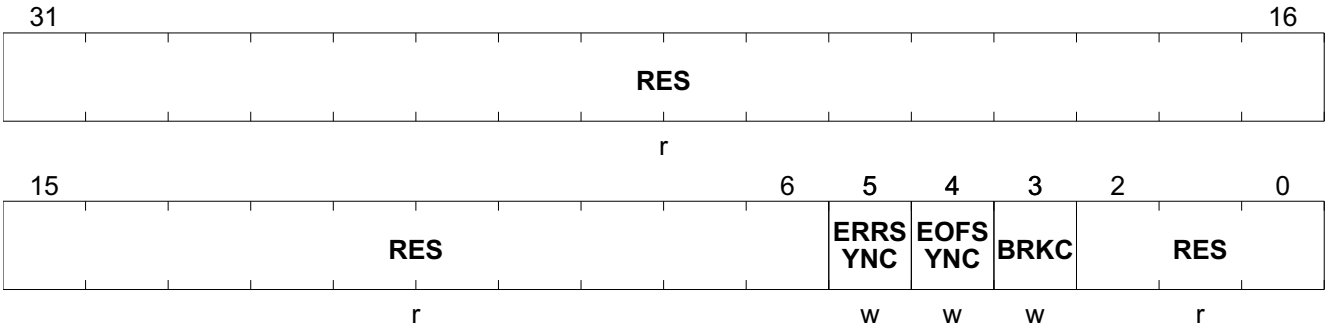
Field	Bits	Type	Description
SYNEN	6	rw	**End of SYN Byte and SYN Byte Error Interrupts Enable** 0_B **Disable**, End of SYN Byte and SYN Byte Error Interrupts are not enabled. 1_B **Enable**, End of SYN Byte and SYN Byte Error Interrupts are enabled.
ERRSYN	5	r	**SYN Byte Error Interrupt Flag** This bit is set by hardware and can only be cleared by software. 0_B **Disable**, Error is not detected in SYN Byte. 1_B **Enable**, Error is detected in SYN Byte.
EOFSYN	4	r	**End of SYN Byte Interrupt Flag** This bit is set by hardware and can only be cleared by software. 0_B **Disable**, End of SYN Byte is not detected. 1_B **Enable**, End of SYN Byte is detected.
BRK	3	r	**Break Field Flag** This bit is set by hardware and can only be cleared by software. 0_B **Disable**, Break Field is not detected. 1_B **Enable**, Break Field is detected.
BGSEL	2:1	rw	**Baud Rate Select for Detection** For different values of BGSEL, the baud rate range for detection is defined by the following formula: f_{pclk}/(2184*2^BGSEL) < baud rate range < f_{pclk}/(72*2^BGSEL) where BGSEL = 00_B, 01_B, 10_B, 11_B. See **Table 30** for bit field BGSEL definition for different input frequencies.
BRDIS	0	rw	**Baud Rate Detection Disable** 0_B **Disable**, Break/Synch detection is enabled. 1_B **Enable**, Break/Synch detection is disabled.

Table 107 RESET of SCU_LINST

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

LIN Status Clear Register

SCU_LINSCLR	Offset	Reset Value
LIN Status Clear Register	$0A4_H$	see Table 108



Field	Bits	Type	Description
RES	31:6	r	**Reserved** Returns 0 if read; should be written with 0.
ERRSYNC	5	w	**SYN Byte Error Interrupt Flag** This bit is set by software and can only be cleared by hardware. 0_B **Not cleared**, Error in SYN Byte not cleared. 1_B **Cleared**, Error in SYN Byte cleared.
EOFSYNC	4	w	**End of SYN Byte Interrupt Flag Clear** This bit is set by software and can only be cleared by hardware. 0_B **Not cleared**, End of SYN Byte is not cleared. 1_B **Cleared**, End of SYN Byte is cleared.
BRKC	3	w	**Break Field Flag Clear** This bit is set by software and can only be cleared by hardware. 0_B **Not cleared**, Break Field is not cleared. 1_B **Cleared**, Break Field is cleared.
RES	2:0	r	**Reserved** Returns 0 if read; should be written with 0.

Table 108 RESET of SCU_LINSCLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

7.13 Watchdog Timer

There are two watchdog timers in the system: SCU Watchdog Timer (WDT) within TLE985xQX, and external watchdog timer (WDT1). The description in this section refers to the SCU WDT.

The Watchdog Timer is a sub-module in the System Control Unit (SCU). The Watchdog Timer (WDT) provides a highly reliable and secure way to detect and recover from software or hardware failures. The WDT helps to abort an accidental malfunction of the TLE985xQX in a user-specified time period. When enabled, the WDT will cause the TLE985xQX system to be reset if the WDT is not serviced within a user-programmable time period. The CPU must service the WDT within this time interval to prevent the WDT from causing a TLE985xQX system reset. Hence, routine service of the WDT confirms that the system is functioning properly.

The WDT is by default disabled.

In debug mode, the WDT is default suspended and stops counting (its debug suspend bit is default set i.e., **SCU_MODSUSP**.WDTSUSP = 1. Therefore during debugging, there is no need to refresh the WDT. Refer to **Section 7.10**.

Features

- 16-bit Watchdog Timer
- Programmable reload value for upper 8 bits of timer
- Programmable window boundary
- Selectable input frequency of $f_{PCLK}/2$ or $f_{PCLK}/128$

7.13.1 Functional Description

The Watchdog Timer is a 16-bit timer, which is incremented by a count rate of $f_{PCLK}/2$ or $f_{PCLK}/128$. This 16-bit timer is realized as two concatenated 8-bit timers. The upper 8 bits of the Watchdog Timer can be preset to a user-programmable value via a watchdog service access in order to vary the watchdog expire time. The lower 8 bits are reset on each service access. **Figure 36** shows the block diagram of the watchdog timer unit.

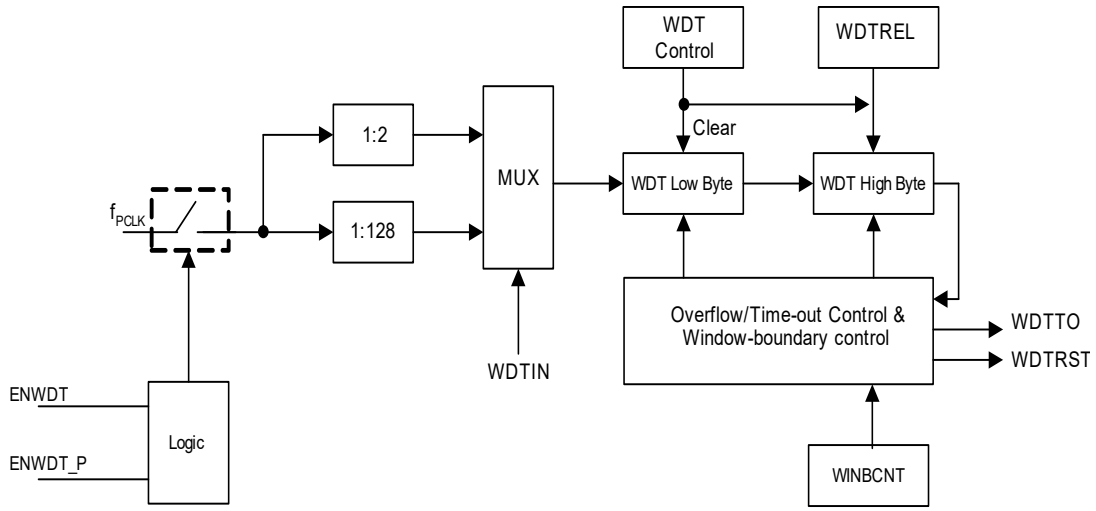


Figure 36 WDT Block Diagram

If the Watchdog Timer is enabled by setting bit WDTEN to 1, the timer is set to a user-defined start value and begins counting up. It must be serviced before the counter overflows. Servicing is performed through refresh. This reloads the timer with the start value, and normal operation continues.

If the WDT is not serviced before the timer overflows, a system malfunction is assumed and normal mode is terminated. A Watchdog Timer NMI request (WDTTO) is asserted and Prewarning is entered. The Prewarning lasts for 30_H counts. During the Prewarning period, refreshing of the Watchdog Timer is ignored and the Watchdog Timer cannot be disabled. A reset (WDTRST) of the TLE985xQX is imminent and can no longer be stopped. If refresh happens at the same time an overflow occurs, Watchdog Timer will not go into Prewarning period.

The Watchdog Timer must be serviced periodically so that its count value will not overflow. Servicing the Watchdog Timer clears the low byte and reloads the high byte with the preset value in bit field WDTREL. Servicing the Watchdog Timer also clears the bit WDTRS.

The Watchdog Timer has a ‘programmable window boundary’, it disallows refresh during the Watchdog Timer’s count-up. A Refresh during this window-boundary will cause the Watchdog Timer to activate WDTRST. The window boundary is from 0000_H to (WDTWINB,00H). This feature can be enabled by WINBEN.

After being serviced, the Watchdog Timer continues counting up from the value (<WDTREL> * 2^8). The time period for an overflow of the Watchdog Timer is programmable in two ways:

- **the input frequency** to the Watchdog Timer can be selected via bit WDTIN in register WDTCON to be either $f_{PCLK}/2$ or $f_{PCLK}/128$.
- **the reload value** WDTREL for the high byte of WDT can be programmed in register WDTREL.

The period PWDT between servicing the Watchdog Timer and the next overflow can be determined by the following formula:

$$P_{WDT} = \frac{2^{(1+WDTIN\times 6)} \times (2^{16} - WDTREL \times 2^{8})}{f_{PCLK}} \quad (7.5)$$

If the Window-Boundary Refresh feature of the Watchdog Timer is enabled, the period P_{WDT} between servicing the Watchdog Timer and the next overflow is shortened if WDTWINB is greater than WDTREL. See also **Figure 37**. This period can be calculated by the same formula by replacing WDTREL with WDTWINB. In order for this feature to be useful, WDTWINB cannot be smaller than WDTREL.

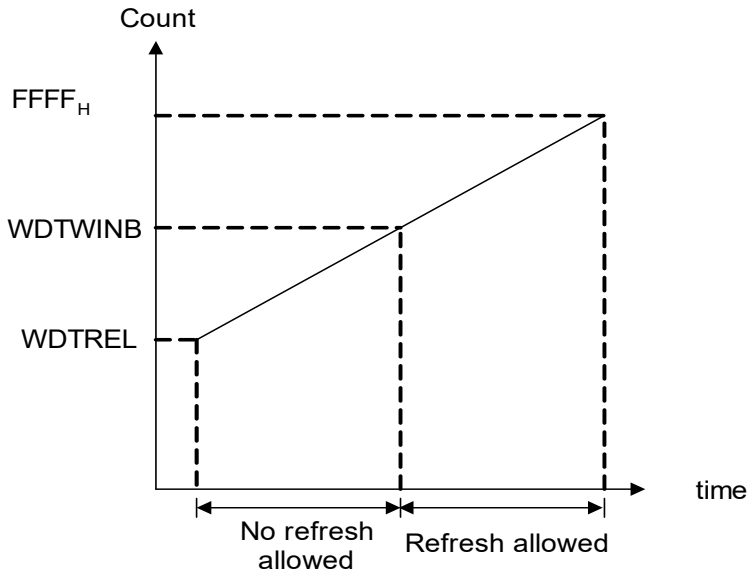


Figure 37 Watchdog Timer Timing Diagram

Table 109 lists the possible ranges for the watchdog time which can be achieved using a certain module clock. Some numbers are rounded to 3 significant digits.

Table 109 Watchdog Time Ranges

Reload Value in WDTREL	Prescaler for f_{PCLK}					
	2 (WDTIN = 0)			128 (WDTIN = 1)		
	40 MHz	20 MHz	13.3 MHz	40 MHz	20 MHz	13.3 MHz
FF_H	12.8 µs	25.6 µs	38.4 µs	0.82 ms	1.64 ms	2.46 ms
$7F_H$	1.65 ms	3.30 ms	4.95 ms	106 ms	211 ms	317 ms
00_H	3.28 ms	6.55 ms	9.83 ms	210 ms	419 ms	629 ms

Notes

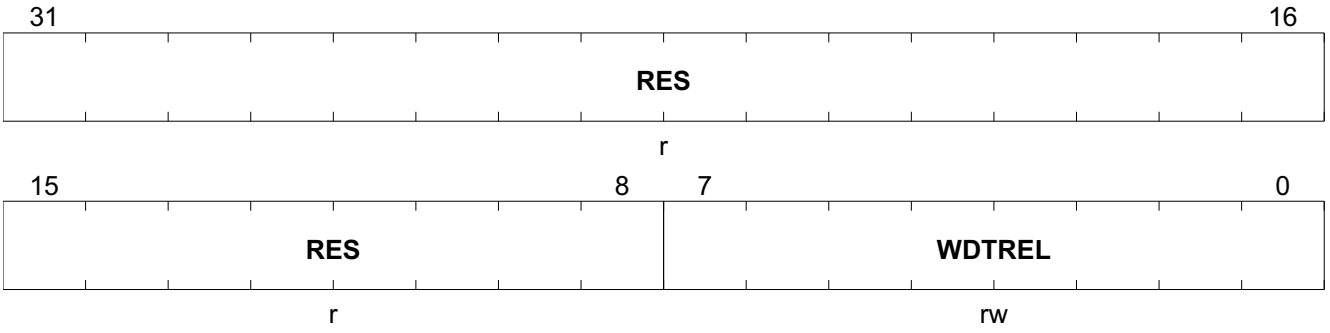
1. *For safety reasons, the user is advised to rewrite WDTCON each time before the Watchdog Timer is serviced.*
2. *The Watchdog Timer can be suspended when Debug Mode enters Monitor Mode and has the Debug-Suspend signal activated, provided the respective suspend bit, WDTSUSP in SFR SCU_MODSUSP, is set. See **Section 7.10**.*

7.13.2 Register Description

The current count value of the Watchdog Timer is contained in the Watchdog Timer Register WDT, which is a non-bit-addressable read-only register. The operation of the Watchdog Timer is controlled by its bit-addressable Watchdog Timer Control Register WDTCON. WDTREL register specifies the reload value for the high byte of the timer. WDTWINB specifies Watchdog Window-Boundary count value.

Watchdog Timer Reload Register

SCU_WDTREL	Offset	Reset Value
Watchdog Timer Reload Register	078_H	see Table 110



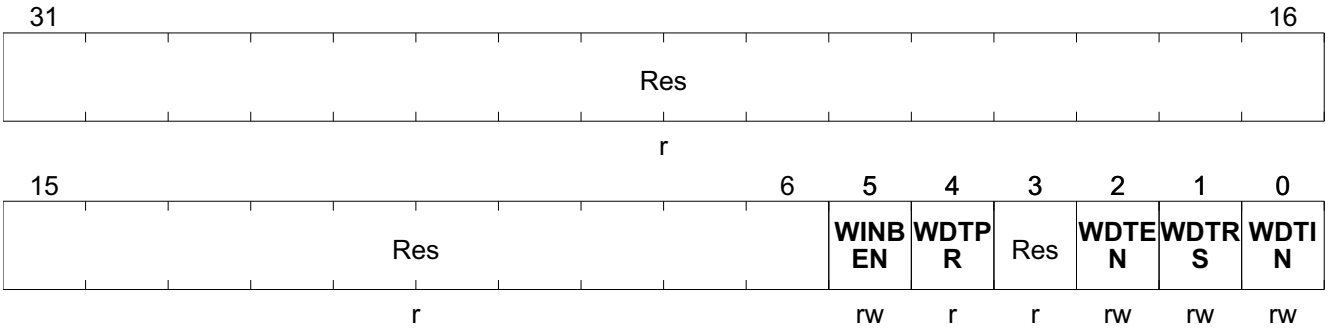
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
WDTREL	7:0	rw	**Watchdog Timer Reload Value** (for the high byte of WDT)

Table 110 RESET of SCU_WDTREL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Watchdog Timer Control Register

SCU_WDTCON	Offset	Reset Value
Watchdog Timer Control Register	050_H	see Table 111



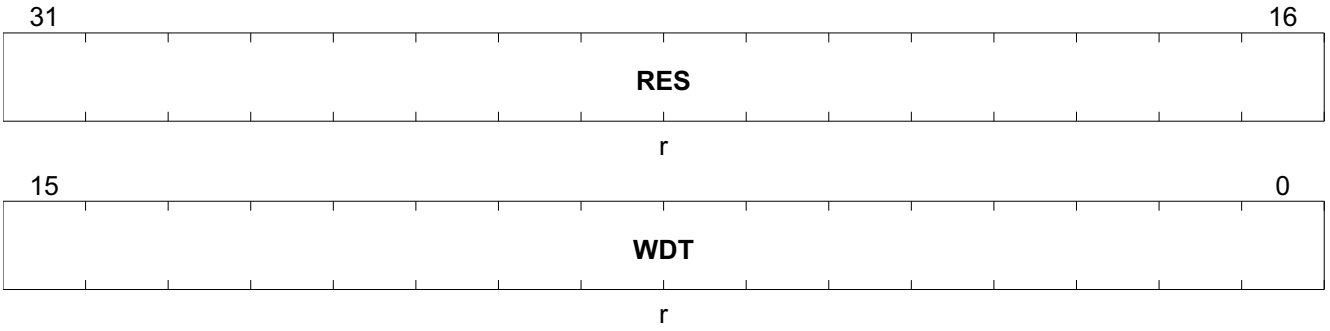
Field	Bits	Type	Description
Res	31:6	r	**Reserved** Returns 0 if read; should be written with 0.
WINBEN	5	rw	**Watchdog Window-Boundary Enable** 0_B **Disable**, Watchdog Window-Boundary feature is disabled. (default) 1_B **Enable**, Watchdog Window-Boundary feature is enabled.
WDTPR	4	r	**Watchdog Prewarning Mode Flag** This bit is set to 1 when a Watchdog error is detected. The Watchdog Timer has issued an NMI trap and is in Prewarning Mode. A reset of the chip occurs after the prewarning period has expired. 0_B **Normal**, Normal mode (default after reset) 1_B **Prewarn**, The Watchdog is operating in Prewarning Mode
Res	3	r	**Reserved** Returns 0 if read; should be written with 0.
WDTEN	2	rw	**WDT Enable** WDTEN is a protected bit. If the Protection Scheme is activated then this bit cannot be written directly. For more information on Protection Scheme, see **Section 7.15**. *Note: Clearing WDTEN bit to 0 during Prewarning Mode (WDTPR = 1) has no effect.* 0_B **Disable**, WDT is disabled 1_B **Enable**, WDT is enabled
WDTRS	1	rw	**WDT Refresh Start** Active high. Set to start refresh operation on the watchdog timer. Cleared automatically by hardware after it is set by software.
WDTIN	0	rw	**Watchdog Timer Input Frequency Selection** 0_B **DIV 2**, Input frequency is $f_{PCLK}/2$ 1_B **DIV 128**, Input frequency is $f_{PCLK}/128$

Table 111 RESET of SCU_WDTCON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Watchdog Timer

SCU_WDT	Offset	Reset Value
Watchdog Timer	080_H	see Table 112



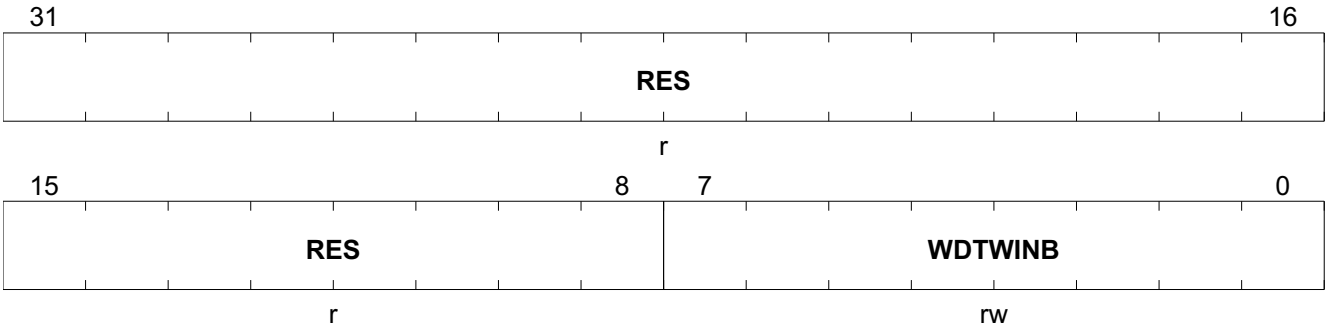
Field	Bits	Type	Description
RES	31:16	r	**Reserved** Returns 0 if read; should be written with 0.
WDT	15:0	r	**Watchdog Timer Current Value**

Table 112 RESET of SCU_WDT

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Watchdog Window-Boundary Count

SCU_WDTWINB	Offset	Reset Value
Watchdog Window-Boundary Count	$07C_H$	see Table 113



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
WDTWINB	7:0	rw	**Watchdog Window-Boundary Count Value** This value is programmable. Within this Window-Boundary range from 0000_H to (WDTWINB, 00_H), the WDT cannot do a Refresh, else it will cause a WDTRST to be asserted. WDTWINB is matched to WDTH.

Table 113 RESET of SCU_WDTWINB

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

7.14 Error Detection and Correction Control for Memories

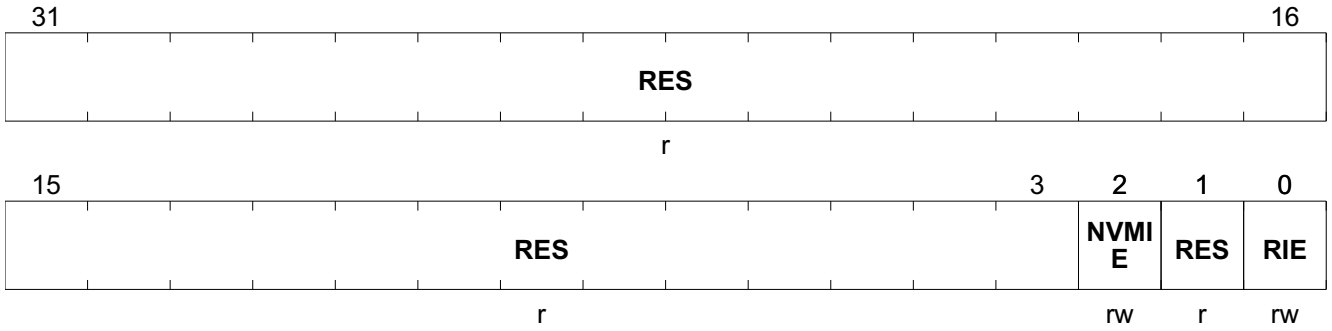
This section defines the registers used for error detection and correction control of memories – namely RAM and NVM, which support this function.

7.14.1 Error Detection and Correction Control Register

The EDCCON register determines the generation of an NMI due to double bit ECC error when read these memories.

Error Detection and Correction Control Register

SCU_EDCCON	Offset	Reset Value
Error Detection and Correction Control Register	$0D4_H$	see Table 114



Field	Bits	Type	Description
RES	31:3	r	**Reserved** Returns 0 if read; should be written with 0.
NVMIE	2	rw	**NVM Double Bit ECC Error Interrupt Enable** 0_B **Disable**, No NMI is generated when a double bit ECC error occurs reading NVM. 1_B **Enable**, An NMI is generated when a double bit ECC error occurs reading NVM.
RES	1	r	**Reserved** Returns 0 if read; should be written with 0.
RIE	0	rw	**RAM Double Bit ECC Error Interrupt Enable** 0_B **Disable**, No NMI is generated when a double bit ECC error occurs reading RAM. 1_B **Enable**, An NMI is generated when a double bit ECC error occurs reading RAM.

Table 114 RESET of SCU_EDCCON

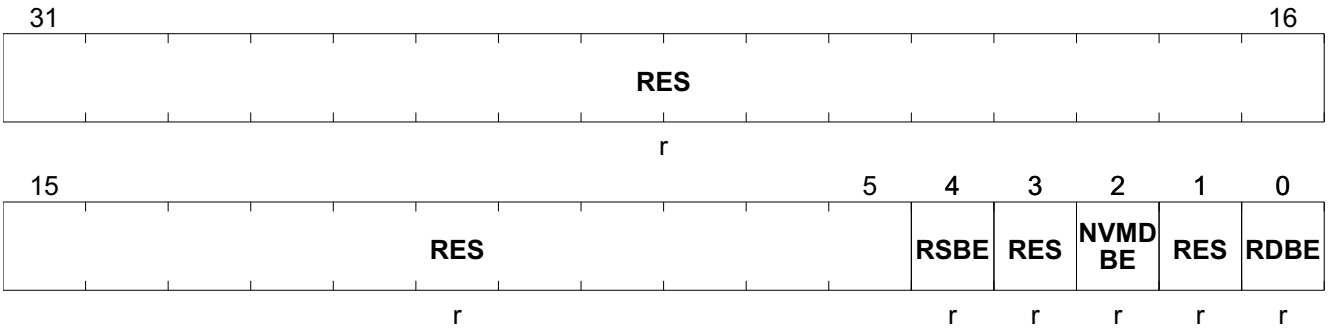
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

7.14.2 Error Detection and Correction Status Register

The EDCSTAT register contains the status flags of ECC errors when read these memories. The corresponding flags for the IRAM are not more necessary, because IRAM was removed.

Error Detection and Correction Status Register

SCU_EDCSTAT	Offset	Reset Value
Error Detection and Correction Status Register	$0D8_H$	see Table 115



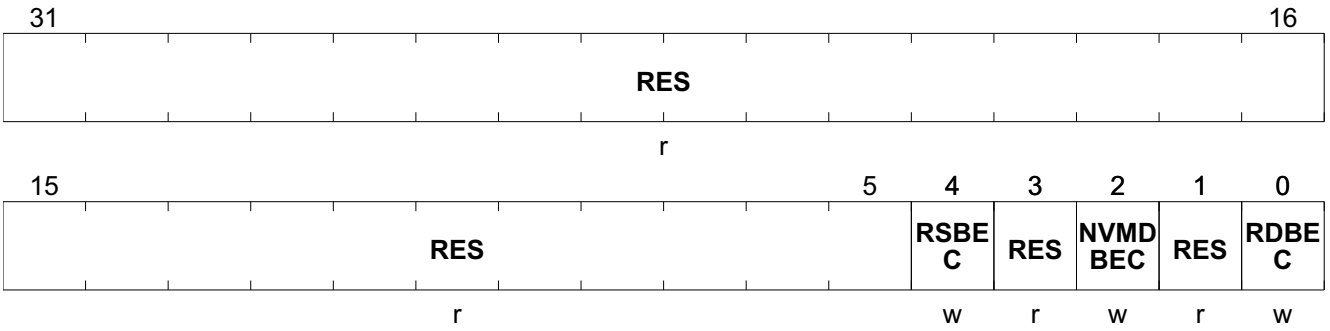
Field	Bits	Type	Description
RES	31:5	r	**Reserved** Returns 0 if read; should be written with 0.
RSBE	4	r	**RAM Single Bit Error** This bit is set by hardware and can be cleared only by software. 0_B **No Error**, No single bit error on RAM has occurred. 1_B **Error**, A single bit error on RAM has occurred.
RES	3	r	**Reserved** Returns 0 if read; should be written with 0.
NVMDBE	2	r	**NVM Double Bit Error** This bit is set by hardware and can be cleared only by software. 0_B **No Error**, No double bit error on NVM has occurred. 1_B **Error**, A double bit error on NVM has occurred.
RES	1	r	**Reserved** Returns 0 if read; should be written with 0.
RDBE	0	r	**RAM Double Bit Error** This bit is set by hardware and can be cleared only by software. 0_B **No Error**, No double bit error on RAM has occurred. 1_B **Error**, A double bit error on RAM has occurred.

Table 115 RESET of SCU_EDCSTAT

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	0000 0000$_H$	RESET_TYPE_4		

Error Detection and Correction Status Clear Register

SCU_EDCSCLR	Offset	Reset Value
Error Detection and Correction Status Clear Register	$10C_H$	see Table 116



Field	Bits	Type	Description
RES	31:5	r	**Reserved** Returns 0 if read; should be written with 0.
RSBEC	4	w	**RAM Single Bit Error Clear** This bit is set by software and can be cleared only by hardware. 0_B **Not cleared**, A single bit error on RAM is not cleared. 1_B **Cleared**, A single bit error on RAM is cleared.
RES	3	r	**Reserved** Returns 0 if read; should be written with 0.
NVMDBEC	2	w	**NVM Double Bit Error Clear** This bit is set by software and can be cleared only by hardware. 0_B **Not Cleared**, A double bit error on NVM is not cleared. 1_B **Cleared**, A double bit error on NVM is cleared.
RES	1	r	**Reserved** Returns 0 if read; should be written with 0.
RDBEC	0	w	**RAM Double Bit Error Clear** This bit is set by software and can be cleared only by hardware. 0_B **Not Cleared**, A double bit error on RAM is not cleared. 1_B **Cleared**, A double bit error on RAM is cleared.

Table 116 RESET of SCU_EDCSCLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

7.15 Miscellaneous Control

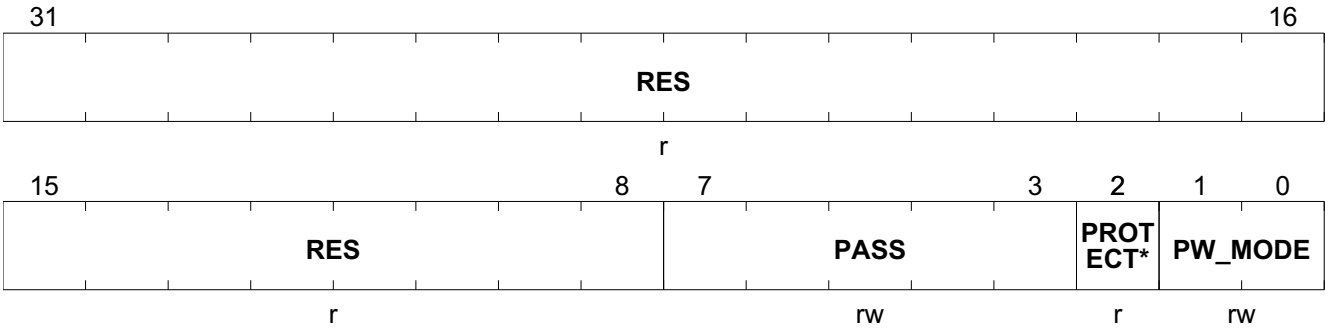
This module consists of the Bit-Protection Scheme and general system control SFRs.

7.15.1 Bit Protection Register

The Bit-Protection Scheme disallows direct software writing of selected bits (i.e. Protected bits) by the SFR PASSWD. When the bit field MODE is 11_B, writing 10011_B to the bit field PASS opens access to writing of all protected bits and writing 10101_B to the bit field PASS closes access to writing of all protected bits. Note that access is opened for maximum 32 CCLKs if the “close access” password is not written. If “open access” password is written again before the end of 32 CCLK cycles, there will be a recount of 32 CCLK cycles.

Password Register

SCU_PASSWD	Offset	Reset Value
Password Register	$0AC_H$	see Table 117



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
PASS	7:3	rw	**Password Bits** The Bit-Protection Scheme only recognizes three patterns. This Bit field is always read as ‘0’. 11000_B **Enable**, Enables writing of the bit field MODE. 10011_B **Open**, Opens access to writing of all protected bits. 10101_B **Close**, Closes access to writing of all protected bits.
PROTECT_S	2	r	**Bit-Protection Signal Status Bit** This bit shows the status of the protection. 0_B **Not protected**, Software is able to write to all protected bits. 1_B **Protected**, Software is unable to write to any protected bits.

Field	Bits	Type	Description
PW_MODE	1:0	rw	**Bit-Protection Scheme Control Bit** These two bits cannot be written directly. To change the value between 11_B and 00_B, the bit field PASS must be written with 11000_B, only then the MODE[1:0] will be registered. Other bit combinations: Scheme Enabled 00_B **Disable**, Scheme Disabled 11_B **Enable**, Scheme Enabled (default)

Table 117 RESET of SCU_PASSWD

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0007_H$	RESET_TYPE_3		

The list of protected bits is shown in **Table 118**.

Table 118 List of Protected Bits

Register	Bit Field
SCU_RSTCON	LOCKUP_EN
SCU_OSC_CON	OSCSS
	XPD
SCU_PLL_CON	NDIV
SCU_CMCON1	K1DIV
	K2DIV
	PDIV
SCU_CMCON2	PBA0CLKREL
SCU_APCLK_CTRL	CLKWDT_IE
SCU_APCLK	CPCLK_DIV
	BGCLK_DIV
	BGCLK_SEL
SCU_PMCON0	SD
	PD
	SL
SCU_VTOR	VTOR_BYP
SCU_XTAL_CTRL	XTAL12EN
ADC1_CAL_CH0_1	All fields
ADC1_CAL_CH2_3	All fields
ADC1_CAL_CH4_5	All fields
ADC1_CAL_CH6_7	All fields
ADC1_CAL_CH8_9	All fields
ADC1_CAL_CH10_11	All fields
ADC1_CAL_CH12_13	All fields

Table 118 List of Protected Bits (cont'd)

Register	Bit Field
BDRV_CP_CTRL	VCP14_15V_SEL
	VTHVCP_TRIM
	VCP9V_SET
	CPLOPWRM_EN
BDRV_DCTRIM_DRVx	COMPENS_LS
	COMPENS_HS

7.15.2 System Control and Status Registers

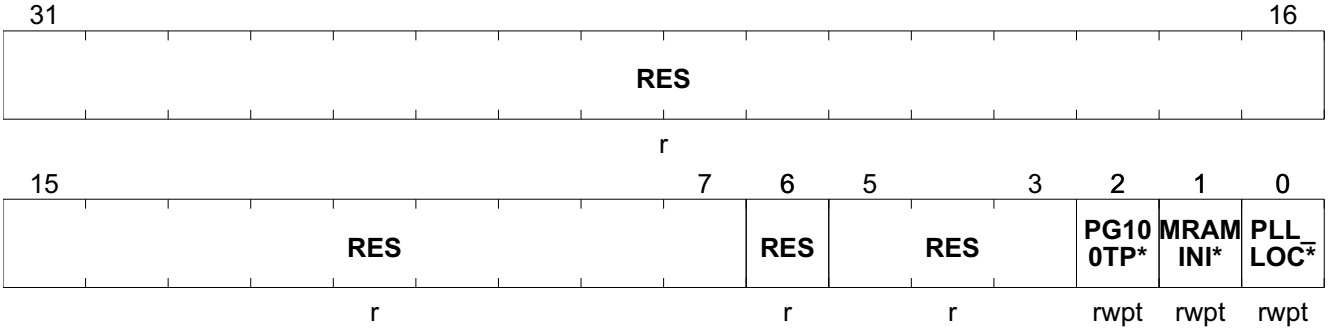
The system startup status register provide information to the user about the system initialisation with the user programmable 100 TP Page at startup. These register is written by firmware.

This register SYS__STS is reset by reset_type_4.

System Startup Status Register

It contains the main system control and status bits.

SCU_SYS_STRTUP_STS	Offset	Reset Value
System Startup Status Register	074_H	see Table 119



Field	Bits	Type	Description
RES	31:7	r	**Reserved** Returns 0 if read; should be written with 0.
RES	6	r	**Reserved** Returns 0 if read; should be written with 0.
RES	5:3	r	**Reserved** Returns 0 if read; should be written with 0.
PG100TP_CHKS_ERR	2	rwpt	**100 TP Page Checksum Error** Initialization status of trimming parameters from NVM. 0_B **OK**, initialisation of trimming parameters from NVM was successfull (checksum was correct) 1_B **Not OK**, initialisation of trimming parameter from NVM was not successfull (checksum was not correct). As a backup default values form Boot-ROM are used

Field	Bits	Type	Description
MRAMINITSTS	1	rwpt	**Map RAM Initialisation Status** Status of Map RAM initialisation. 0_B **No Fail**, Map RAM initialisation was successfull 1_B **Fail**, Map RAM initialisation was not successfull
PLL_LOCK_STS	0	rwpt	**PLL LOCK STATUS** PLL_LOCK status after FW startup. 0_B **No Fail**, 1_B **Fail**,

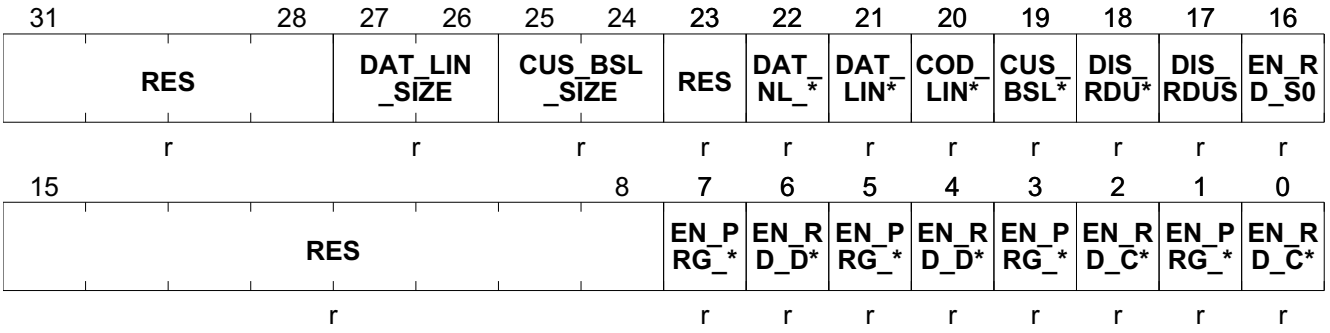
Table 119 RESET of SCU_SYS_STRTUP_STS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		

NVM Protection Status Register

This register reflects the NVM Protection Status. It is written by firmware only.

SCU_NVM_PROT_STS	Offset	Reset Value
NVM Protection Status Register	$0E0_H$	see Table 120



Field	Bits	Type	Description
RES	31:28	r	**Reserved** Returns 0 if read; should be written with 0.
DAT_LIN_SIZE	27:26	r	**Data linear Region Size Definition** Size definition of linear Data Region 00_B **0K**, data linear Size is 0K 01_B **4K**, data linear Size is 4K 10_B **8K**, data linear Size is 8K 11_B **12K**, data linear Size is 12K
CUS_BSL_SIZE	25:24	r	**CBSL Region Size Definition** Size definition of Customer BSL Region 00_B **0K**, CBSL Size is 0K 01_B **4K**, CBSL Size is 4K 10_B **8K**, CBSL Size is 8K 11_B **16K**, CBSL Size is 16K
RES	23	r	**Reserved** Returns 0 if read; should be written with 0.
DAT_NL_PW	22	r	**Status of Non-Linear Region Password / Protection** 0_B **Not Protected**, Non-Linear Region Password is not installed; Linear region is not protected. 1_B **Protected**, Non-Linear Region Password is installed; Linear region is protected.
DAT_LIN_PW	21	r	**Status of Data linear Region Password / Protection** 0_B **Not protected**, Non-Linear Region Password is not installed; Linear region is not protected. 1_B **Protected**, Non-Linear Region Password is installed; Linear region is protected.

Field	Bits	Type	Description
COD_LIN_PW	20	r	**Status of Linear Region Password / Protection** 0_B **Not Protected**, Linear Region Password is not installed; Linear region is not protected. 1_B **Protected**, Linear Region Password is installed; Linear region is protected.
CUS_BSL_PW	19	r	**Status of CBSL Region Password / Protection** 0_B **Not Protected**, CBSL Region Password is not installed; CBSL region is not protected. 1_B **Protected**, CBSL Region Password is installed; CBSL region is protected.
DIS_RDUS_S0	18	r	**Configuration of NVM Read Protection for Sector 0 with EN_RD_S0 = 0** 0_B **Not Protected**, only active when nvm_read_S0_unsafe_i = 1 and not for nvm_read_S0_unsafe_i = 0 1_B **Protected**, independent from nvm_read_S0_unsafe_i; Also write accesess to Sector 0 are prevented
DIS_RDUS	17	r	**Configuration of NVM Read Protection for Sector 1...n with EN_RD_* = 0** 0_B **Not Protected**, only active when nvm_read_unsafe_i = 1 and not for nvm_read_unsafe_i = 0 1_B **Protected**, independent from nvm_read_unsafe_i; Also write accesess to Sector 1...n are prevented
EN_RD_S0	16	r	**NVM Read Protection for Sector 0** 0_B **Protected**, The data in sector 0 can not be read over AHB-Lite Interface 1_B **Not Protected**, The data in sector 0 can be read over AHB-Lite Interface
RES	15:8	r	**Reserved** Returns 0 if read; should be written with 0.
EN_PRG_DAT_NL	7	r	**NVM Protection of Data in Non-Linear Data Sectors** 0_B **Protected**, The data in sectors of the non-linearly mapped area can not be changed 1_B **Not Protected**, The data in sectors of the non-linearly mapped area can be changed (erased or written)
EN_RD_DAT_NL	6	r	**NVM Read Protection of Data in Non-Linear Data Sectors** 0_B **Protected**, The data in sectors of the non-linearly mapped area can not be read 1_B **Not Protected**, The data in sectors of the non-linearly mapped area can be read

Field	Bits	Type	Description
EN_PRG_DAT_LIN	5	r	**NVM Protection of Data in Linear Data Sectors** 0_B **Protected**, The data in sectors of the linearly mapped area can not be changed 1_B **Not Protected**, The data in sectors of the linearly mapped area can be changed (erased or written)
EN_RD_DAT_LIN	4	r	**NVM Read Protection of Data in Linear Data Sectors** 0_B **Protected**, The data in sectors of the linearly mapped area can not be read 1_B **Not Protected**, The data in sectors of the linearly mapped area can be read
EN_PRG_COD_LIN	3	r	**NVM Protection of Data in Linear Code Sectors** 0_B **Protected**, The data in sectors of the linearly mapped area can not be changed 1_B **Not Protected**, The data in sectors of the linearly mapped area can be changed (erased or written)
EN_RD_COD_LIN	2	r	**NVM Read Protection of Data in Linear Code Sectors** 0_B **Protected**, The data in sectors of the linearly mapped area can not be read 1_B **Not Protected**, The data in sectors of the linearly mapped area can be read
EN_PRG_CUS_BSL	1	r	**NVM Protection of Data in Customer BSL Region** 0_B **Protected**, The data in region defined by NVMBSL can not be changed 1_B **Not Protected**, The data in region defined by NVMBSL can be changed (erased or written)
EN_RD_CUS_BSL	0	r	**NVM Read Protection of Data in Customer BSL Region** 0_B **Protected**, The data in region defined by NVMBSL can not be read 1_B **Not Protected**, The data in region defined by NVMBSL sectors of can be read

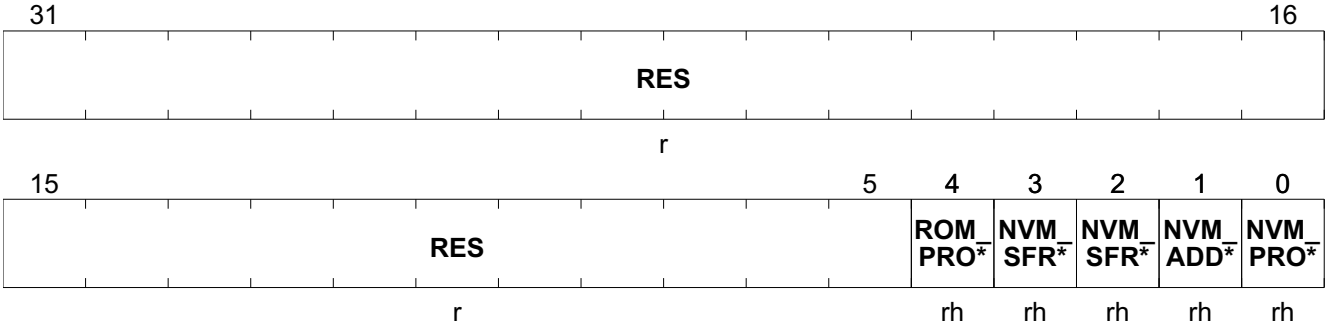
Table 120 RESET of SCU_NVM_PROT_STS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
VARIANT	$0000\ 0000_H$	VARIANT		

Memory Access Status Register

This register reflects the Memory Access Status of all System Memories. Software can only clear this register.

SCU_MEM_ACC_STS	Offset	Reset Value
Memory Access Status Register	$0E4_H$	see Table 121



Field	Bits	Type	Description
RES	31:5	r	**Reserved** Returns 0 if read; should be written with 0.
ROM_PROT_ERR	4	rh	**ROM Access Protection** 0_B **No Error**, No Protection error 1_B **Error**, Protection error
NVM_SFR_ADDR_ERR	3	rh	**NVM SFR Address Protection** 0_B **No Error**, No Protection error 1_B **Error**, Protection error
NVM_SFR_PROT_ERR	2	rh	**NVM SFR Access Protection** 0_B **No Error**, No Protection error 1_B **Error**, Protection error
NVM_ADDR_ERR	1	rh	**NVM Address Protection** 0_B **No Error**, No Protection error 1_B **Error**, Protection error
NVM_PROT_ERR	0	rh	**NVM Access Protection** 0_B **No Error**, No Protection error 1_B **Error**, Protection error

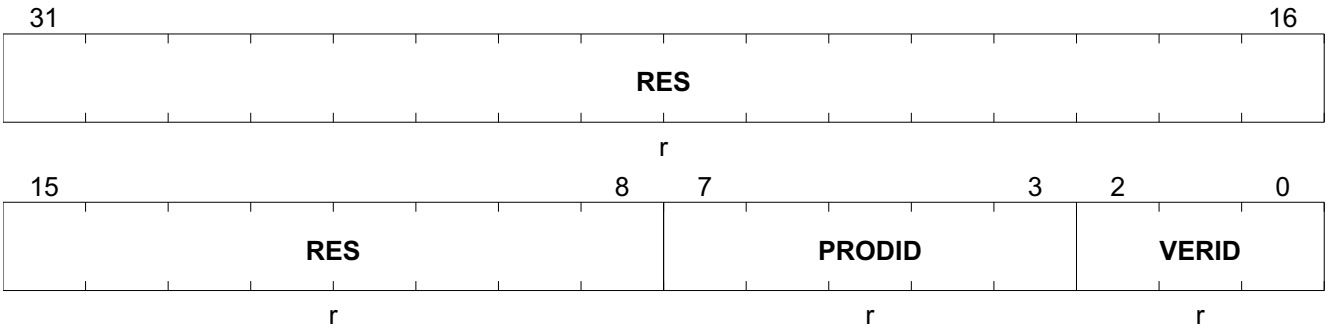
Table 121 RESET of SCU_MEM_ACC_STS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		

Identity Register

The Identity Register identifies the product and versioning.

SCU_ID	Offset	Reset Value
Identity Register	$0A8_H$	see Table 122



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
PRODID	7:3	r	**Product ID** 10000_B
VERID	2:0	r	**Version ID** Defines the stepping code of the device. 001_B 010_B

Table 122 RESET of SCU_ID

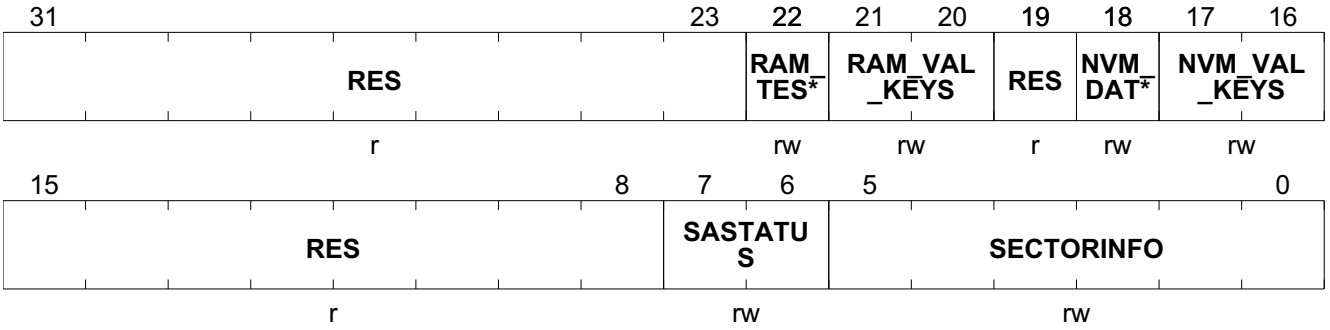
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0080_H$	RESET_TYPE_3		

Memory Status Register

The Memory Status Register can be used in two ways. Upon the completion of the Boot ROM startup following a reset, the register stores the NVM initialization status. Subsequently, the register can be used by the user code to store the status of the NVM program and emergency program operation status.

For Boot ROM to indicate NVM initialization status upon completion of startup:

SCU_MEMSTAT	Offset	Reset Value
Memory Status Register	$0DC_H$	see Table 123



Field	Bits	Type	Description
RES	31:23	r	**Reserved** Returns 0 if read; should be written with 0.
RAM_TEST_MODE	22	rw	**RAM Data Mode** 0_B **Full**, RAM test at cold reset executed on the whole RAM 1_B **1K**, RAM test at cold reset executed only on 1st kb of RAM
RAM_VAL_KEYS	21:20	rw	**RAM valid keys** Number of valid NVM switching keys
RES	19	r	**Reserved** Returns 0 if read; should be written with 0.
NVM_DATA_MODE	18	rw	**NVM Data Mode** 0_B **1**, 1 Non linearly mapped data sector 1_B **2**, 2 linearly mapped data sectors.
NVM_VAL_KEYS	17:16	rw	**NVM valid keys** Number of valid NVM switching keys
RES	15:8	r	**Reserved** Returns 0 if read; should be written with 0.

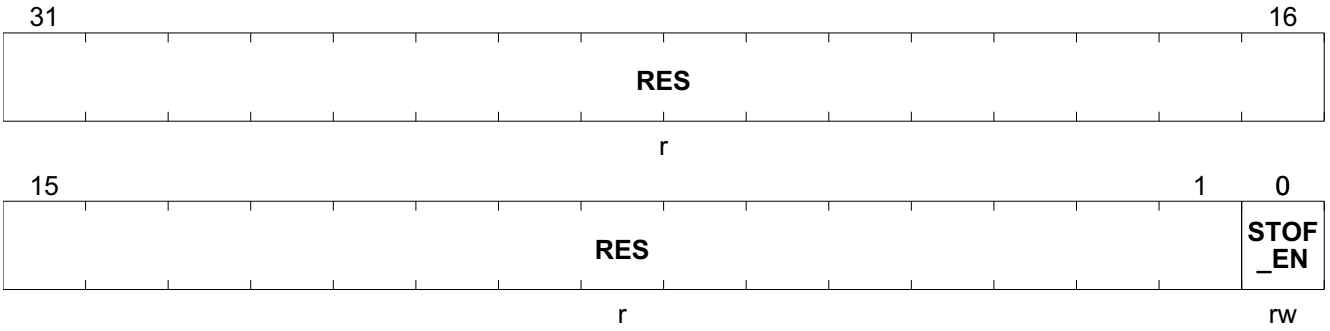
Field	Bits	Type	Description
SASTATUS	7:6	rw	**Service Algorithm Status** 00_B **Success_1**, Depending on SECTORINFO, there are two possible outcomes: For SECTORINFO = 00_H, NVM initialization is successful and no SA is executed. For SECTORINFO = values other than 00_H, SA execution is successful and only one map error is fixed. 01_B **Success_2**, SA execution is successful. More than one mapping error is fixed. 10_B **Error_1**, SA execution is not successful. Map error exists 11_B **Error_2**, NVM initialization failed, SA called but no page to be repaired has been found. Soft error present.
SECTORINFO	5:0	rw	**Sector Information** 01_H to 18_H, which represent the different sector addresses. For values not within this range, the data will be considered invalid. Once the SA has been executed, regardless of the execution status, the last accessed sector information will be stored here.

Table 123 RESET of SCU_MEMSTAT

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Stack Overflow Control Register

SCU_STACK_OVF_CTRL	Offset	Reset Value
Stack Overflow Control Register	144_H	see Table 124



Field	Bits	Type	Description
RES	31:1	r	**Reserved** Returns 0 if read; should be written with 0.
STOF_EN	0	rw	**Stack Overflow Enable** 0_B **Disable**, stack overflow detection disabled. 1_B **Enable**, stack overflow detection enabled

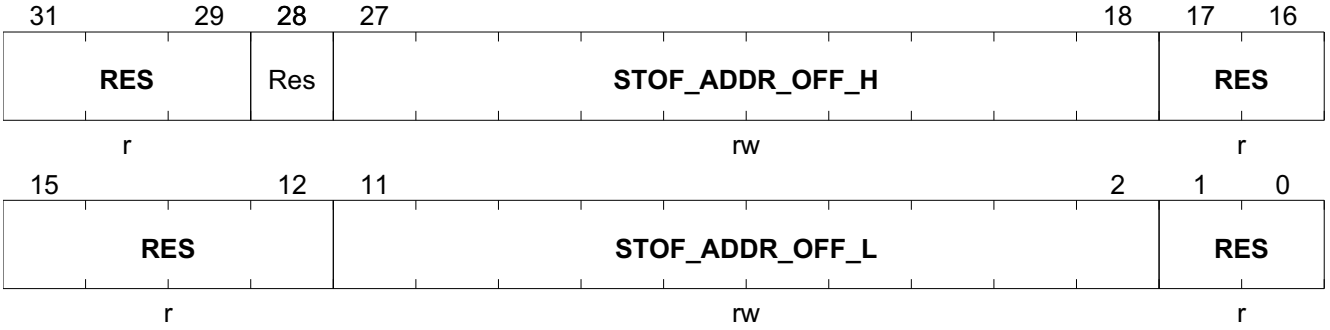
Table 124 RESET of SCU_STACK_OVF_CTRL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000_H	RESET_TYPE_3		

Stack Overflow Address Register

The **SCU_STACK_OVF_ADDR** defines an address region inside the RAM address range which is monitored for stack overflow protection purposes. In case there is an read/write access detected in the specified region an a stack overflow NMI is generated (FSTOFNMI).

SCU_STACK_OVF_ADDR	Offset	Reset Value
Stack Overflow Address Register	148_H	see Table 125



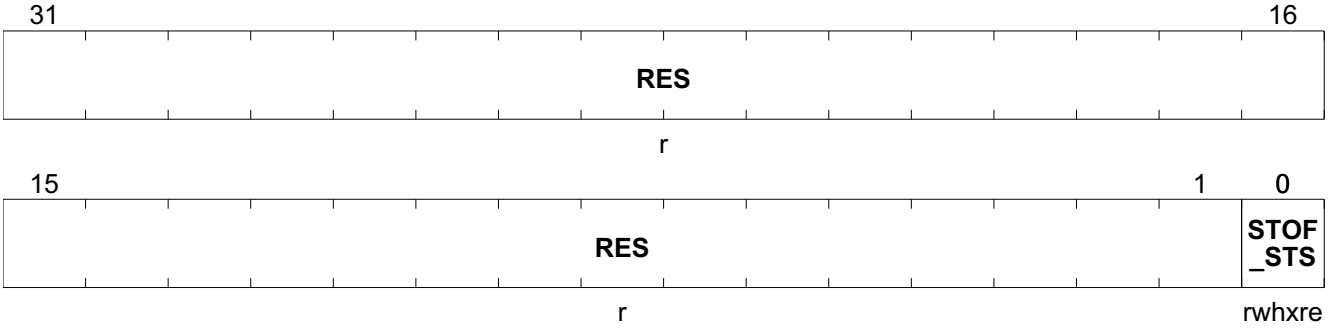
Field	Bits	Type	Description
RES	31:29	r	**Reserved** Returns 0 if read; should be written with 0.
STOF_ADDR_OFF_H	27:18	rw	**Stack Overflow High Address Offset** Defines the higher RAM address offset boundary for the stack overflow protection. Note: It defines Bits(11:2) of the higher RAM address boundary, Bits (1:0) of the higher RAM address boundary are not writeable and fixed to “00”. Therefore only Word aligned addresses are supported.
RES	17:16	r	**Reserved** Returns 0 if read; should be written with 0.
RES	15:12	r	**Reserved** Returns 0 if read; should be written with 0.
STOF_ADDR_OFF_L	11:2	rw	**Stack Overflow Low Address Offset** Defines the lower RAM address offset boundary for the stack overflow protection Note: It defines Bits(11:2) of the lower RAM address boundary, Bits (1:0) of the lower RAM address boundary are not writeable and fixed to “00”. Therefore only Word aligned addresses are supported.
RES	1:0	r	**Reserved** Returns 0 if read; should be written with 0.

Table 125 RESET of SCU_STACK_OVF_ADDR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		

Stack Overflow Status Register

SCU_STACK_OVF_STS	Offset	Reset Value
Stack Overflow Status Register	$14C_H$	see Table 126



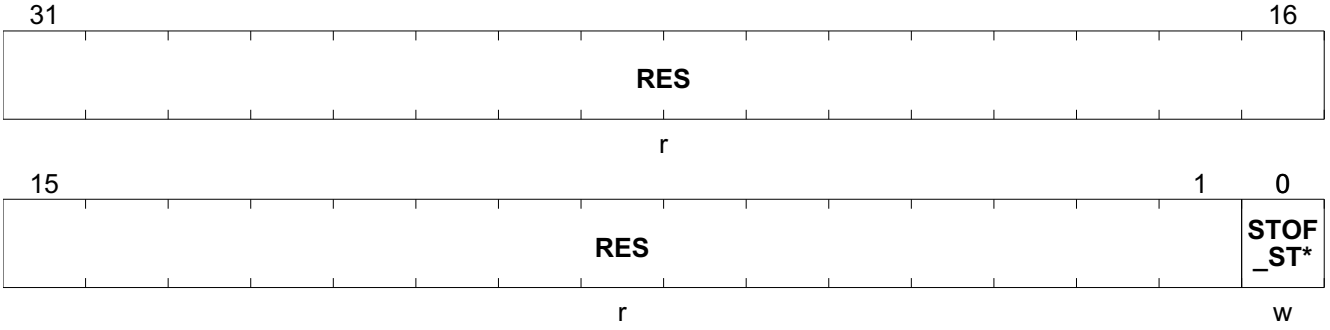
Field	Bits	Type	Description
RES	31:1	r	**Reserved** Returns 0 if read; should be written with 0.
STOF_STS	0	rwhxre	**Stack Overflow Status** 0_B **No Error**, No stack overflow detected. 1_B **Error**, stack overflow detected.

Table 126 RESET of SCU_STACK_OVF_STS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		

Stack Overflow Status Clear Register

SCU_STACK_OVFCLR	Offset	Reset Value
Stack Overflow Status Clear Register	$12C_H$	see Table 127



Field	Bits	Type	Description
RES	31:1	r	**Reserved** Returns 0 if read; should be written with 0.
STOF_STSC	0	w	**Clear Stack Overflow Status** 0_B **Not Cleared**, stack overflow not cleared. 1_B **Cleared**, stack overflow cleared.

Table 127 RESET of SCU_STACK_OVFCLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

8 System Control Unit - Power Modules (SCU-PM)

8.1 Features

- Clock Watchdog Unit (CWU): supervision of all power modules relevant clocks with NMI signalling.
- Interrupt Control Unit (ICU): all system relevant interrupt flags and status flags.
- Power Control Unit (PCU): takes over control when device enters and exits Sleep and Stop Mode.
- External Watchdog (WDT1): independent system watchdog to monitor system activity

8.2 Introduction

8.2.1 Block Diagram

The System Control Unit of the power modules consists of the sub-modules in the figure shown below:

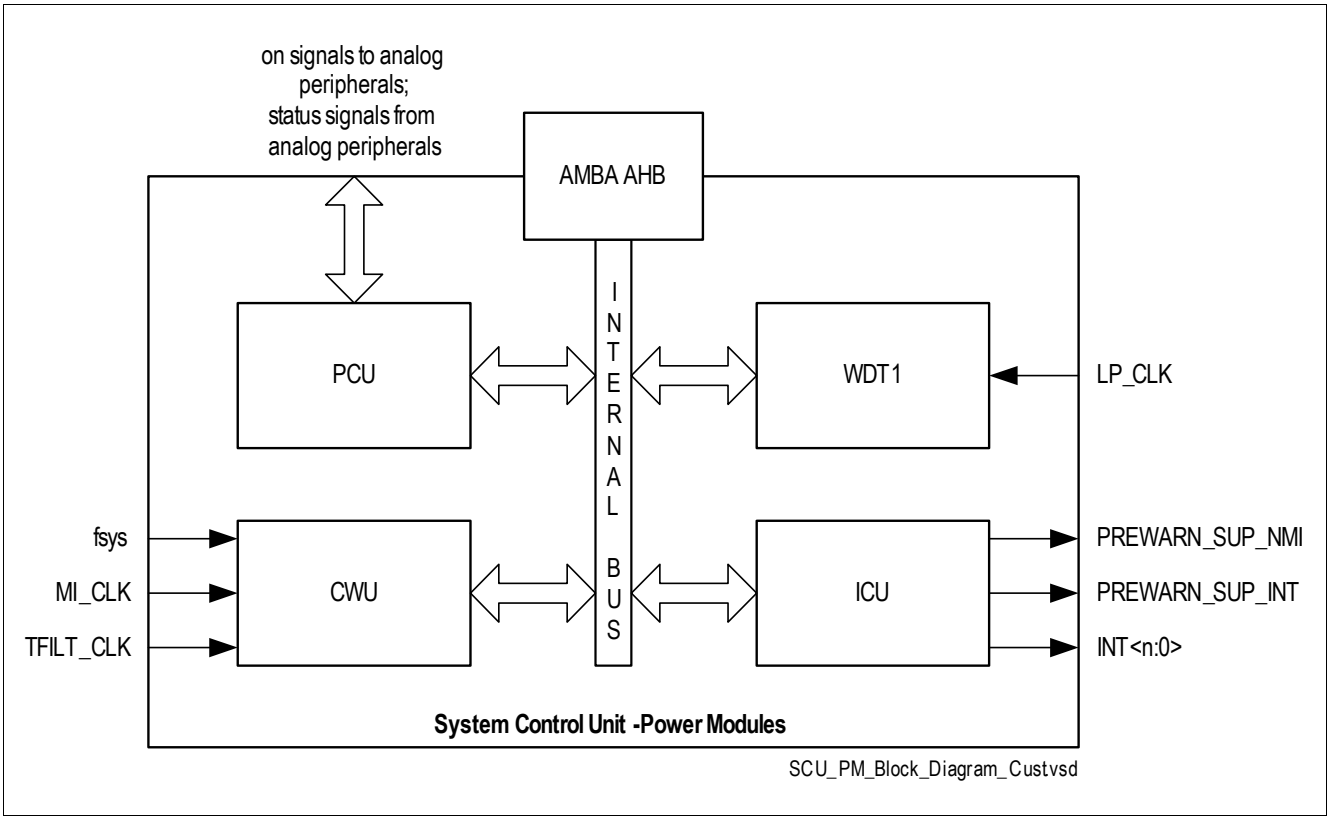


Figure 38 Block diagram of System Control Unit - Power Modules

IO description of SCU_PM:

- CWU:
  - check of f_{sys} = system frequency: output of PLL
  - check of MI_CLK = measurement interface clock (analog clock): derived out of f_{sys} by division factors 1/2/3/4
  - check of TFILT_CLK = clock used for digital filters: derived out of f_{sys} by configurable division factors

- ICU:
  - PREWARN_SUP_NMI = generation of Prewarn-Supply NMI
  - PREWARN_CLK_INT = generation of Prewarn-Clock Watchdog NMI
  - INT = generation of MISC interrupts

8.3 Clock Watchdog Unit (CWU)

There are two clock watchdogs available. One main purpose of them, is to monitor the derived switched capacitor clocks, which are used for analog module operation. If the clocks are not in the required range, a proper functionality of those modules is not given.

The following chapter describes the functionality and the configuration possibilities of these clock watchdogs.

8.3.1 Fail Safe Functionality of Clock Generation Unit (Clock Watchdog)

The Clock Generation Unit provides also fail safe functionalities, which are related to the input clock, the generated clocks and the clock settings. Those are:

- **MI_CLK** and **TFILT_CLK** are out of Range: MI Clock settings for f_{sys}, MI_CLK and TFILT_CLK Clock settings are out of required range and as a result the analog functionalities cannot be guaranteed. This failure triggers the clock watchdog NMI. The current status can be seen in the corresponding registers APCLK1 (in SCU) for the **MI_CLK** and APCLK2 (in SCU) for the **TFILT_CLK**.
- **Loss of clock:** When there is a loss of clock in the system, there is no possibility for the software to react upon this situation, like to enter a fail safe mode or switch to another backup clock source. For this purpose there is a clock watchdog implemented in the system which monitors the f_{sys} and in case of this emergency situation, disables all critical system functions, which are:
  - High Side
  - LIN

As shown in **Figure 39** all analog clocks are derived from **MI_CLK**. This clock structure requires to place a monitor on this clock, because f_{sys} and therefore **MI_CLK** are adjustable in a wide range (see also Chapter **System Control Unit - CGU**). As an important clock, also the TFILT_CLK is monitored by a clock watchdog. This clock watchdogs have an adjustable lower and upper limits including hysteresis. The placement of the clock watchdogs in the clock structure is sketched below:

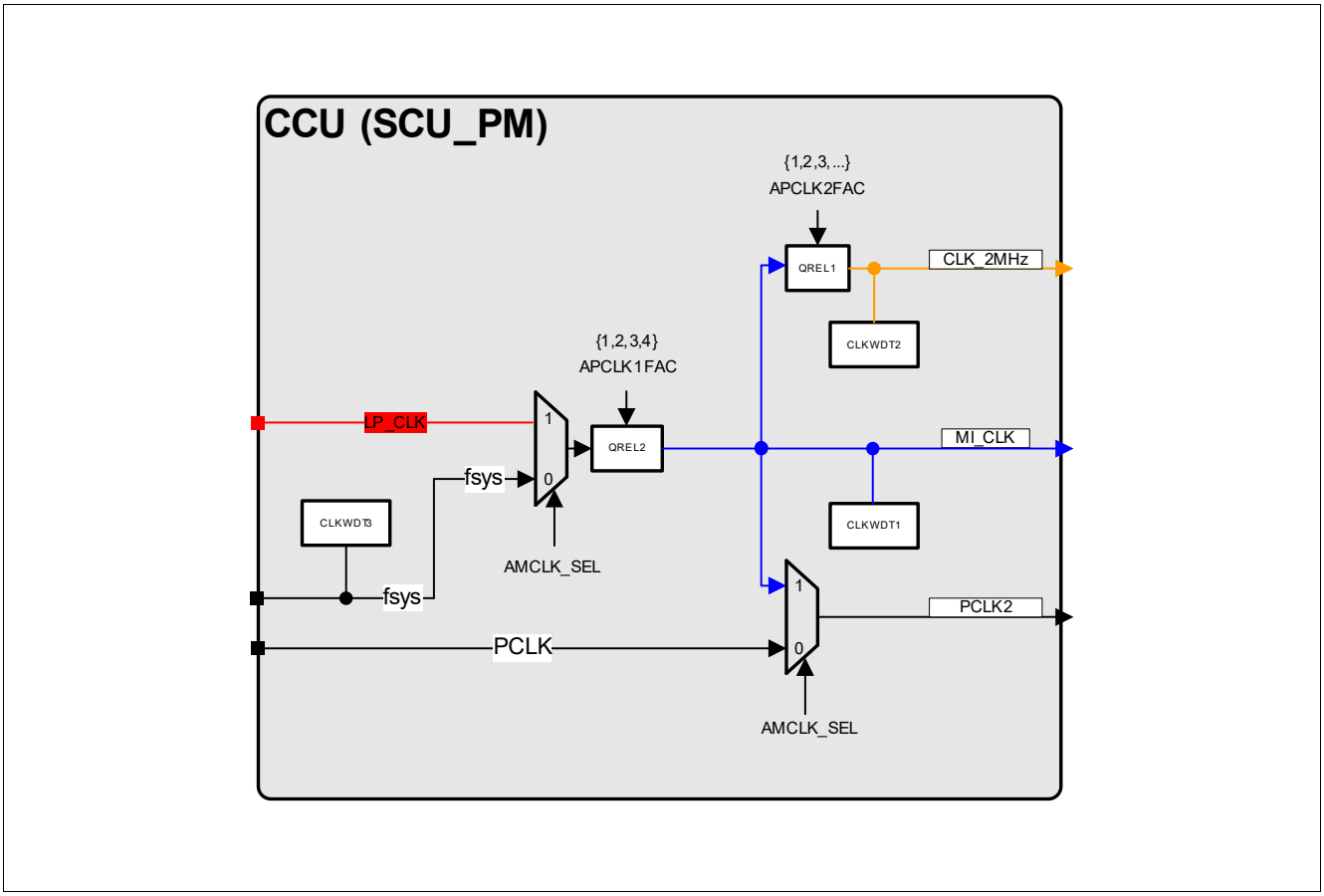


Figure 39 Block diagram of CGU including Clock Watchdogs

8.3.1.1 Functional Description of Clock Watchdog Module

The clock watchdog module consists of a counter. This counter monitors the number of system clocks within a defined time window. The duration of the time window is defined by a clock (**LP_CLK**), which is independent from the monitored system clock (**MI_CLK**). If the required number of clock cycles is not reached within this time window an clock watchdog NMI will be issued.

In case the clock watchdog NMI will be issued, indicating that the clock is not within the required frequency range, then the user has different options to overcome this situation:

- stay on mi_clk but reconfigure PLL to re-gain the required clock frequency. This would be the most time consuming measure to avoid emergency shutdown of the above listed modules.
- switch to divider factors 2, 3 and 4 to try to come back to specified frequency range.
- switch to LP_CLK, which also can be divided by factor 2, 3 and 4. This is the fastest option which allows the user to operate with a well defined backup clock rate. After this has been done the user can start investigating the root cause of the issued clock watchdog NMI, while operating on **LP_CLK**.

The register chapter below includes all necessary flags for setting up the analog module clock and monitoring its status during operation.

The events of the clock watchdog are hold in APCLK1.APCLK1STS for mi_clk and APCLK2.APCLK2STS for tfilt_clk.

The coding is

00: No event

01: early warning frequency too high 10: early warning frequency too low

11: fail(long window) frequency too low

The ouput clkwdt_fail_o is generated out of APCLK1.APCLK1STS or APCLK2.APCLK2STS. It indicates a fail frequency(too low in a long window) of i_clk or tfilt_clk.

(clkwdt_fail_o <= APCLK1.APCLK1STS = “11” or APCLK2.APCLK2STS = “11”).

clkwdt_fail_o is used for shutdown of analog module and for the PMU reset generation.

8.3.2 Clock Generation Unit Register

The analog module clock generation unit is fully controllable by the register described in this chapter.

Table 128 shows the module base addresses.

Table 128 Register Address Space

Module	Base Address	End Address	Note
SCUPM	50006000_H	$50006FFF_H$	SCU_PM

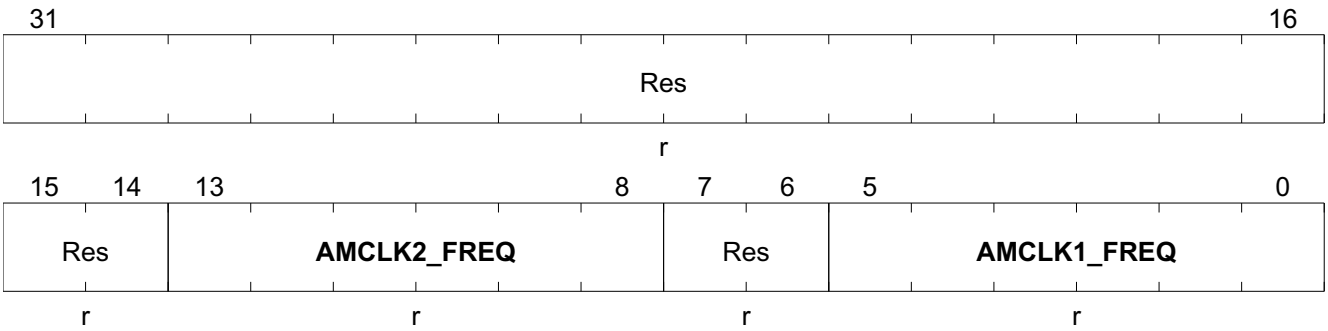
Table 129 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Clock Generation Unit Register,			
SCUPM_AMCLK_FREQ_STS	Analog Module Clock Frequency Status Register	00_H	00000000_H
SCUPM_AMCLK_CTRL	Analog Module Clock Control Register	04_H	$0000\ 0001_H$
SCUPM_AMCLK_TH_HYS	Analog Module Clock Limit Register	$0C_H$	$D4E1\ 94B3_H$
SCUPM_STCALIB	System Tick Calibration Register	$6C_H$	$0000\ 0000_H$

The registers are addressed wordwise.

Analog Module Clock Frequency Status Register

SCUPM_AMCLK_FREQ_STS	Offset	Reset Value
Analog Module Clock Frequency Status Register	00_H	see Table 130



Field	Bits	Type	Description
Res	31:16	r	**Reserved** Always read as 0
Res	15:14	r	**Reserved** Always read as 0

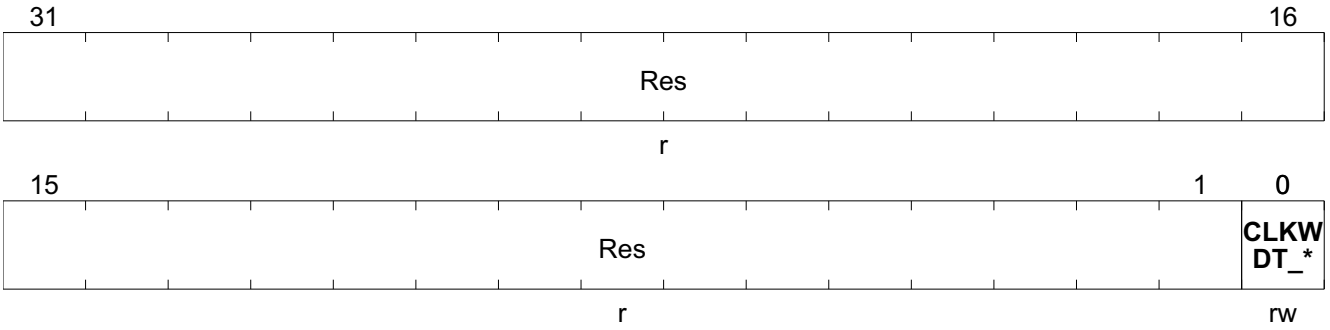
Field	Bits	Type	Description
AMCLK2_FREQ	13:8	r	**Current frequency of Analog Module Clock 2 (TFILT_CLK)** 0.09375 Mhz * AMCLK2_FREQ
Res	7:6	r	**Reserved** Always read as 0
AMCLK1_FREQ	5:0	r	**Current frequency of Analog Module Clock System Clock (MI_CLK)** 0.75 Mhz * AMCLK1_FREQ

Table 130 RESET of SCUPM_AMCLK_FREQ_STS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		

Analog Module Clock Control Register

SCUPM_AMCLK_CTRL	Offset	Reset Value
Analog Module Clock Control Register	04_H	see Table 131



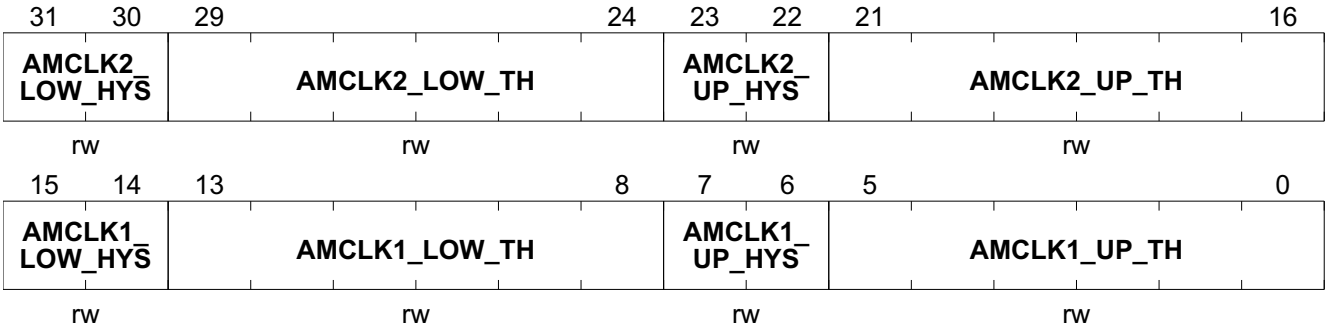
Field	Bits	Type	Description
Res	31:1	r	**Reserved** Always read as 0
CLKWDT_PD_N	0	rw	**Clock Watchdog Powerdown** 0_B **DISABLE**, Clock Watchdog disabled 1_B **ENABLE**, Clock Watchdog enabled

Table 131 RESET of SCUPM_AMCLK_CTRL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000001_H	RESET_TYPE_4		

Analog Module Clock Limit Register

SCUPM_AMCLK_TH_HYS	Offset	Reset Value
Analog Module Clock Limit Register	$0C_H$	see Table 132



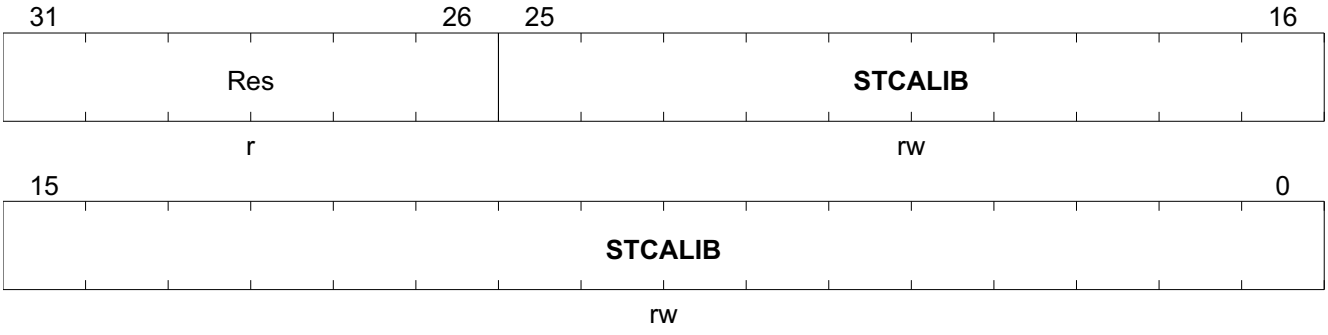
Field	Bits	Type	Description
AMCLK2_LOW_HYS	31:30	rw	**Analog Module Clock 2 (TFILT_CLK) Lower Hysteresis**
AMCLK2_LOW_TH	29:24	rw	**Analog Module Clock 2 (TFILT_CLK) Lower Limit Threshold** 0.09375 Mhz * AMCLK2_LOW_TH
AMCLK2_UP_HYS	23:22	rw	**Analog Module Clock 2 (TFILT_CLK) Upper Hysteresis**
AMCLK2_UP_TH	21:16	rw	**Analog Module Clock 2 (TFILT_CLK) Upper Limit Threshold** 0.09375 Mhz * AMCLK2_UP_TH
AMCLK1_LOW_HYS	15:14	rw	**Analog Module Clock 1 (MI_CLK) Lower Hysteresis**
AMCLK1_LOW_TH	13:8	rw	**Analog Module Clock 1 (MI_CLK) Lower Limit Threshold** 0.75 Mhz * AMCLK1_LOW_TH
AMCLK1_UP_HYS	7:6	rw	**Analog Module Clock 1 (MI_CLK) Upper Hysteresis**
AMCLK1_UP_TH	5:0	rw	**Analog Module Clock 1 (MI_CLK) Upper Limit Threshold** 0.75 Mhz * AMCLK1_UP_TH

Table 132 RESET of SCUPM_AMCLK_TH_HYS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$D4E194B3_H$	RESET_TYPE_4		
VARIANT	$D4E194B3_H$	VARIANT		

System Tick Calibration Register

SCUPM_STCALIB	Offset	Reset Value
System Tick Calibration Register	$6C_H$	see Table 133



Field	Bits	Type	Description
Res	31:26	r	**Reserved** Always read as 0
STCALIB	25:0	rw	**System Tick Calibration** [25]: Noref [24] Skew [23:0] Reload value to use for 10ms (100 Hz) timing STCALIB[23:0] = HCLK (in Hz) / 100 Hz - 1, e.g. 0x7A11F

Table 133 RESET of SCUPM_STCALIB

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		

8.4 Interrupt Control Unit (ICU)

The Subblock Interrupt Control Unit (ICU) of the System Control Unit - Power Modules (SCU_PM) is responsible for controlling and generating all analog peripheral relevant interrupts. Those analog interrupts are presented to the NVIC nodes 13-24 and NMI. Those are:

- **PREWARN_SUP_NMI:** combines all supply relevant interrupts to NMI.
- **Analog Module Interrupts:** combines all analog modules related interrupts.

The following two chapters describe the structure of the interrupt nodes.

8.4.1 Structure of PREWARN_SUP_NMI

This interrupt groups all system supply relevant interrupts. They can be divided into two groups:

- **voltages monitored by the Measurement Unit and 10 Bit ADC.** The supply voltages VS, VBAT_SENSE, VDDP and VDDC are monitored by the Measurement Unit and the 10 Bit ADC module. The Measurement Unit can be considered as an independent monitoring instance for external supply voltages and internal voltages generated by PMU. This monitoring is done with an independent reference and supply voltage to ensure fail safe operation.
- **voltages monitored by measurement functions of the PMU:** The PMU itself is checking its output voltages. Here failures due to undervoltage (overload), overvoltage and overcurrent are detected.

The **Figure 40** shows the structure of the **PREWARN_SUP**:

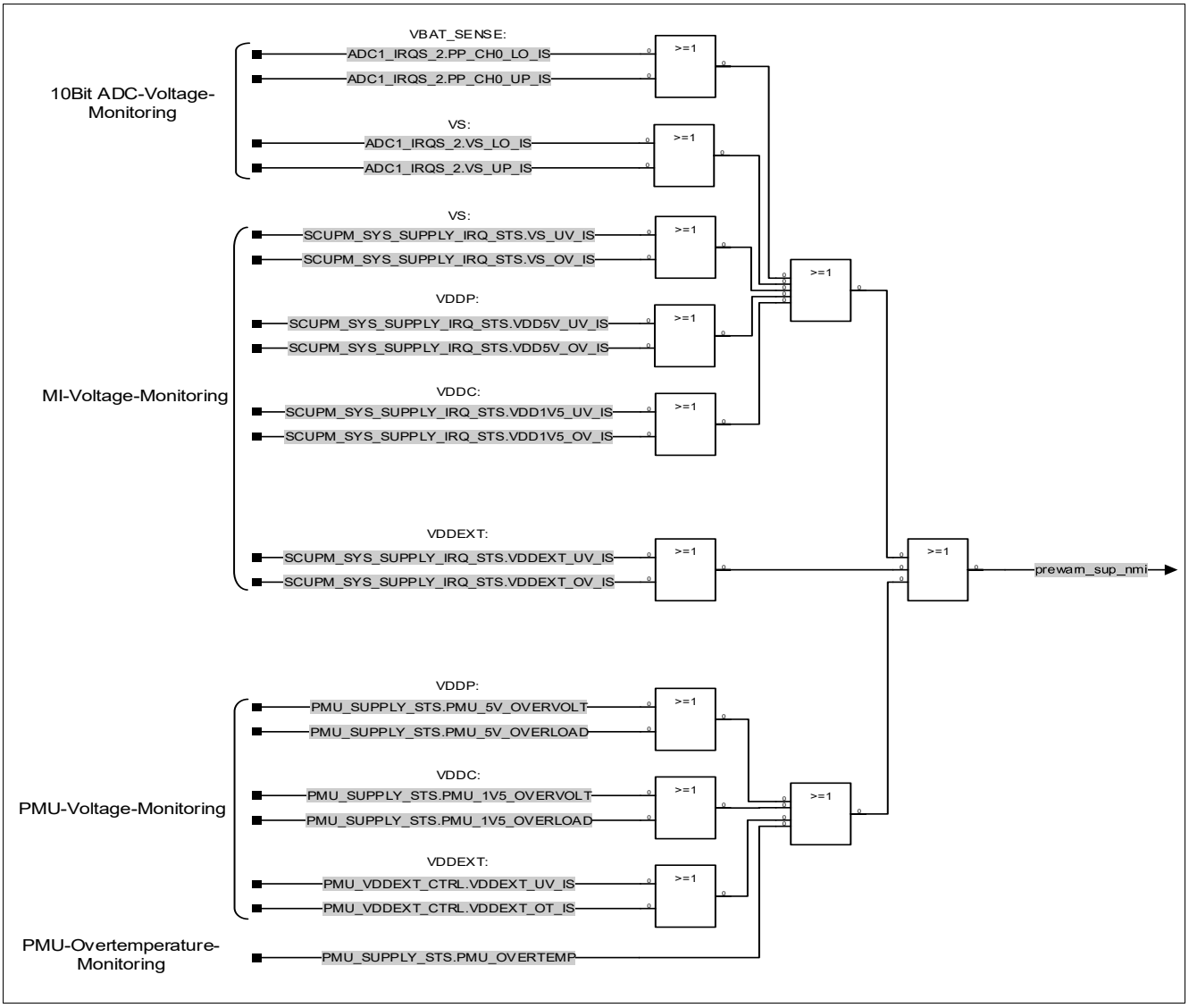


Figure 40 Structure of PREWARN_SUP

All PREWARN_SUP related flags are grouped in register **SCUPM_SYS_SUPPLY_IRQ_STS**. All measurement interface related flags are edge triggered. Therefore each IRQ_STS register has also an STS register where the current supply status can be monitored.

8.4.2 Interrupt Control Unit Status Register

All analog modules interrupt functionality is described in this chapter.

Table 134 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Interrupt Control Unit Status Register, Interrupt Control Unit Status Overview Register			
SCUPM_SYS_IS	System Interrupt Status Register	18_H	0000 0000$_H$
SCUPM_SYS_SUPPLY_IRQ_STS	System Supply Interrupt Status Register	$1C_H$	0000 0000$_H$
Interrupt Control Unit Status Register, Interrupt Control Unit - Interrupt Clear Register			
SCUPM_SYS_ISCLR	System Interrupt Status Clear Register	14_H	0000 0000$_H$
SCUPM_SYS_SUPPLY_IRQ_CLR	System Supply Interrupt Status Clear Register	24_H	0000 0000$_H$
Interrupt Control Unit Status Register, Interrupt Control Unit - Interrupt Enable Register			
SCUPM_SYS_SUPPLY_IRQ_CTRL	System Supply Interrupt Control Register	20_H	0000 00FF$_H$
SCUPM_SYS_IRQ_CTRL	System Interrupt Control Register	28_H	0000 0000$_H$

The registers are addressed wordwise.

8.4.2.1 Interrupt Control Unit Status Overview Register

Due to the large variety of diagnosis possibilities of TLE985xQX, the system offers several overview registers, to help the user finding the right source of interrupt. Those registers are described in this sub-chapter.

Overview Register, Switches Interrupt Status Register and System Supply Interrupt Status Register

- **SCUPM_SYS_SUPPLY_IRQ_STS**: Flags for Under- and Overvoltage detection for all system relevant supplies. These Interrupts are edge triggered Interrupts.
- **SCUPM_SYS_IS**: Interrupts for Analog Modules.

System Interrupt Status Register

SCUPM_SYS_IS	Offset	Reset Value
System Interrupt Status Register	18_H	see Table 135

31	30	29	28	27	26	25	24	23	20	19	18	17	16
Res	SYS_SUP*	VREF1V2*	VREF1V2*	Res		SYS_OT_*	SYS_OTW*	Res		HS_FAIL*	DRV_FAI*	CP_FAIL*	LIN_FAI*
r	r	rwhxr	rwhxr	r		rwhxr	rwhxr	r		r	r	r	r

15	14	13	12	11	10	9	8	7	4	3	2	1	0
Res	SYS_SUP*	VREF1V2*	VREF1V2*	Res	CLKWDT_*	SYS_OT_*	SYS_OTW*	Res		HS_FAIL*	DRV_FAI*	CP_FAIL*	LIN_FAI*
r	r	rwhxr	rwhxr	r	r	rwhxre	rwhxre	r		r	r	r	r

Field	Bits	Type	Description
Res	31	r	**Reserved** Always read as 0
SYS_SUPPLY_STS	30	r	**System Supply Status** *Note: This flag is an OR combination of all Supply Status Flags of the **SCUPM_SYS_SUPPLY_IRQ_STS** register* 0_B **OK**, no status set 1_B **FAIL**, at least one status set
VREF1V2_OV_STS	29	rwhxr	**8 Bit ADC2 Reference Overvoltage (ADC2, Channel 5) interrupt status** 0_B **OK**, no interrupt status set 1_B **FAIL**, at least one interrupt status set
VREF1V2_UV_STS	28	rwhxr	**8 Bit ADC2 Reference Undervoltage (ADC2, Channel 5) interrstatus** 0_B **OK**, no interrupt status set 1_B **FAIL**, at least one interrupt status set
Res	27:26	r	**Reserved** Always read as 0
SYS_OT_STS	25	rwhxr	**System Overtemperature Shutdown (ADC2, Channel 8) status** 0_B **OK**, no status set 1_B **FAIL**, at least one status set
SYS_OTWARN_STS	24	rwhxr	**System Overtemperature Prewarning (ADC2, Channel 8) status** 0_B **OK**, no status set 1_B **FAIL**, at least one status set
Res	23:20	r	**Reserved** Always read as 0

Field	Bits	Type	Description
HS_FAIL_STS	19	r	**High Side Driver Fail Status** *Note: This flag is an OR combination of HS1_OT_STS and HS1_OL_STS* 0_B **OK**, no status set 1_B **FAIL**, at least one status set
DRV_FAIL_STS	18	r	**Gate Driver Fail Status** *Note: This flag is an OR combination of gate driver status SFRs(STS)* 0_B **OK**, no status set 1_B **FAIL**, at least one status set
CP_FAIL_STS	17	r	**Charge Pump Fail Status** *Note: This flag is an OR combination of all charge pump status SFRs(STS)* 0_B **OK**, no status set 1_B **FAIL**, at least one status set
LIN_FAIL_STS	16	r	**LIN Fail Status** *Note: This flag is the LIN_OT_STS* 0_B **OK**, no status set 1_B **FAIL**, at least one status set
Res	15	r	**Reserved** Always read as 0
SYS_SUPPLY_IS	14	r	**System Supply Interrupt Status** *Note: This flag is an OR combination of all Supply Interrupt Status Flags of the **SCUPM_SYS_SUPPLY_IRQ_STS** register* 0_B **OK**, no interrupt status set 1_B **FAIL**, at least one interrupt status set
VREF1V2_OV_IS	13	rwhxr	**8 Bit ADC2 Reference Overvoltage (ADC2, Channel 5) interrupt status** 0_B **OK**, no interrupt status set 1_B **FAIL**, at least one interrupt status set
VREF1V2_UV_IS	12	rwhxr	**8 Bit ADC2 Reference Undervoltage (ADC2, Channel 5) interrupt status** 0_B **OK**, no interrupt status set 1_B **FAIL**, at least one interrupt status set
Res	11	r	**Reserved** Always read as 0

Field	Bits	Type	Description
CLKWDT_IS	10	r	**Clock Watchdog Interrupt Status** 0_B **OK**, no interrupt status set 1_B **FAIL**, at least one interrupt status set
SYS_OT_IS	9	rwhxre	**System Overtemperature Shutdown (ADC2, Channel 8) interrupt status** 0_B **OK**, no interrupt status set 1_B **FAIL**, at least one interrupt status set
SYS_OTWARN_IS	8	rwhxre	**System Overtemperature Prewarning (ADC2, Channel 8) interrupt status** 0_B **OK**, no interrupt status set 1_B **FAIL**, at least one interrupt status set
Res	7:4	r	**Reserved** Always read as 0
HS_FAIL_IS	3	r	**High Side Driver Fail Interrupt Status** *Note: This flag is an OR combination of HS1_OC_IS, HS1_OT_IS and HS1_OL_IS* 0_B **OK**, no status set 1_B **FAIL**, at least one status set
DRV_FAIL_IS	2	r	**Gate Driver Fail Interrupt Status** *Note: This flag is an OR combination of gate driver interrupts* 0_B **OK**, no status set 1_B **FAIL**, at least one status set
CP_FAIL_IS	1	r	**Charge Pump Fail Interrupt Status** *Note: This flag is an OR combination of charge pump interrupts* 0_B **OK**, no status set 1_B **FAIL**, at least one status set
LIN_FAIL_IS	0	r	**LIN Fail Interrupt Status** *Note: This flag is an OR combination of LIN_OC_IS and LIN_OT_IS* 0_B **OK**, no status set 1_B **FAIL**, at least one status set

Table 135 RESET of SCUPM_SYS_IS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		

System Supply Interrupt Status Register

SCUPM_SYS_SUPPLY_IRQ_STS	Offset	Reset Value
System Supply Interrupt Status Register	$1C_H$	see Table 136

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res	VDD1V5_*	Res	VDD5V_O*	VDDEXT_*	Res		VS_OV_S*	Res	VDD1V5_*	Res	VDD5V_U*	VDDEXT_*	Res		VS_UV_S*
r	rwhxr	r	rwhxr	rwhxr	r		rwhxr	r	rwhxr	r	rwhxr	rwhxr	r		rwhxr

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res	VDD1V5_*	Res	VDD5V_O*	VDDEXT_*	Res		VS_OV_IS	Res	VDD1V5_*	Res	VDD5V_U*	VDDEXT_*	Res		VS_UV_IS
r	rwhxre	r	rwhxre	rwhxre	r		rwhxre	r	rwhxre	r	rwhxre	rwhxre	r		rwhxre

Field	Bits	Type	Description
Res	31	r	**Reserved** Always read as 0
VDD1V5_OV_STS	30	rwhxr	**VDDC Overvoltage Status (ADC2 channel 6)** 0_B **No Overvoltage**, occurred 1_B **Overvoltage**, occurred
Res	29	r	**Reserved** Always read as 0
VDD5V_OV_STS	28	rwhxr	**VDDP Overvoltage Status (ADC2 channel 4)** 0_B **No Overvoltage**, occurred 1_B **Overvoltage**, occurred
VDDEXT_OV_STS	27	rwhxr	**VDDEXT Overvoltage Status (ADC2 channel 3)** 0_B **No Overvoltage**, occurred 1_B **Overvoltage**, occurred
Res	26:25	r	**Reserved** Always read as 0
VS_OV_STS	24	rwhxr	**VS Overvoltage Status (ADC2 channel 0)** 0_B **No Overvoltage**, occurred 1_B **Overvoltage**, occurred
Res	23	r	**Reserved** Always read as 0
VDD1V5_UV_STS	22	rwhxr	**VDDC Undervoltage Status (ADC2 channel 6)** 0_B **No Undervoltage**, occurred 1_B **Undervoltage**, occurred
Res	21	r	**Reserved** Always read as 0
VDD5V_UV_STS	20	rwhxr	**VDDP Undervoltage Status (ADC2 channel 4)** 0_B **No Undervoltage**, occurred 1_B **Undervoltage**, occurred

Field	Bits	Type	Description
VDDEXT_UV_STS	19	rwhxr	**VDDEXT Undervoltage Status (ADC2 channel 3)** 0_B **No Undervoltage**, occurred 1_B **Undervoltage**, occurred
Res	18:17	r	**Reserved** Always read as 0
VS_UV_STS	16	rwhxr	**VS Undervoltage Status (ADC2 channel 0)** 0_B **No Undervoltage**, occurred 1_B **Undervoltage**, occurred
Res	15	r	**Reserved** Always read as 0
VDD1V5_OV_IS	14	rwhxre	**VDDC Overvoltage Interrupt Status (ADC2 channel 6)** 0_B **No Overvoltage Interrupt**, occurred 1_B **Overvoltage Interrupt**, occurred
Res	13	r	**Reserved** Always read as 0
VDD5V_OV_IS	12	rwhxre	**VDDP Overvoltage Interrupt Status (ADC2 channel 4)** 0_B **No Overvoltage Interrupt**, occurred 1_B **Overvoltage Interrupt**, occurred
VDDEXT_OV_IS	11	rwhxre	**VDDEXT Overvoltage Interrupt Status (ADC2 channel 3)** 0_B **No Overvoltage Interrupt**, occurred 1_B **Overvoltage Interrupt**, occurred
Res	10:9	r	**Reserved** Always read as 0
VS_OV_IS	8	rwhxre	**VS Overvoltage Interrupt Status (ADC2 channel 0)** 0_B **No Overvoltage Interrupt**, occurred 1_B **Overvoltage Interrupt**, occurred
Res	7	r	**Reserved** Always read as 0
VDD1V5_UV_IS	6	rwhxre	**VDDC Undervoltage Interrupt Status (ADC2 channel 6)** 0_B **No Undervoltage Interrupt**, occurred 1_B **Undervoltage Interrupt**, occurred
Res	5	r	**Reserved** Always read as 0
VDD5V_UV_IS	4	rwhxre	**VDDP Undervoltage Interrupt Status (ADC2 channel 4)** 0_B **No Undervoltage Interrupt**, occurred 1_B **Undervoltage Interrupt**, occurred
VDDEXT_UV_IS	3	rwhxre	**VDDEXT Undervoltage Interrupt Status (ADC2 channel 3)** 0_B **No Undervoltage Interrupt**, occurred 1_B **Undervoltage Interrupt**, occurred

Field	Bits	Type	Description
Res	2:1	r	**Reserved** Always read as 0
VS_UV_IS	0	rwhxre	**VS Undervoltage Interrupt Status (ADC2 channel 0)** 0_B **No Undervoltage Interrupt**, occurred 1_B **Undervoltage Interrupt**, occurred

Table 136 RESET of SCUPM_SYS_SUPPLY_IRQ_STS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		

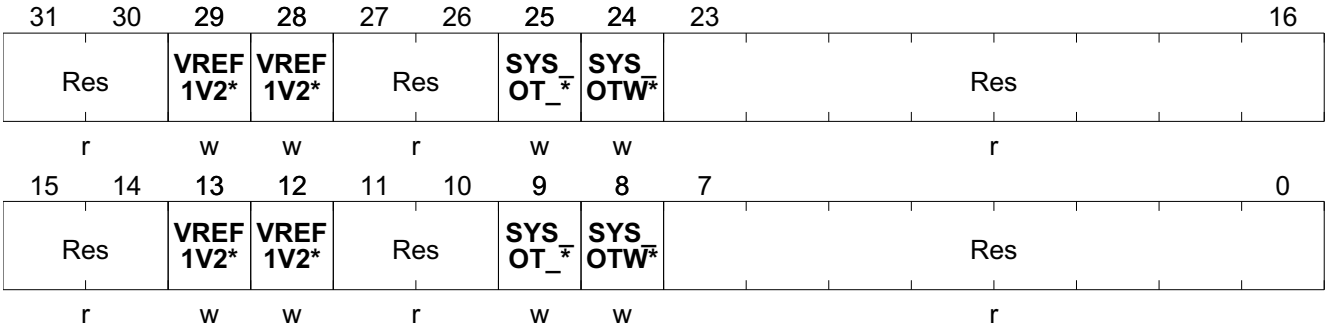
8.4.2.2 Interrupt Control Unit - Interrupt Clear Register

The Analog Module Interrupts can be cleared by their corresponding enable bits which are located in Registers:

- **SCUPM_SYS_SUPPLY_IRQ_CLR**: Clear of Interrupts for Under- and Overvoltage detection for all system relevant supplies. These interrupts are edge triggered interrupts to reduce interrupt load of the µC.
- **SCUPM_SYS_ISCLR**: Clear of interrupts related to Analog Modules.

System Interrupt Status Clear Register

SCUPM_SYS_ISCLR	Offset	Reset Value
System Interrupt Status Clear Register	14_H	see Table 137



Field	Bits	Type	Description
Res	31:30	r	**Reserved** Always read as 0
VREF1V2_OV_SC	29	w	**8 Bit ADC2 Reference Overvoltage Status Clear** 0_B **No clear**, 1_B **Clear**,
VREF1V2_UV_SC	28	w	**8 Bit ADC2 Reference Undervoltage Status Clear** 0_B **No clear**, 1_B **Clear**,
Res	27:26	r	**Reserved** Always read as 0
SYS_OT_SC	25	w	**System Overtemperature Shutdown Status Clear** 0_B **No clear**, 1_B **Clear**,
SYS_OTWARN_SC	24	w	**System Overtemperature Prewarning Status Clear** 0_B **No clear**, 1_B **Clear**,
Res	23:14	r	**Reserved** Always read as 0

Field	Bits	Type	Description
VREF1V2_OV_ISC	13	w	**8 Bit ADC2 Reference Overvoltage Interrupt Status Clear** 0_B **No clear**, 1_B **Clear**,
VREF1V2_UV_ISC	12	w	**8 Bit ADC2 Reference Undervoltage Interrupt Status Clear** 0_B **No clear**, 1_B **Clear**,
Res	11:10	r	**Reserved** Always read as 0
SYS_OT_ISC	9	w	**System Overtemperature Shutdown Interrupt Status Clear** 0_B **No clear**, 1_B **Clear**,
SYS_OTWARN_ISC	8	w	**System Overtemperature Prewarning Interrupt Status Clear** 0_B **No clear**, 1_B **Clear**,
Res	7:0	r	**Reserved** Always read as 0

Table 137 RESET of SCUPM_SYS_ISCLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

System Supply Interrupt Status Clear Register

SCUPM_SYS_SUPPLY_IRQ_CLR	Offset	Reset Value
System Supply Interrupt Status Clear Register	24_H	see Table 138

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res	VDD1V5_*	Res	VDD5V_O*	VDDEXT_*	Res		VS_OV_SC	Res	VDD1V5_*	Res	VDD5V_U*	VDDEXT_*	Res		VS_UV_SC
r	w	r	w	w	r		w	r	w	r	w	w	r		w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res	VDD1V5_*	Res	VDD5V_O*	VDDEXT_*	Res		VS_OV_I*	Res	VDD1V5_*	Res	VDD5V_U*	VDDEXT_*	Res		VS_UV_I*
r	w	r	w	w	r		w	r	w	r	w	w	r		w

Field	Bits	Type	Description
Res	31	r	**Reserved** Always read as 0
VDD1V5_OV_SC	30	w	**VDDC Overvoltage Status clear** 0_B **No Clear** , 1_B **Clear** ,
Res	29	r	**Reserved** Always read as 0
VDD5V_OV_SC	28	w	**VDDP Overvoltage Status clear** 0_B **No Clear** , 1_B **Clear** ,
VDDEXT_OV_SC	27	w	**VDDEXT Overvoltage Status clear** 0_B **No Clear** , 1_B **Clear** ,
Res	26:25	r	**Reserved** Always read as 0
VS_OV_SC	24	w	**VS Overvoltage Status clear** 0_B **No Clear** , 1_B **Clear** ,
Res	23	r	**Reserved** Always read as 0
VDD1V5_UV_SC	22	w	**VDDC Undervoltage Status clear** 0_B **No Clear** , 1_B **Clear** ,
Res	21	r	**Reserved** Always read as 0
VDD5V_UV_SC	20	w	**VDDP Undervoltage Status clear** 0_B **No Clear** , 1_B **Clear** ,

Field	Bits	Type	Description
VDDEXT_UV_SC	19	w	**VDDEXT Undervoltage Status clear** 0_B **No Clear** , 1_B **Clear** ,
Res	18:17	r	**Reserved** Always read as 0
VS_UV_SC	16	w	**VS Undervoltage Status clear** 0_B **No Clear** , 1_B **Clear** ,
Res	15	r	**Reserved** Always read as 0
VDD1V5_OV_ISC	14	w	**VDDC Overvoltage Interrupt Status clear** 0_B **No Clear** , 1_B **Clear** ,
Res	13	r	**Reserved** Always read as 0
VDD5V_OV_ISC	12	w	**VDDP Overvoltage Interrupt Status clear** 0_B **No Clear** , 1_B **Clear** ,
VDDEXT_OV_ISC	11	w	**VDDEXT Overvoltage Interrupt Status clear** 0_B **No Clear** , 1_B **Clear** ,
Res	10:9	r	**Reserved** Always read as 0
VS_OV_ISC	8	w	**VS Overvoltage Interrupt Status clear** 0_B **No Clear** , 1_B **Clear** ,
Res	7	r	**Reserved** Always read as 0
VDD1V5_UV_ISC	6	w	**VDDC Undervoltage Interrupt Status clear** 0_B **No Clear** , 1_B **Clear** ,
Res	5	r	**Reserved** Always read as 0
VDD5V_UV_ISC	4	w	**VDDP Undervoltage Interrupt Status clear** 0_B **No Clear** , 1_B **Clear** ,
VDDEXT_UV_ISC	3	w	**VDDEXT Undervoltage Interrupt Status clear** 0_B **No Clear** , 1_B **Clear** ,
Res	2:1	r	**Reserved** Always read as 0
VS_UV_ISC	0	w	**VS Undervoltage Interrupt Status clear** 0_B **No Clear** , 1_B **Clear** ,

Table 138 RESET of SCUPM_SYS_SUPPLY_IRQ_CLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		

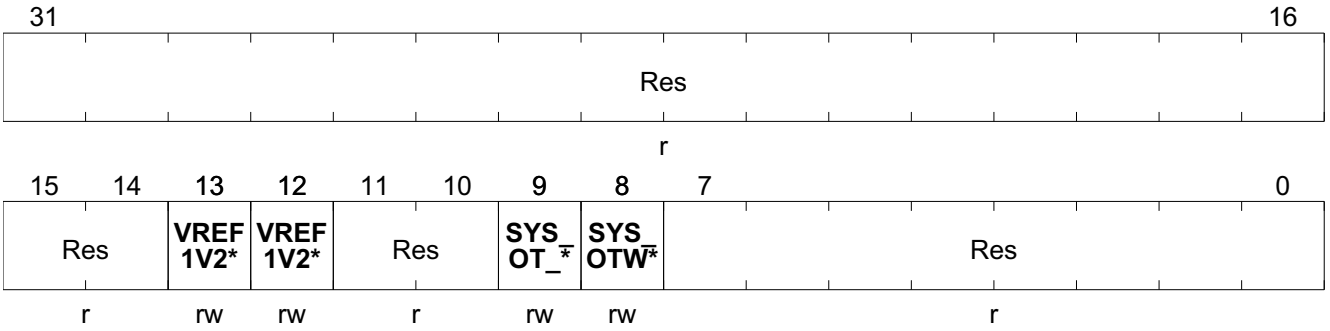
8.4.2.3 Interrupt Control Unit - Interrupt Enable Register

The Analog Module Interrupts can be enabled and disabled by there corresponding enable bits which are located in Registers:

- **SCUPM_SYS_SUPPLY_IRQ_CTRL**: Enable of Interrupts for Under- and Overvoltage detection for all system relevant supplies. These interrupts are edge triggered interrupts to reduce interrupt load of the µC.
- **SCUPM_SYS_IRQ_CTRL**: Enable of interrupts for Analog Modules.

System Interrupt Control Register

SCUPM_SYS_IRQ_CTRL	Offset	Reset Value
System Interrupt Control Register	28_H	see Table 139



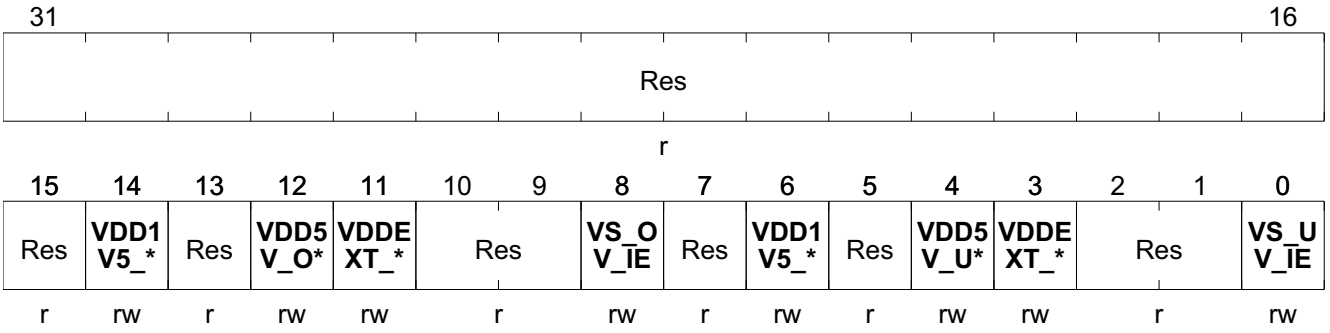
Field	Bits	Type	Description
Res	31:14	r	**Reserved** Always read as 0
VREF1V2_OV_IE	13	rw	**8 Bit ADC2 Reference Overvoltage Interrupt Enable** 0_B **Interrupt is disabled**, 1_B **Interrupt is enabled**,
VREF1V2_UV_IE	12	rw	**8 Bit ADC2 Reference Undervoltage Interrupt Enable** 0_B **Interrupt is disabled**, 1_B **Interrupt is enabled**,
Res	11:10	r	**Reserved** Always read as 0
SYS_OT_IE	9	rw	**System Overtemperature Shutdown Interrupt Enable** 0_B **Interrupt is disabled**, 1_B **Interrupt is enabled**,
SYS_OTWARN_IE	8	rw	**System Overtemperature Prewarning Interrupt Enable** 0_B **Interrupt is disabled**, 1_B **Interrupt is enabled**,
Res	7:0	r	**Reserved** Always read as 0

Table 139 RESET of SCUPM_SYS_IRQ_CTRL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		

System Supply Interrupt Control Register

SCUPM_SYS_SUPPLY_IRQ_CTRL	Offset	Reset Value
System Supply Interrupt Control Register	20_H	see Table 140



Field	Bits	Type	Description
Res	31:15	r	**Reserved** Always read as 0
VDD1V5_OV_IE	14	rw	**VDDC Overvoltage Interrupt Enable** 0_B **Disable**, Interrupt is disabled 1_B **Enable**, Interrupt is enabled
Res	13	r	**Reserved** Always read as 0
VDD5V_OV_IE	12	rw	**VDDP Overvoltage Interrupt Enable** 0_B **Disable**, Interrupt is disabled 1_B **Enable**, Interrupt is enabled
VDDEXT_OV_IE	11	rw	**VDDEXT Overvoltage Interrupt Enable** 0_B **Disable**, Interrupt is disabled 1_B **Enable**, Interrupt is enabled
Res	10:9	r	**Reserved** Always read as 0
VS_OV_IE	8	rw	**VS Overvoltage Interrupt Enable** 0_B **Disable**, Interrupt is disabled 1_B **Enable**, Interrupt is enabled
Res	7	r	**Reserved** Always read as 0
VDD1V5_UV_IE	6	rw	**VDDC Undervoltage Interrupt Enable** 0_B **Disable**, Interrupt is disabled 1_B **Enable**, Interrupt is enabled
Res	5	r	**Reserved** Always read as 0
VDD5V_UV_IE	4	rw	**VDDP Undervoltage Interrupt Enable** 0_B **Disable**, Interrupt is disabled 1_B **Enable**, Interrupt is enabled

Field	Bits	Type	Description
VDDEXT_UV_IE	3	rw	**VDDEXT Undervoltage Interrupt Enable** 0_B **Disable**, Interrupt is disabled 1_B **Enable**, Interrupt is enabled
Res	2:1	r	**Reserved** Always read as 0
VS_UV_IE	0	rw	**VS Undervoltage Interrupt Enable** 0_B **Disable**, Interrupt is disabled 1_B **Enable**, Interrupt is enabled

Table 140 RESET of SCUPM_SYS_SUPPLY_IRQ_CTRL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		

8.5 Power Control Unit for Power Modules (PCU_PM)

The chapter describes the implementation of the power modules state machine. This state machine is responsible for powering up and powering down the on-board power modules. It takes care about the interaction between the Measurement Unit and the modules which are evaluated by the Unit. The following modules are controlled by this statemachine:

Analog Modules controlled by Power Control Unit:

- Central Reference Voltage Generation
- Central Bias Current Generation
- 8 Bit ADC Core
- Supply Voltage Attenuators
- Monitoring Inputs Voltage Attenuators
- LIN Transceiver
- High Side Driver
- Gate Driver

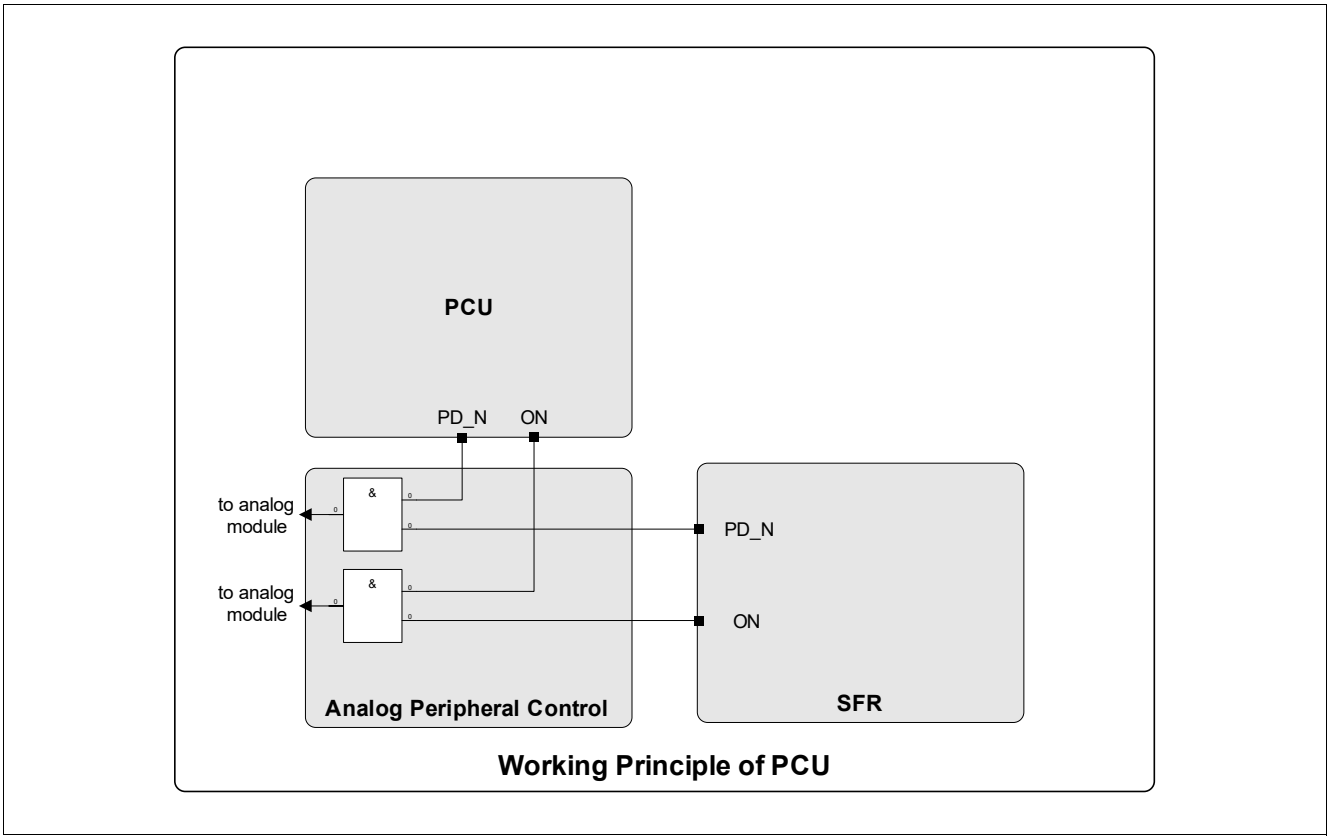


Figure 41 Function of AP_SUB_CTRL

If the device will power up the analog modules statemachine will startup all analog modules. First of all, the reference voltage will be enabled. After that the biasing module will be enabled. If this step is completed the analog modules will be enabled step by step. After this is done the measurement interface will start-up.

When leaving Stop Mode, this sequence restores the SFR register contents with the values written before entering Stop Mode.

The Sleep and Stop Mode entry is as well controlled by this state machine. This ensures a smooth shutdown of the modules avoiding disturbances (like load jumps) on the supplies.

The power control unit also handles system failures indicated by the analog measurement interface. They are:

System failures handled by SCU_PM:

- automatic shutdown of power modules in case of VS Overvoltage
- automatic shutdown of power modules in case of System Overtemperature
- automatic shutdown of power modules in case of loss of clock
- automatic shutdown of system in case of System Overtemperature
- automatic shutdown of system in case of internal supply fail
- automatic switch to receive only mode of LIN module in case of VS Undervoltage

How to configure this actions on the above described system failures will be described in the following chapters.

8.5.1 VS-Overvoltage System Shutdown

The PCU provides the possibility of an system shutdown in case of VS Overvoltage. This feature can be used to reduce power dissipation in case of an increased supply voltage VS. This feature can be enabled by bit **SYS_VS_OV_SLM_DIS**. **This bit is low active!**. When there is an overvoltage, the system will be set in System Shutdown and all Power Modules

LIN,

High Side,

Driver

are switched off automatically. The Power Modules will be switched on when the VS-Overvoltage condition is left again. The figure below shows the principle of the enable bit:

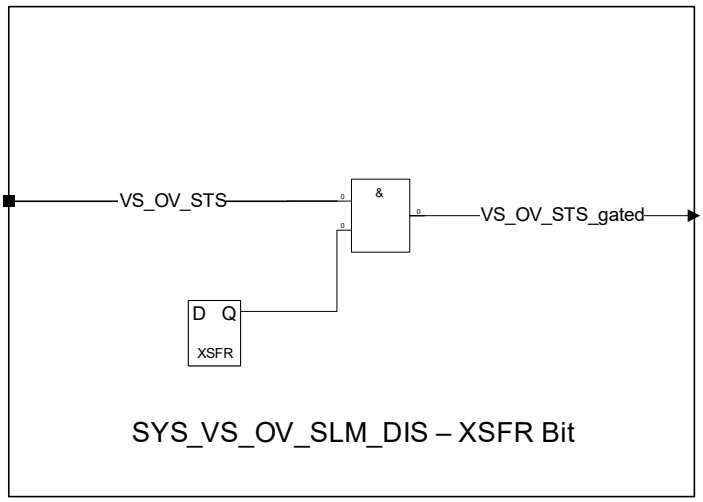


Figure 42 Implementation of Module Shutdown because of VS Overvoltage

8.5.2 Overtemperature System Shutdown

In case of overtemperature (T_j > 175 °C) the system will be set to Sleep Mode. This functionality is used to protect the system from thermal overstress. One possibility of avoiding this thermal shutdown is to stick to an emergency procedure, which helps to minimize the power dissipation in the system. This routine would require to shutdown all modules which have big contribution to power dissipation (e.g. Low Sides, High Sides). This procedure has to be implemented in user software. Another possibility is to use the implemented hardware shutdown procedure. This procedure can be activated by the flag **SYS_OTWARN_PS_DIS**. **This flag is low active!** When this flag is set all power dissipation contributors will be automatically shutdown.

- Main power dissipation contributors are:
  - High Side Driver
  - LIN

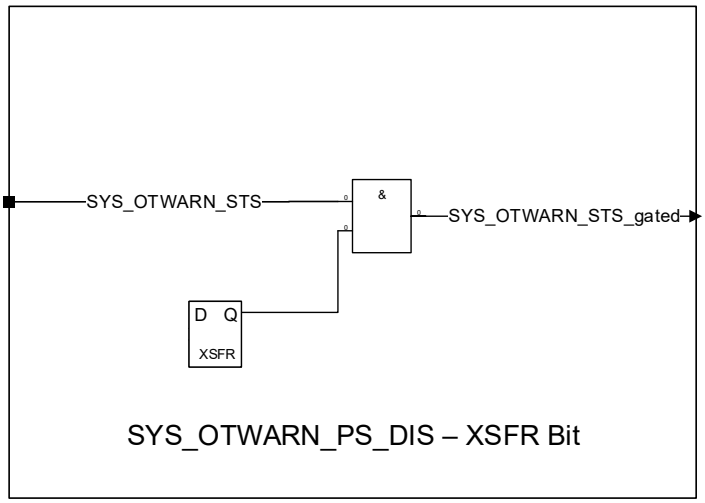


Figure 43 Implementation of Power Module Shutdown in case of System Overtemperature Warning

As it can be seen, the bit is gating the status flag VS_OTWARN_STS. If this bit is set, 1ms after the indication the system will be set into Sleep Mode.

8.5.3 Power Control Unit Register

The PCU is fully controllable by the below listed SFR Registers.

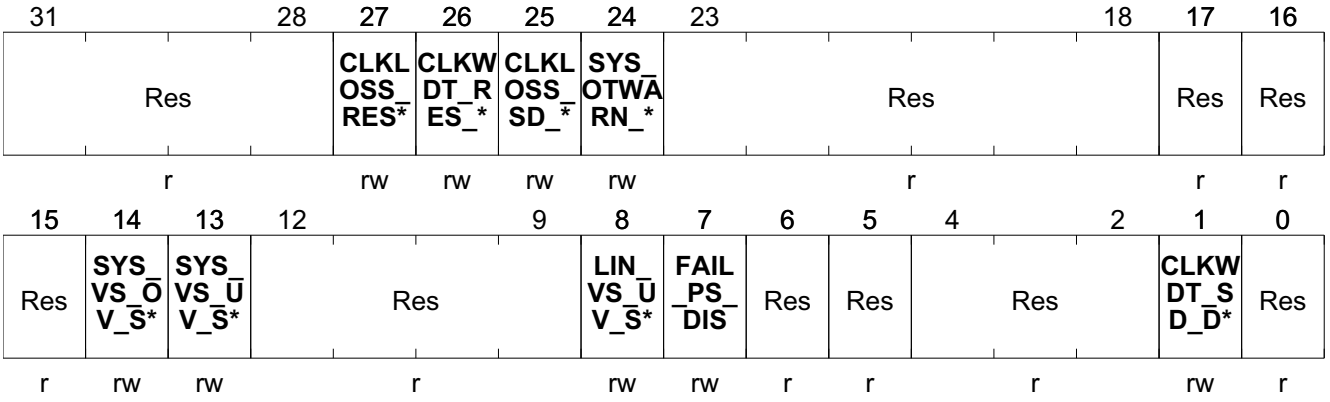
Table 141 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Power Control Unit Register			
SCUPM_PCU_CTRL_STS	Power Control Unit Control Status Register	30_H	$0EE37EF3_H$

The registers are addressed wordwise.

Power Control Unit Control Status Register

SCUPM_PCU_CTRL_STS	Offset	Reset Value
Power Control Unit Control Status Register	30_H	see Table 142



Field	Bits	Type	Description
Res	31:28	r	**Reserved** Always read as 0
CLKLOSS_RES_SD_DIS	27	rw	**Loss of Clock Reset Disable** 0_B **Enable**, Loss of Clock Reset Enable 1_B **Disable** , Loss of Clock Reset Disable
CLKWDT_RES_SD_DIS	26	rw	**Clock Watchdog Reset Disable** 0_B **Enable**, Clock Watchdog Reset Enable 1_B **Disable** , Clock Watchdog Reset Disable
CLKLOSS_SD_DIS	25	rw	**System Loss of Clock Shutdown Disable (AMCLK3)** 0_B **Enable**, Automatic Shutdown Signal for Power Switches in case of loss of clock 1_B **Disable** , Automatic Shutdown Signal for Power Switches in case of loss of clock

Field	Bits	Type	Description
SYS_OTWARN_PS_DIS	24	rw	**System Overtemperature Warning Power Switches Shutdown Disable** 0_B **Enable**, Automatic Shutdown Signal for Power Switches in case of system overtemperature warning enable 1_B **Disable** , Automatic Shutdown Signal for Power Switches in case of system overtemperature warning enable
Res	23:18	r	**Reserved** Always read as 0
Res	17	r	**Reserved** Always read as 0
Res	16:15	r	**Reserved** Always read as 0
SYS_VS_OV_SLM_DIS	14	rw	**VS Overvoltage Shutdown for Peripherals Disable** 0_B **Enable**, Automatic Shutdown for Power modules in case of VS Overvoltage enabled 1_B **Disable** , Automatic Shutdown for Power modules in case of VS Overvoltage disabled
SYS_VS_UV_SLM_DIS	13	rw	**VS Undervoltage Shutdown for Peripherals Disable** 0_B **Enable**, Automatic Shutdown for Power modules in case of VS Undervoltage enabled 1_B **Disable** , Automatic Shutdown for Power modules in case of VS Undervoltage disabled
Res	12:9	r	**Reserved** Always read as 0
LIN_VS_UV_SD_DIS	8	rw	**LIN Module VS Undervoltage Transmitter Shutdown** 0_B **Enable**, Automatic Shutdown for Power modules in case of VS Undervoltage enabled 1_B **Disable** , Automatic Shutdown for Power modules in case of VS Undervoltage disabled
FAIL_PS_DIS	7	rw	**Disable LIN Tx and HS and because of Overtemperature Warning or VS OV/UV** 0_B **Switch off Enabled**, LIN Tx andHS will be turned off when Overtemperature Warning occurs 1_B **Switch off Disabled**, LIN Tx andHS will be kept on when Overtemperature Warning occurs
Res	6	r	**Reserved** Always read as 0
Res	5	r	**Reserved** Always read as 0
Res	4:2	r	**Reserved** Always read as 0

Field	Bits	Type	Description
CLKWDT_SD_DIS	1	rw	**Power Modules Clock Watchdog Shutdown Disable** 0_B **Shutdown Enable**, Power Devices will be switched off when Clock Watchdog. 1_B **Shutdown Disable**, Power Devices will not be shutdown when Clock Watchdog occurs
Res	0	r	**Reserved** Always read as 0

Table 142 RESET of SCUPM_PCU_CTRL_STS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0FE97EF3_H$	RESET_TYPE_4		
TRIM_1	$0FE97EF3_H$	RESET		

9 Arm® Cortex®-M0 Core

9.1 Features

The key features of the Arm® Cortex®-M0 implemented are listed below.

Processor Core. A low gate count core, with low latency interrupt processing:

- Thumb® + Thumb-2® Instruction Set
- Banked stack pointer (SP) only
- Handler and thread modes
- Thumb and debug states
- Interruptible-continued instructions LDM/STM, Push/Pop for low interrupt latency
- Automatic processor state saving and restoration for low latency Interrupt Service Routine (ISR) entry and exit
- Arm® architecture v6-M Style
- Arm®v6 unaligned accesses
- Systick (typ. 1ms)

Nested Vectored Interrupt Controller (NVIC) closely integrated with the processor core to achieve low latency interrupt processing:

- External interrupts, configurable from 1 to 24
- 7 interrupt priority registers for levels from 0 up to 192 in steps of 64
- Dynamic reprioritization of interrupts
- Priority grouping. This enables selection of pre-empting interrupt levels and non pre-empting interrupt levels
- Support for tail-chaining and late arrival of interrupts. This enables back-to-back interrupt processing without the overhead of state saving and restoration between interrupts.
- Processor state automatically saved on interrupt entry, and restored on interrupt exit, with no instruction overhead

Bus interfaces

- Advanced High-performance Bus-Lite (AHB-Lite) interfaces

9.2 Introduction

The Arm® Cortex®-M0 processor is a leading 32-bit processor and provides a high-performance and cost-optimized platform for a broad range of applications including microcontrollers, automotive body systems and industrial control systems. Like the other Arm® Cortex® family processors, the Arm® Cortex®-M0 processor implements the Thumb®-2 instruction set architecture. With the optimized feature set the Arm® Cortex®-M0 delivers 32-bit performance in an application space that is usually associated with 8- and 16-bit microcontrollers.

9.2.1 Block Diagram

Figure 44 shows the functional blocks of the Arm® Cortex®-M0.

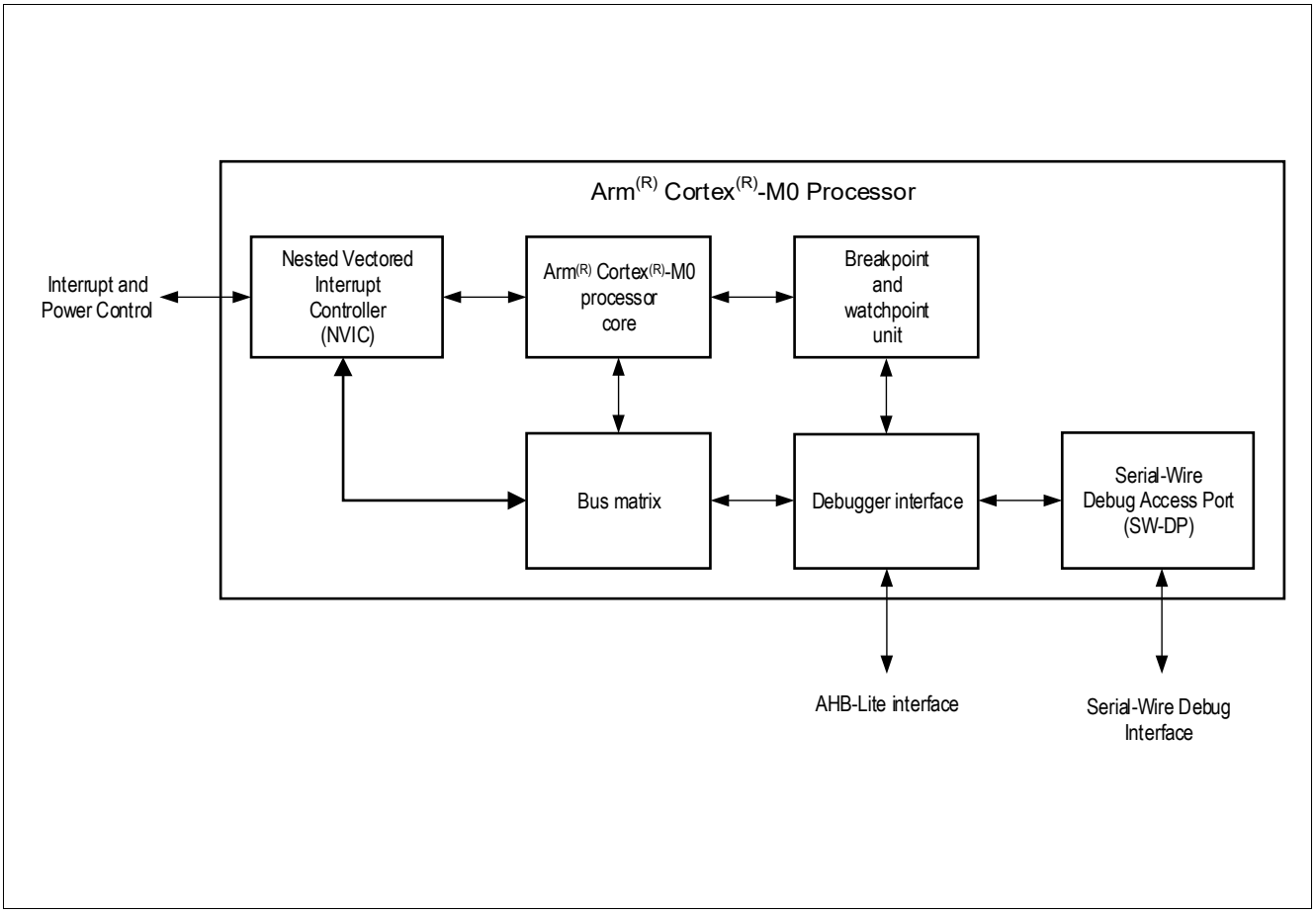


Figure 44 Arm® Cortex®-M0 Block Diagram

9.3 Funtional Description

9.3.1 Registers

The processor has the following 32-bit registers:

- 13 general-purpose registers, R0-R12
- Stack pointer (SP), R13 alias of banked registers, SP_process and SP_main
- Link register (LR), R14
- Program counter (PC), R15
- Special-purpose registers

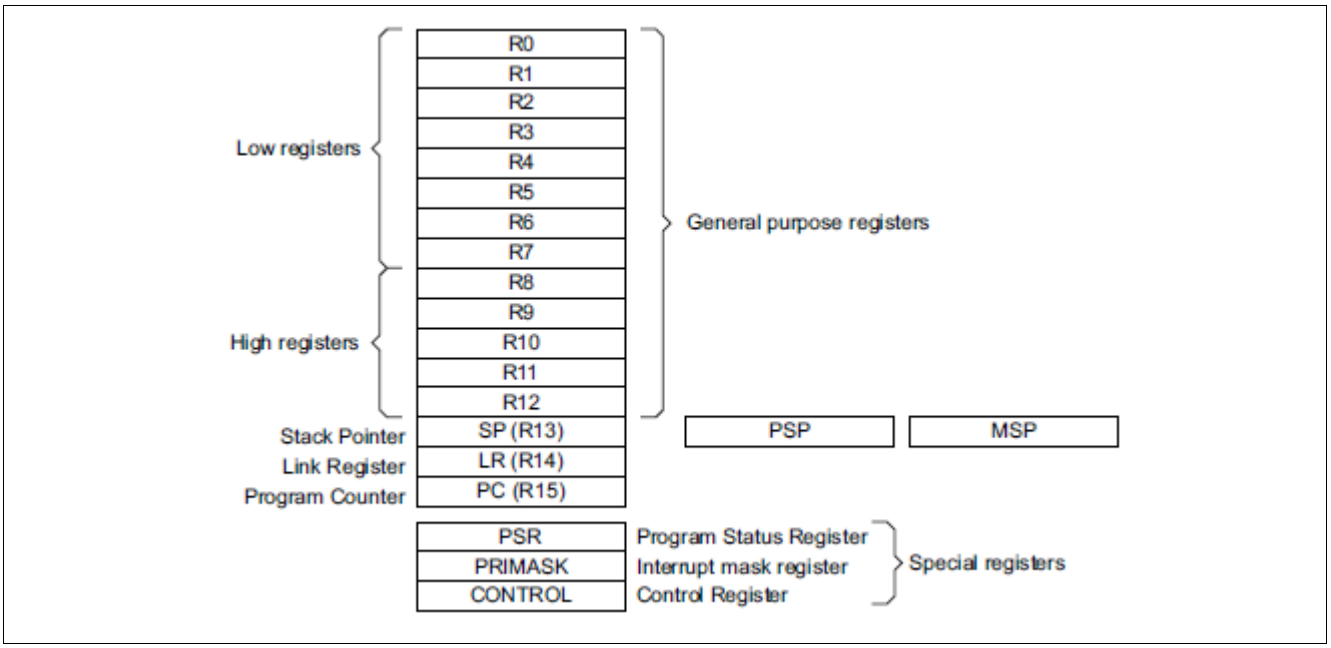


Figure 45 Processor Register Set

9.3.1.1 General-Purpose Registers

The general-purpose registers R0-R12 are 32-bit registers for data operations.

Registers R13, R14, and R15 have the following special functions:

Stack Pointer

Register R13 is used as Stack Pointer (SP).

Link Register

Register R14 is the subroutine Link Register (LR).

Program counter

Register R15 is the Program Counter (PC).

9.3.1.2 Special-Purpose Registers

Program Status Register

Register PSR is the Program Status Register.

Interrupt MaskRegister

Register PRIMASK is the Interrupt Mask Register.

Control Register

Register CONTROL is the Control Register.

9.4 Summary of Processor Registers

The processor has the following 32-bit registers that control functionality:

Table 143 Register Address SpaceAddress Space for Processor Registers

Module	Base Address	End Address	Note
CPU	$E000E000_H$	$E000EFFF_H$	Arm® Cortex®-M0 Core SCS (System Control Space), Systick, NVIC Processor Registers

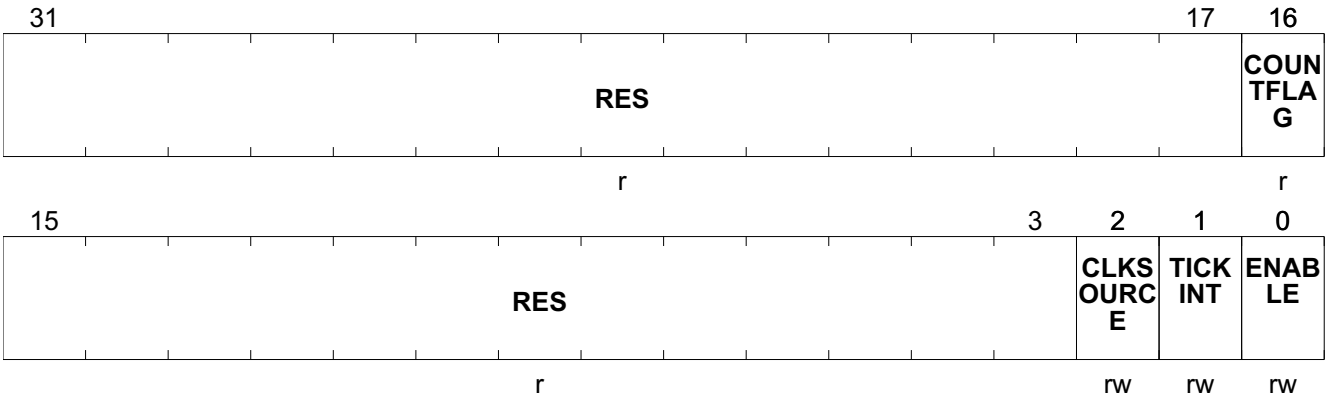
Table 144 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Summary of Processor Registers,			
CPU_SYSTICK_CSR	**SysTick Control and Status Register**	010_H	**00000000_H**
CPU_SYSTICK_RVR	**SysTick Reload Value Register**	014_H	**$00XXXXXX_H$**
CPU_SYSTICK_CVR	**SysTick Current Value Register**	018_H	**$00XXXXXX_H$**
CPU_SYSTICK_CALIB	**SysTick Calibration Value Register**	$01C_H$	**$X0XXXXXX_H$**
CPU_NVIC_ISER	**Interrupt Set-Enable**	100_H	**00000000_H**
CPU_NVIC_ICER	**Interrupt Clear-Enable**	**180_H**	**00000000_H**
CPU_NVIC_ISPR	**Interrupt Set-Pending**	**200_H**	**00000000_H**
CPU_NVIC_ICPR	**Interrupt Clear-Pending**	**280_H**	**00000000_H**
CPU_NVIC_IPR0	**Interrupt Priority**	**400_H**	**00000000_H**
CPU_NVIC_IPR1	**Interrupt Priority**	**404_H**	**00000000_H**
CPU_NVIC_IPR2	**Interrupt Priority**	**408_H**	**00000000_H**
CPU_NVIC_IPR3	**Interrupt Priority**	**$40C_H$**	**00000000_H**
CPU_NVIC_IPR4	**Interrupt Priority**	**410_H**	**00000000_H**
CPU_NVIC_IPR5	**Interrupt Priority**	**414_H**	**00000000_H**
CPU_CPUID	**CPU ID Base Register**	**$D00_H$**	**$410CC200_H$**
CPU_ICSR	**Interrupt Control and State Register**	**$D04_H$**	**00000000_H**
CPU_AIRCR	**Application Interrupt/Reset Control Register**	**$D0C_H$**	**$FA050000_H$**
CPU_SCR	**System Control Register**	**$D10_H$**	**00000000_H**
CPU_CCR	**Configuration Control Register**	**$D14_H$**	**00000208_H**
CPU_SHPR2	**System Handler Priority Register 2**	**$D1C_H$**	**00000000_H**
CPU_SHPR3	**System Handler Priority Register 3**	**$D20_H$**	**00000000_H**

The registers are addressed wordwise.

SysTick Control and Status Register

CPU_SYSTICK_CSR	Offset	Reset Value
SysTick Control and Status Register	010_H	see Table 145



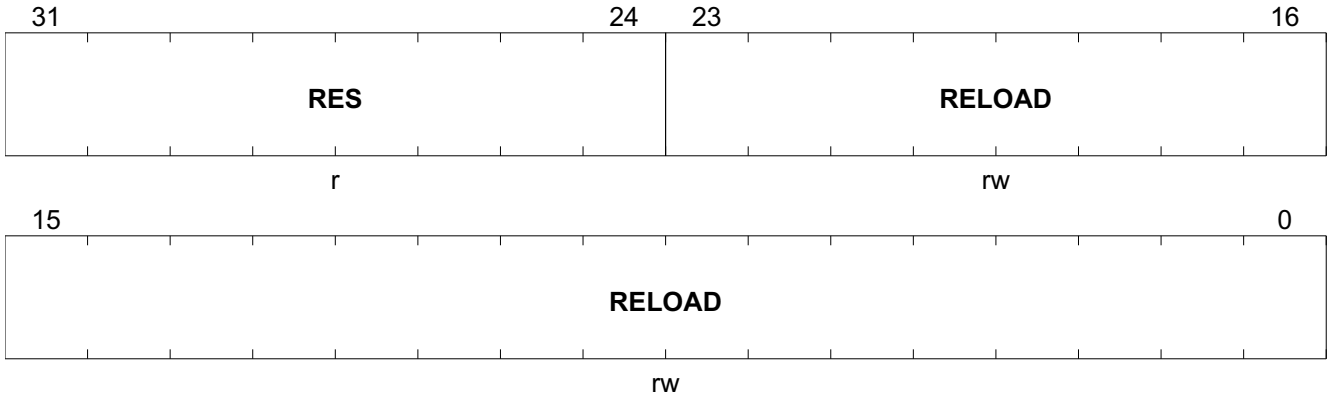
Field	Bits	Type	Description
RES	31:17	r	**Reserved**
COUNTFLAG	16	r	**Count Flag** Returns 1 if timer counted to 0 since the last read of this register.
RES	15:3	r	**Reserved**
CLKSOURCE	2	rw	**CLK Source** Selects the SysTick timer clock source. 0_B **External**, external reference clock 1_B **Processor**, processor clock
TICKINT	1	rw	**TICKINT** Enables SysTick exception request 0_B **No Exception**, counting down to 0 does not assert the SysTick exception request. 1_B **Exception**, counting down to 0 asserts the SysTick exception request.
ENABLE	0	rw	**Enable** 0_B **Disable**, counter disabled. 1_B **Enable**, counter enabled.

Table 145 RESET of CPU_SYSTICK_CSR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

SysTick Reload Value Register

CPU_SYSTICK_RVR	Offset	Reset Value
SysTick Reload Value Register	014_H	see Table 146



Field	Bits	Type	Description
RES	31:24	r	**Reserved**
RELOAD	23:0	rw	**Reload** Value to load into the SysTick Current Value Register when the counter is enabled and when it reaches 0, see **Calculating the RELOAD Value**.

Table 146 RESET of CPU_SYSTICK_RVR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$00XXXXXX_H$	RESET_TYPE_3		

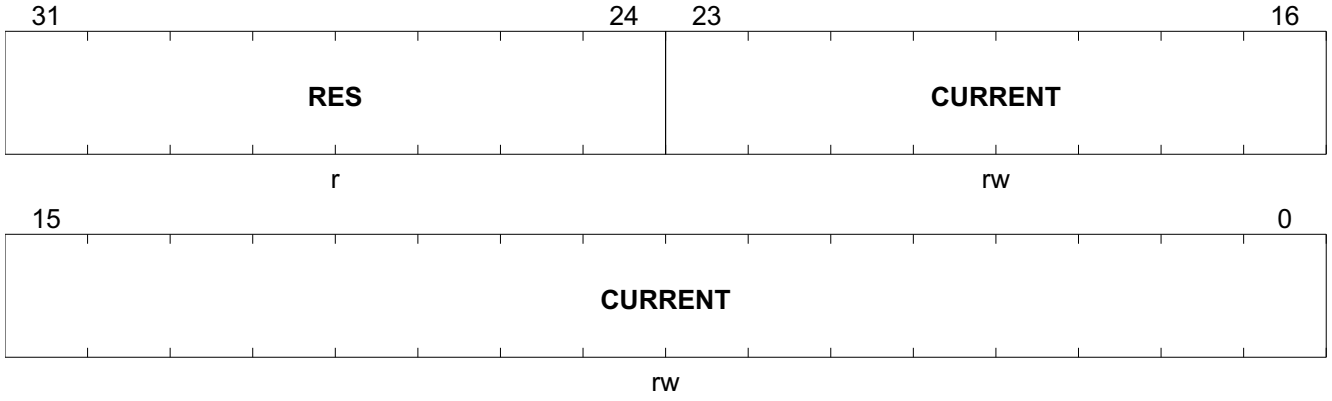
Calculating the RELOAD Value

The RELOAD value can be any value in the range 00000001_H to $00FFFFFF_H$. You can program a value of 0, but this has no effect because the SysTick exception request and COUNTFLAG are activated when counting from 1 to 0.

To generate a multi-shot timer with a period of N processor clock cycles, use a RELOAD value of N-1. For example, if the SysTick interrupt is required every 100 clock pulses, set RELOAD to 99.

SysTick Current Value Registers

CPU_SYSTICK_CVR	Offset	Reset Value
SysTick Current Value Register	018_H	see Table 147



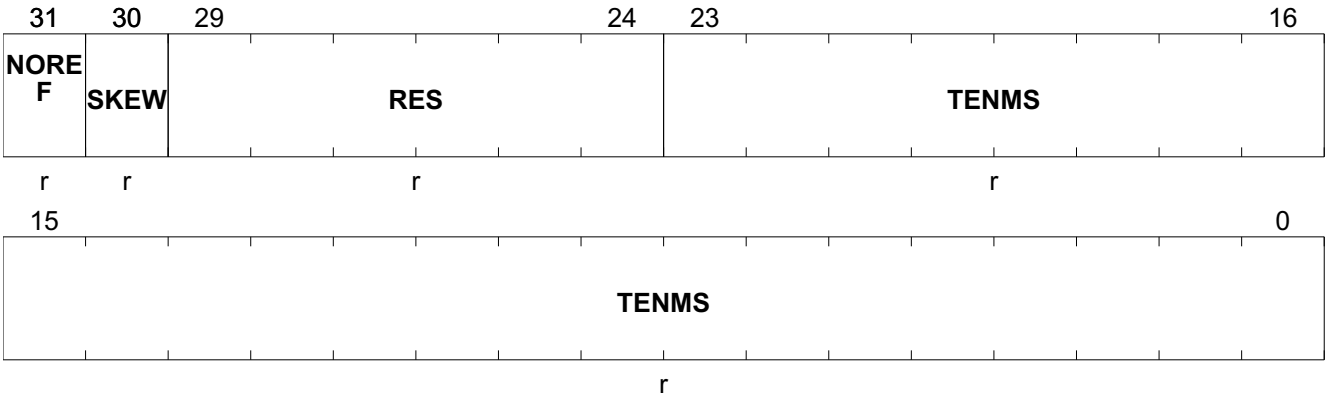
Field	Bits	Type	Description
RES	31:24	r	**Reserved**
CURRENT	23:0	rw	**Current** Reads return the current value of the SysTick counter. A write of any value clears the field to 0, and also clears the SYST_CSR.COUNTFLAG bit to 0.

Table 147 RESET of CPU_SYSTICK_CVR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$00XXXXXX_H$	RESET_TYPE_3		

SysTick Calibration Value Registers

CPU_SYSTICK_CALIB	Offset	Reset Value
SysTick Calibration Value Register	$01C_H$	see Table 148



Field	Bits	Type	Description
NOREF	31	r	**No Reference Clock** Indicates that no separate reference clock is provided. Reads as 0_B.
SKEW	30	r	**Skew** Calibration value for the 10ms inexact timing is not known because TENMS is not known. This can affect the suitability of SysTick as a software real time clock. Reads as 0_B.
RES	29:24	r	**Reserved**
TENMS	23:0	r	**Tenms** Indicates calibration value is not known. If calibration information is not known, calculate the calibration value required from the frequency of the processor clock or external clock. Reads as zero.

Table 148 RESET of CPU_SYSTICK_CALIB

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$X0XXXXXX_H$	RESET_TYPE_3		Exact Reset Values: XX00 0000 XXXX XXXX XXXX XXXX XXXX XXXX(B)

Interrupt Set-Enable Register

CPU_NVIC_ISER — **Offset** — **Reset Value**

Interrupt Set-Enable — **100_H** — **see Table 149**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RES								Int_PORT2	Int_MON	Int_DU	Int_OPA	Int_HS	Int_BDRV	Int_CP	RES
r								rw	rw	rw	rw	rw	rw	rw	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Int_MATHDIV	Int_WAKEUP	Int_EXINT1	Int_EXINT0	Int_UART2	Int_UART1	Int_SSC2	Int_SSC1	Int_CCU6SR3	Int_CCU6SR2	Int_CCU6SR1	Int_CCU6SR0	Int_ADC1	Int_ADC2	Int_GPT2	Int_GPT1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Field	Bits	Type	Description
RES	31:24	r	**Reserved**
Int_PORT2	23	rw	**Interrupt Set for PORT2** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_MON	22	rw	**Interrupt Set for MON** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_DU	21	rw	**Interrupt Set for Differential Unit** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_OPA	20	rw	**Interrupt Set for Current Sense Amplifier** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_HS	19	rw	**Interrupt Set for High-Side Switch** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_BDRV	18	rw	**Interrupt Set for Bridge Driver** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt

Field	Bits	Type	Description
Int_CP	17	rw	**Interrupt Set for Charge Pump** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
RES	16	r	**Reserved**
Int_MATHDIV	15	rw	**Interrupt Set for Math Divider** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_WAKEUP	14	rw	**Interrupt Set for WAKEUP** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_EXINT1	13	rw	**Interrupt Set for External Int 1** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_EXINT0	12	rw	**Interrupt Set for External Int 0** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_UART2	11	rw	**Interrupt Set for UART2** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_UART1	10	rw	**Interrupt Set for UART1** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_SSC2	9	rw	**Interrupt Set for SSC2** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_SSC1	8	rw	**Interrupt Set for SSC1** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_CCU6SR3	7	rw	**Interrupt Set for CCU6 SR3** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_CCU6SR2	6	rw	**Interrupt Set for CCU6 SR2** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_CCU6SR1	5	rw	**Interrupt Set for CCU6 SR1** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_CCU6SR0	4	rw	**Interrupt Set for CCU6 SR0** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_ADC1	3	rw	**Interrupt Set for ADC1** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt

Field	Bits	Type	Description
Int_ADC2	2	rw	**Interrupt Set for MU, ADC2** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_GPT2	1	rw	**Interrupt Set for GPT2** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt
Int_GPT1	0	rw	**Interrupt Set for GPT1** 0_B **DISABLED**, no effect on write 1_B **ENABLE**, enables the associated interrupt

Table 149 RESET of CPU_NVIC_ISER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Interrupt Clear-Enable Register

CPU_NVIC_ICER	Offset	Reset Value
Interrupt Clear-Enable	**180_H**	**see Table 150**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RES								Int_PORT2	Int_MON	Int_DU	Int_OPA	Int_HS	Int_BDRV	Int_CP	RES
r								rw	rw	rw	rw	rw	rw	rw	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Int_MATHDIV	Int_WAKEUP	Int_EXINT1	Int_EXINT0	Int_UART2	Int_UART1	Int_SSC2	Int_SSC1	Int_CCU6SR3	Int_CCU6SR2	Int_CCU6SR1	Int_CCU6SR0	Int_ADC1	Int_ADC2	Int_GPT2	Int_GPT1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Field	Bits	Type	Description
RES	31:24	r	**Reserved**
Int_PORT2	23	rw	**Interrupt Clear for PORT2** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_MON	22	rw	**Interrupt Clear for MON** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled

Field	Bits	Type	Description
Int_DU	21	rw	**Interrupt Clear for Differential Unit** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_OPA	20	rw	**Interrupt Clear for Current Sense Amplifier** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_HS	19	rw	**Interrupt Clear for High-Side Switch** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_BDRV	18	rw	**Interrupt Clear for Bridge Driver** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_CP	17	rw	**Interrupt Clear for Charge Pump** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
RES	16	r	**Reserved**
Int_MATHDIV	15	rw	**Interrupt Clear for Math Divider** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_WAKEUP	14	rw	**Interrupt Clear for WAKEUP** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_EXINT1	13	rw	**Interrupt Clear for External Int 1** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_EXINT0	12	rw	**Interrupt Clear for External Int 0** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled

Field	Bits	Type	Description
Int_UART2	11	rw	**Interrupt Clear for UART2** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_UART1	10	rw	**Interrupt Clear for UART1** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_SSC2	9	rw	**Interrupt Clear for SSC2** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_SSC1	8	rw	**Interrupt Clear for SSC1** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_CCU6SR3	7	rw	**Interrupt Clear for CCU6 SR3** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_CCU6SR2	6	rw	**Interrupt Clear for CCU6 SR2** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_CCU6SR1	5	rw	**Interrupt Clear for CCU6 SR1** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_CCU6SR0	4	rw	**Interrupt Clear for CCU6 SR0** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_ADC1	3	rw	**Interrupt Clear for ADC1** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled

Field	Bits	Type	Description
Int_ADC2	2	rw	**Interrupt Clear for MU, ADC2** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_GPT2	1	rw	**Interrupt Clear for GPT2** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled
Int_GPT1	0	rw	**Interrupt Clear for GPT1** 0_B **DISABLE**, on reads the associated interrupt is disabled, no effect on write 1_B **ENABLE**, on reads the associated interrupt is enabled, on writes the associated interrupt is disabled

Table 150 RESET of CPU_NVIC_ICER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Interrupt Set-Pending Register

CPU_NVIC_ISPR	Offset	Reset Value
Interrupt Set-Pending	200_H	see Table 151

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RES								Int_PORT2	Int_MON	Int_DU	Int_OPA	Int_HS	Int_BDRV	Int_CP	RES
r								rw	rw	rw	rw	rw	rw	rw	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Int_MATHDIV	Int_WAKEUP	Int_EXINT1	Int_EXINT0	Int_UART2	Int_UART1	Int_SSC2	Int_SSC1	Int_CCU6SR3	Int_CCU6SR2	Int_CCU6SR1	Int_CCU6SR0	Int_ADC1	Int_ADC2	Int_GPT2	Int_GPT1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Field	Bits	Type	Description
RES	31:24	r	**Reserved**
Int_PORT2	23	rw	**Interrupt Set Pending for PORT2** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending

Field	Bits	Type	Description
Int_MON	22	rw	**Interrupt Set Pending for MON** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_DU	21	rw	**Interrupt Set Pending for Differential Unit** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_OPA	20	rw	**Interrupt Set Pending for Current Sense Amplifier** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_HS	19	rw	**Interrupt Set Pending for High-Side Switch** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_BDRV	18	rw	**Interrupt Set Pending for Bridge Driver** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_CP	17	rw	**Interrupt Set Pending for Charge Pump** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
RES	16	r	**Reserved**
Int_MATHDIV	15	rw	**Interrupt Set Pending for Math Divider** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_WAKEUP	14	rw	**Interrupt Set Pending for WAKEUP** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_EXINT1	13	rw	**Interrupt Set Pending for External Int 1** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_EXINT0	12	rw	**Interrupt Set Pending for External Int 0** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_UART2	11	rw	**Interrupt Set Pending for UART2** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending

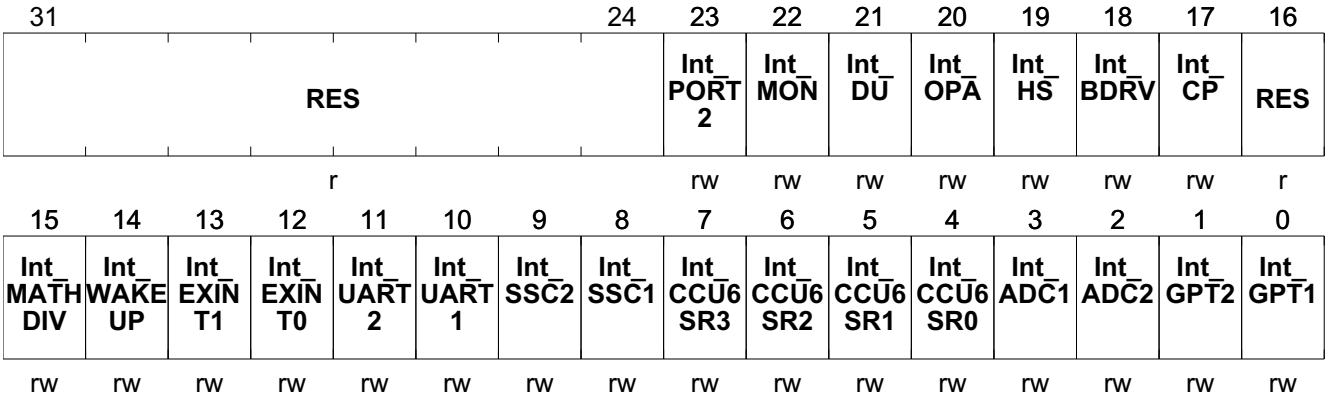
Field	Bits	Type	Description
Int_UART1	10	rw	**Interrupt Set Pending for UART1** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_SSC2	9	rw	**Interrupt Set Pending for SSC2** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_SSC1	8	rw	**Interrupt Set Pending for SSC1** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_CCU6SR3	7	rw	**Interrupt Set Pending for CCU6 SR3** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_CCU6SR2	6	rw	**Interrupt Set Pending for CCU6 SR2** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_CCU6SR1	5	rw	**Interrupt Set Pending for CCU6 SR1** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_CCU6SR0	4	rw	**Interrupt Set Pending for CCU6 SR0** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_ADC1	3	rw	**Interrupt Set Pending for ADC1** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_ADC2	2	rw	**Interrupt Set Pending for MU, ADC2** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_GPT2	1	rw	**Interrupt Set Pending for GPT2** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending
Int_GPT1	0	rw	**Interrupt Set Pending for GPT1** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, the associated interrupt is pending

Table 151 RESET of CPU_NVIC_ISPR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Interrupt Clear-Pending Register

CPU_NVIC_ICPR	Offset	Reset Value
Interrupt Clear-Pending	280_H	see Table 152



Field	Bits	Type	Description
RES	31:24	r	**Reserved**
Int_PORT2	23	rw	**Interrupt Clear Pending for PORT2** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_MON	22	rw	**Interrupt Clear Pending for MON** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_DU	21	rw	**Interrupt Clear Pending for Differential Unit** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_OPA	20	rw	**Interrupt Clear Pending for Current Sense Amplifier** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending

Field	Bits	Type	Description
Int_HS	19	rw	**Interrupt Clear Pending for High-Side Switch** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_BDRV	18	rw	**Interrupt Clear Pending for Bridge Driver** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_CP	17	rw	**Interrupt Clear Pending for Charge Pump** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
RES	16	r	**Reserved**
Int_MATHDIV	15	rw	**Interrupt Clear Pending for Math Divider** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_WAKEUP	14	rw	**Interrupt Clear Pending for WAKEUP** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_EXINT1	13	rw	**Interrupt Clear Pending for External Int 1** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_EXINT0	12	rw	**Interrupt Clear Pending for External Int 0** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_UART2	11	rw	**Interrupt Clear Pending for UART2** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_UART1	10	rw	**Interrupt Clear Pending for UART1** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending

Field	Bits	Type	Description
Int_SSC2	9	rw	**Interrupt Clear Pending for SSC2** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_SSC1	8	rw	**Interrupt Clear Pending for SSC1** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_CCU6SR3	7	rw	**Interrupt Clear Pending for CCU6 SR3** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_CCU6SR2	6	rw	**Interrupt Clear Pending for CCU6 SR2** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_CCU6SR1	5	rw	**Interrupt Clear Pending for CCU6 SR1** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_CCU6SR0	4	rw	**Interrupt Clear Pending for CCU6 SR0** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_ADC1	3	rw	**Interrupt Clear Pending for ADC1** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_ADC2	2	rw	**Interrupt Clear Pending for MU, ADC2** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending
Int_GPT2	1	rw	**Interrupt Clear Pending for GPT2** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending

Field	Bits	Type	Description
Int_GPT1	0	rw	**Interrupt Clear Pending for GPT1** 0_B **Not Pending**, on reads the associated interrupt is not pending, no effect on writes 1_B **Pending**, on reads the associated interrupt is pending, on writes the status of the associated interrupt is changed to not pending

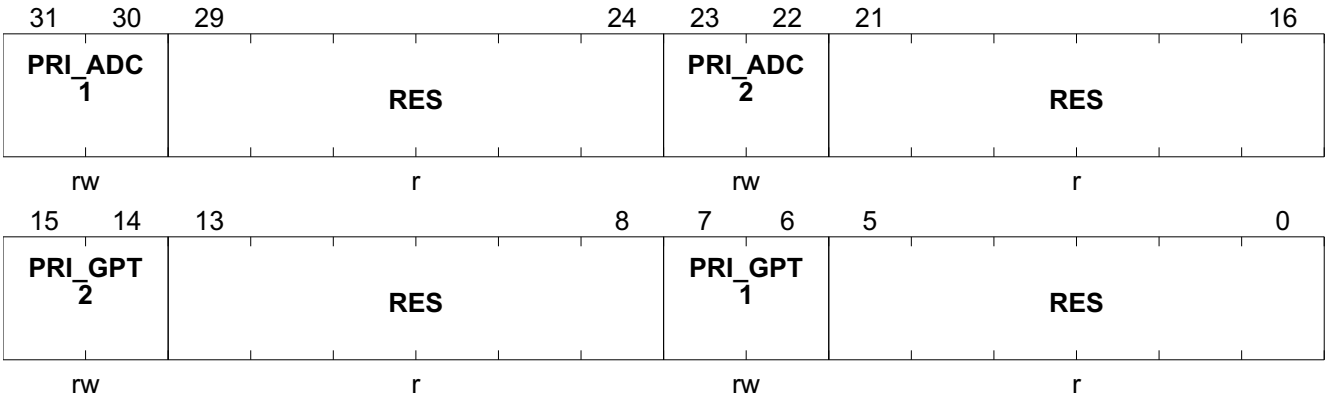
Table 152 RESET of CPU_NVIC_ICPR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Interrupt Priority Register 0

CPU_NVIC_IPR0 **Offset** **Reset Value**

Interrupt Priority **400_H** **see Table 153**



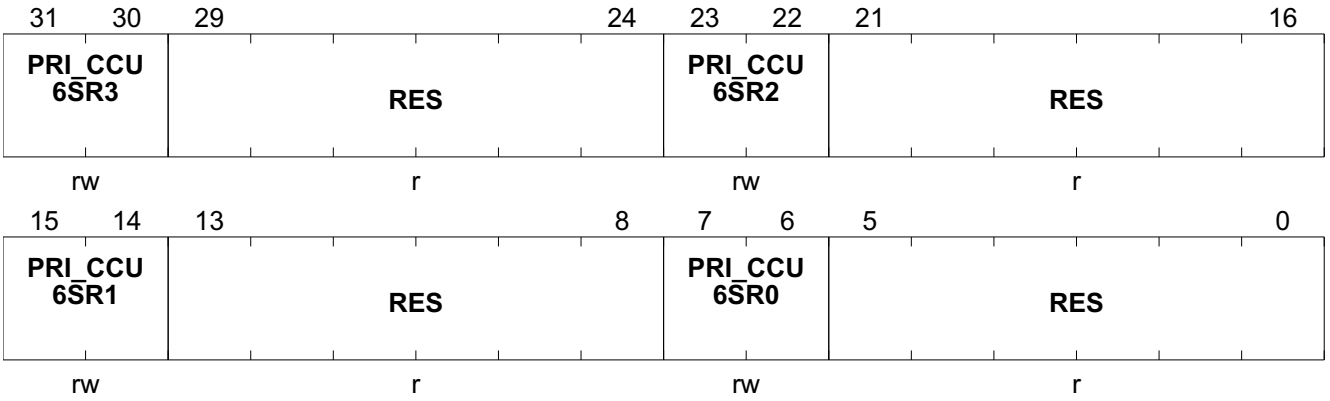
Field	Bits	Type	Description
PRI_ADC1	31:30	rw	**Priority for ADC1**
RES	29:24	r	**Reserved**
PRI_ADC2	23:22	rw	**Priority for MU, ADC2**
RES	21:16	r	**Reserved**
PRI_GPT2	15:14	rw	**Priority for GPT2**
RES	13:8	r	**Reserved**
PRI_GPT1	7:6	rw	**Priority for GPT1**
RES	5:0	r	**Reserved**

Table 153 RESET of CPU_NVIC_IPR0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Interrupt Priority Register 1

CPU_NVIC_IPR1	Offset	Reset Value
Interrupt Priority	404_H	see Table 154



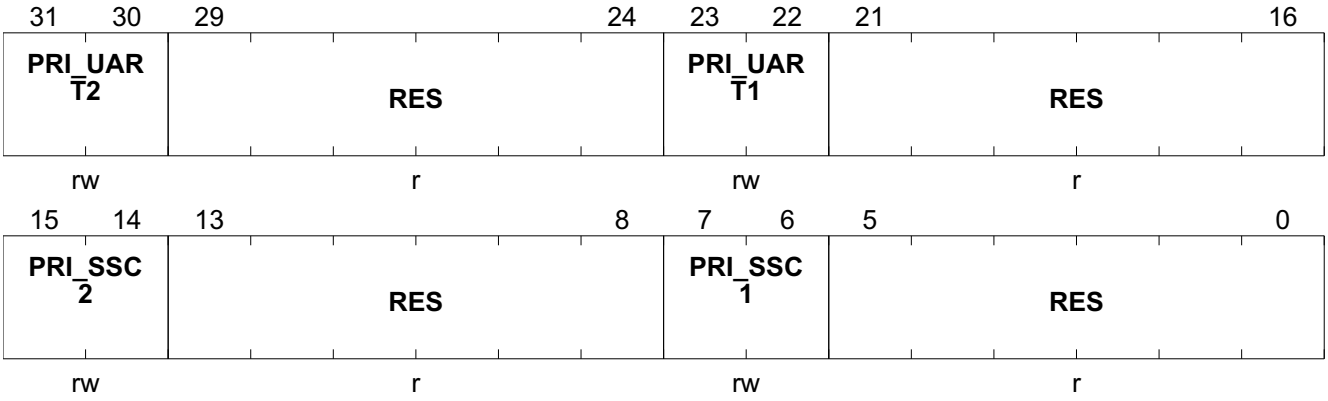
Field	Bits	Type	Description
PRI_CCU6SR3	31:30	rw	**Priority for CCU6 SR3**
RES	29:24	r	**Reserved**
PRI_CCU6SR2	23:22	rw	**Priority for CCU6 SR2**
RES	21:16	r	**Reserved**
PRI_CCU6SR1	15:14	rw	**Priority for CCU6 SR1**
RES	13:8	r	**Reserved**
PRI_CCU6SR0	7:6	rw	**Priority for CCU6 SR0**
RES	5:0	r	**Reserved**

Table 154 RESET of CPU_NVIC_IPR1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Interrupt Priority Register 2

CPU_NVIC_IPR2	Offset	Reset Value
Interrupt Priority	408_H	see Table 155



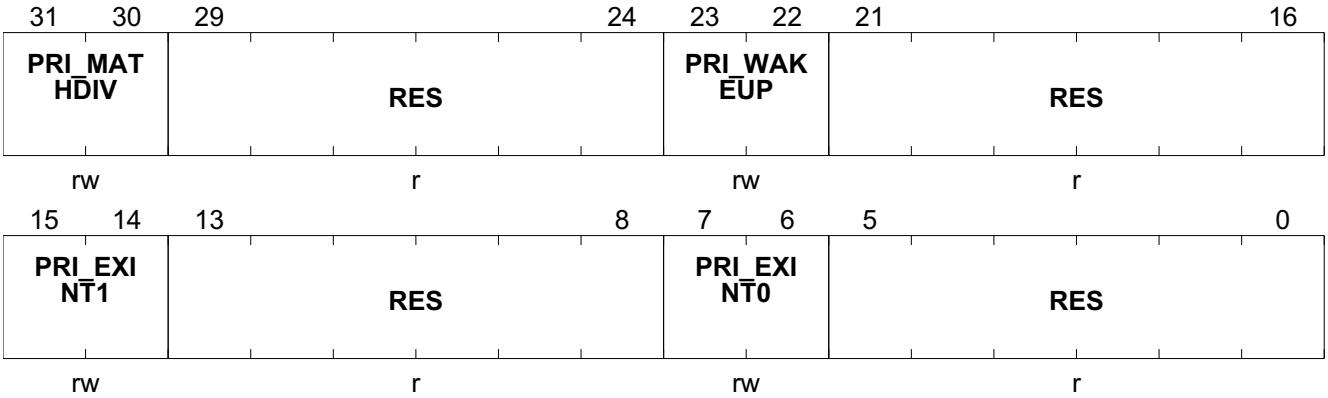
Field	Bits	Type	Description
PRI_UART2	31:30	rw	**Priority for UART2**
RES	29:24	r	**Reserved**
PRI_UART1	23:22	rw	**Priority for UART1**
RES	21:16	r	**Reserved**
PRI_SSC2	15:14	rw	**Priority for SSC2**
RES	13:8	r	**Reserved**
PRI_SSC1	7:6	rw	**Priority for SSC1**
RES	5:0	r	**Reserved**

Table 155 RESET of CPU_NVIC_IPR2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Interrupt Priority Register 3

CPU_NVIC_IPR3	Offset	Reset Value
Interrupt Priority	**$40C_H$**	**see Table 156**



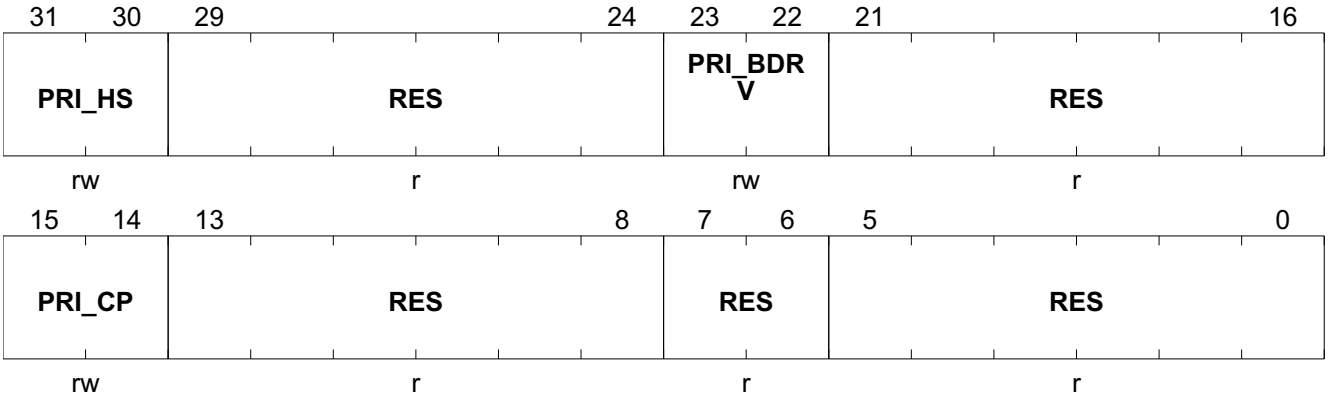
Field	Bits	Type	Description
PRI_MATHDIV	31:30	rw	**Priority for Math Divider**
RES	29:24	r	**Reserved**
PRI_WAKEUP	23:22	rw	**Priority for WAKEUP**
RES	21:16	r	**Reserved**
PRI_EXINT1	15:14	rw	**Priority for External Int 1**
RES	13:8	r	**Reserved**
PRI_EXINT0	7:6	rw	**Priority for External Int 0**
RES	5:0	r	**Reserved**

Table 156 RESET of CPU_NVIC_IPR3

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Interrupt Priority Register 4

CPU_NVIC_IPR4	Offset	Reset Value
Interrupt Priority	**410_H**	**see Table 157**



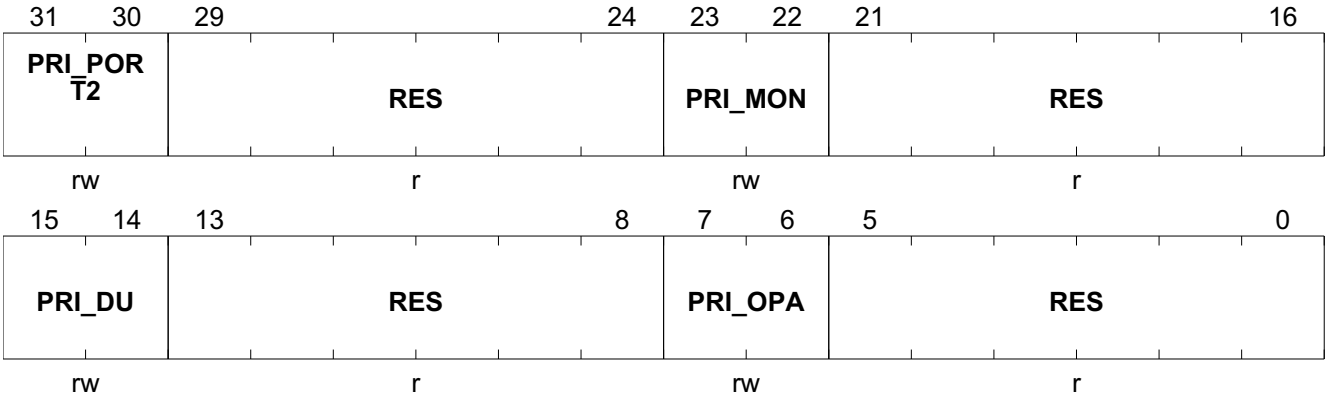
Field	Bits	Type	Description
PRI_HS	31:30	rw	**Priority for High-Side Switch**
RES	29:24	r	**Reserved**
PRI_BDRV	23:22	rw	**Priority for Bridge Driver**
RES	21:16	r	**Reserved**
PRI_CP	15:14	rw	**Priority for Charge Pump**
RES	13:8	r	**Reserved**
RES	7:6	r	**Reserved**
RES	5:0	r	**Reserved**

Table 157 RESET of CPU_NVIC_IPR4

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Interrupt Priority Register 5

CPU_NVIC_IPR5	Offset	Reset Value
Interrupt Priority	414_H	see Table 158



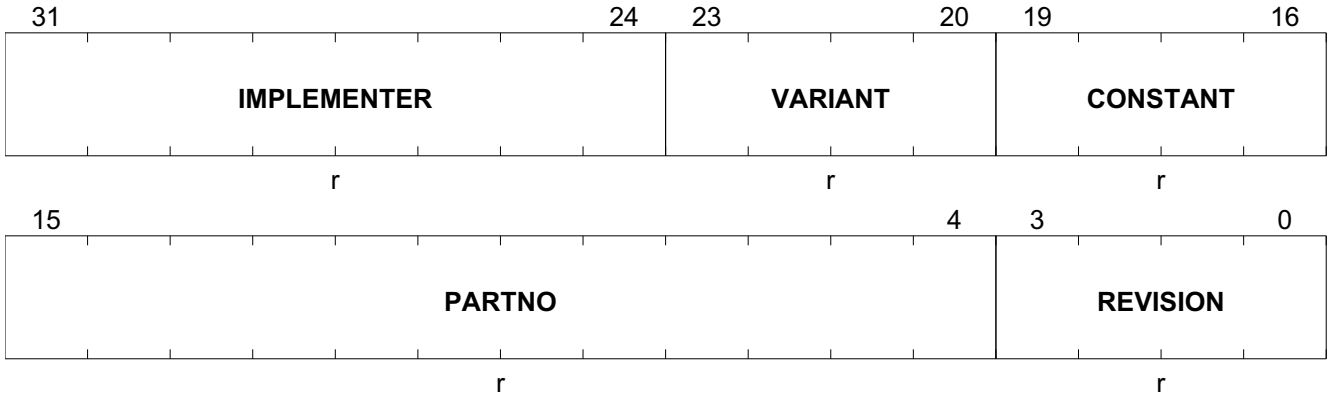
Field	Bits	Type	Description
PRI_PORT2	31:30	rw	**Priority for PORT2**
RES	29:24	r	**Reserved**
PRI_MON	23:22	rw	**Priority for MON**
RES	21:16	r	**Reserved**
PRI_DU	15:14	rw	**Priority for Differential Unit**
RES	13:8	r	**Reserved**
PRI_OPA	7:6	rw	**Priority for Current Sense Amplifier**
RES	5:0	r	**Reserved**

Table 158 RESET of CPU_NVIC_IPR5

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

CPU ID Base Register

CPU_CPUID	Offset	Reset Value
CPU ID Base Register	$D00_H$	see Table 159



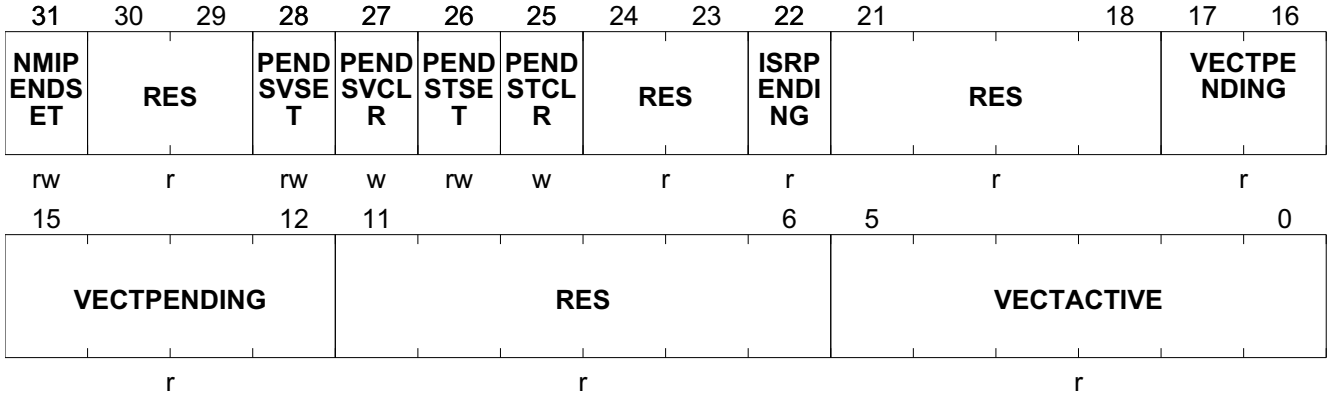
Field	Bits	Type	Description
IMPLEMENTER	31:24	r	**Implementer Code** Assigned by Arm®. Read as 41_H for a processor implemented by Arm®.
VARIANT	23:20	r	**Variant Number** Implementation defined.
CONSTANT	19:16	r	**Constant** Defines the architecture of the processor. Read as 0_H.
PARTNO	15:4	r	**Part Number** Implementation defined.
REVISION	3:0	r	**Revision Number** Implementation defined.

Table 159 RESET of CPU_CPUID

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$410CC200_H$	RESET_TYPE_3		

Interrupt Control and State Register

CPU_ICSR	Offset	Reset Value
Interrupt Control and State Register	**$D04_H$**	**see Table 160**



Field	Bits	Type	Description
NMIPENDSET	31	rw	**NMI Set Pending** On writes, makes the NMI exception state pending. On reads, indicates the state of the exception. *Note: Because NMI is the highest-priority exception, normally the processor enters the NMI exception handler as soon as it detects a write of 1 to this bit. Entering the handler then clears this bit to 0. This means a read of this bit by the NMI exception handler returns 1 only if the NMI signal is reasserted while the processor is executing that handler.* 0_B **Not Pending**, on writes, has no effect. On reads, NMI exception is not pending. 1_B **Pending**, on writes, changes the NMI exception state to pending. On reads, NMI exception is pending.
RES	30:29	r	**Reserved**
PENDSVSET	28	rw	**PENDSV Set Pending** On writes, sets the PendSV exception as pending. On reads, indicates the current state of the exception. *Note: Writing 1 to this bit is the only way to set the PENDSV exception state to pending.* 0_B **Not Pending**, on writes, has no effect. On reads, PendSV exception is not pending. 1_B **Pending**, on writes, changes PendSV exception state to pending. On reads, PendSV is pending.
PENDSVCLR	27	w	**PENDSV Clear Pending** Removes the pending status of the PendSV exception 0_B **No Clear**, no effect 1_B **Clear**, remove pending state from the PENDSV exception
PENDSTSET	26	rw	**SysTick Exception Set Pending** On writes, sets the SysTick exception as pending. On reads, indicates the current state of the exception. 0_B **Not Pending**, on writes, has no effect. On reads, SysTick exception is not pending. 1_B **Pending**, on writes, changes SysTick exception state to pending. On reads, SysTick exception is pending.
PENDSTCLR	25	w	**SysTick Exception Clear Pending** Removes the pending status of the SysTick exception. *Note: This bit is write-only. On a register read is value is unknown.* 0_B **No Clear**, no effect 1_B **Clear**, removes the pending state from the SysTick exception
RES	24:23	r	**Reserved**

Field	Bits	Type	Description
ISRPENDING	22	r	**Interrupt Pending Flag** Excluding NMI and Faults. 0_B **Not Pending**, interrupt not pending 1_B **Pending**, interrupt is pending
RES	21:18	r	**Reserved**
VECTPENDING	17:12	r	**VECTPENDING** Indicates the exception number of the highest priority pending enabled exception. Nonzero is the exception number of the highest priority pending enables exception. 0_B **no pending exceptions**,
RES	11:6	r	**Reserved**
VECTACTIVE	5:0	r	**VECTACTIVE**[1)] Contains the active exception number. Nonzero is the exception number[1)] of the currently active exception. *Note: Subtract 16 from this value to obtain the CMSIS IRQ number that identifies the corresponding bit in the Interrupt Clear-Enable, Set-Enable, Clear-Pending, Set-pending, and Priority Register.* *3. When y write to the ICSR the effect is unpredictable if you:* *- write 1 to the PENDSVSET bit and write 1 to the PENDSVCLR bit* *- write 1 to the PENDSTSET bit and write 1 to the PENDSTCLR bit* 0_B **Thread mode**,

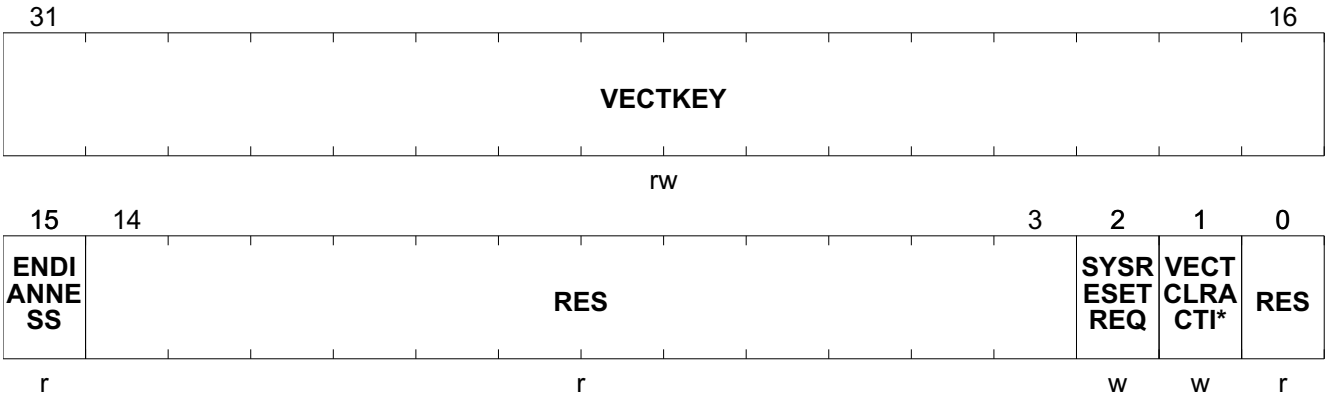
1) This is the same value as IPSR bits 5:0.

Table 160 RESET of CPU_ICSR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Application Interrupt/Reset Control Register

CPU_AIRCR	Offset	Reset Value
Application Interrupt/Reset Control Register	**$D0C_H$**	**see Table 161**



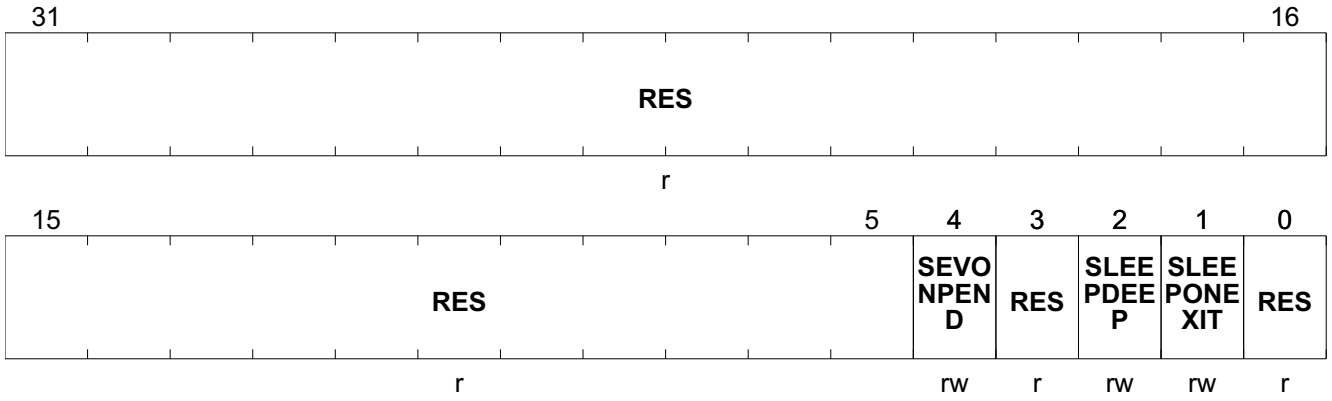
Field	Bits	Type	Description
VECTKEY	31:16	rw	**Vector Key** Register writes must write $05FA_H$ to this field, otherwise the write is ignored. On reads, returns Unknown.
ENDIANNESS	15	r	**Data Endianness** 0_B **Little Endian**, 1_B **Big Endian**,
RES	14:3	r	**Reserved**
SYSRESETREQ	2	w	**System Reset Request** This bit reads as 0_B. 0_B **No Reset**, no effect 1_B **Reset**, request a system level reset
VECTCLRACTIVE	1	w	**VECTCLRACTIVE** Reserved for debug use. This bit reads as 0_B. *Note: When writing to this register you must write 0_B to this bit, otherwise behavior is **unpredictable**.*
RES	0	r	**Reserved**

Table 161 RESET of CPU_AIRCR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$FA050000_H$	RESET_TYPE_3		

System Control Register

CPU_SCR	Offset	Reset Value
System Control Register	$D10_H$	see Table 162



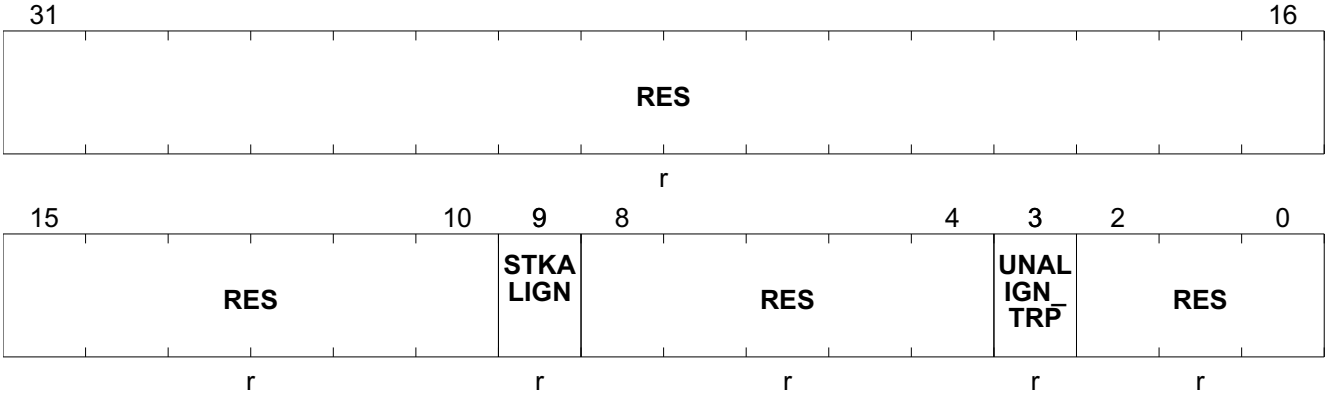
Field	Bits	Type	Description
RES	31:5	r	**Reserved**
SEVONPEND	4	rw	**SEVONPEND** Send event on pending bit. When an event or interrupt enters pending state, the event signal wakes up the processor from WFE. If the processor is not waiting for an event, the event is registered and affects the next WFE. The processor also wakes up on execution of an SEV instruction or an external event. 0_B **Enabled**, only enabled interrupts or events can wake-up the processor, disabled interrupts are excluded 1_B **All**, enabled events and all interrupts, including disabled interrupts, can wake-up the processor
RES	3	r	**Reserved**
SLEEPDEEP	2	rw	**Sleep Deep** Controls whether the processor uses sleep or deep sleep as its low power mode. 0_B **sleep**, 1_B **deep sleep**,
SLEEPONEXIT	1	rw	**Sleep on Exit** Indicates sleep-on-exit when returning from Handler mode to Thread mode. Setting this bit to 1 enables an interrupt driven application to avoid returning to an empty main application. 0_B **Do Not Sleep**, do not sleep when returning to Thread mode 1_B **Enter Sleep**, enter sleep, or deep sleep, on return from an ISR to Thread mode
RES	0	r	**Reserved**

Table 162 RESET of CPU_SCR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Configuration Control Register

CPU_CCR	Offset	Reset Value
Configuration Control Register	$D14_H$	see Table 163



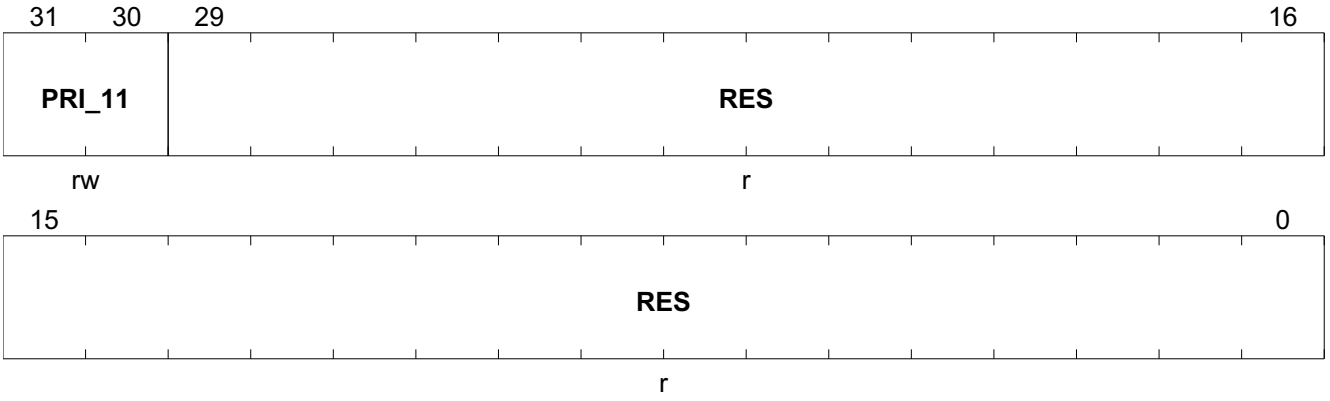
Field	Bits	Type	Description
RES	31:10	r	**Reserved**
STKALIGN	9	r	**STKALIGN** Always reads as one, indicates 8-byte stack alignment on exception entry. On exception entry, the processor uses bit[9] of the stacked PSR to indicate the stack alignment. On return from the exception it uses this stacked bit to restore the correct stack alignment
RES	8:4	r	**Reserved**
UNALIGN_TRP	3	r	**UNALIGN_TRP** Indicates that all unaligned accesses generate a HardFault. Always reads as 1_B.
RES	2:0	r	**Reserved**

Table 163 RESET of CPU_CCR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000208_H	RESET_TYPE_3		

System Handler Priority Register 2

CPU_SHPR2	Offset	Reset Value
System Handler Priority Register 2	$D1C_H$	see Table 164



Field	Bits	Type	Description
PRI_11	31:30	rw	**Priority of System Handler 11, SVCall**
RES	29:0	r	**Reserved**

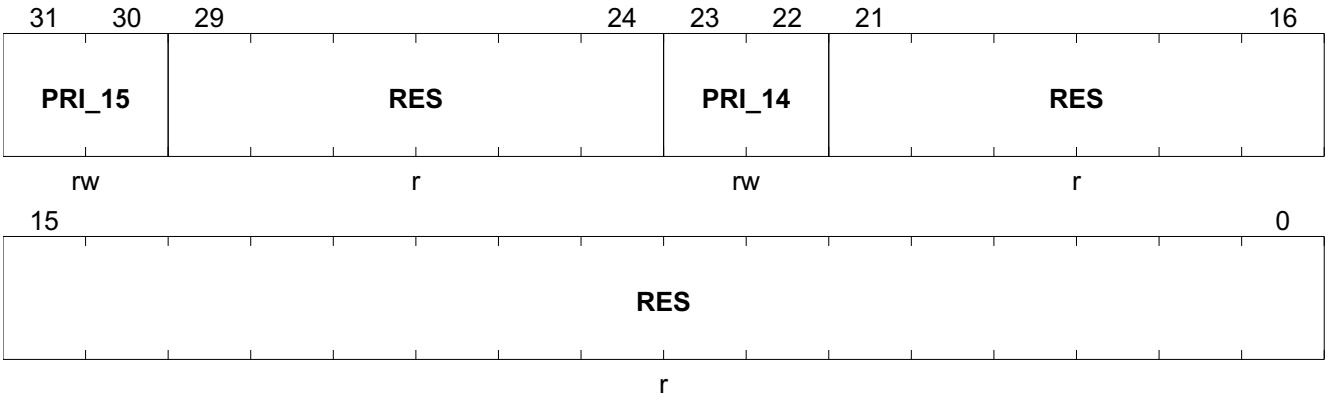
Table 164 RESET of CPU_SHPR2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

System Handler Priority Register 3

CPU_SHPR3 **Offset** **Reset Value**

System Handler Priority Register 3 **$D20_H$** **see Table 165**



Field	Bits	Type	Description
PRI_15	31:30	rw	**Priority of System Handler 15, SysTick**
RES	29:24	r	**Reserved**
PRI_14	23:22	rw	**Priority of System Handler 14, PendSV**
RES	21:0	r	**Reserved**

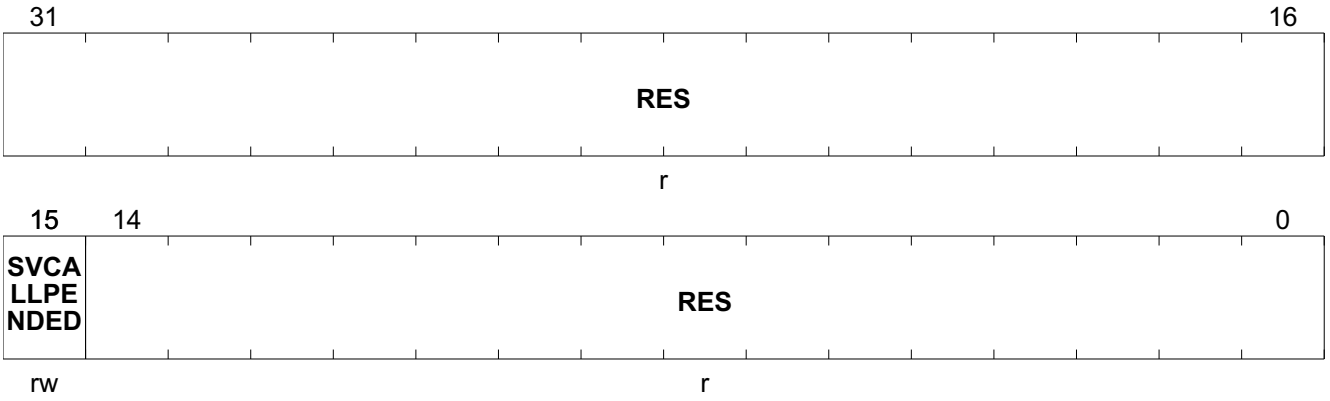
Table 165 RESET of CPU_SHPR3

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

System Handler Control and State Register

CPU_SHCSR **Offset** **Reset Value**

System Handler Control and State Register **$D24_H$** **see Table 166**



Field	Bits	Type	Description
RES	31:16	r	**Reserved**
SVCALLPENDED	15	rw	**SVCALLPENDED** This bit reflects the pending state on a read, and updates the pending state, to the value written, on a write.. Pending state bits are set to 1 when an exception occurs and are cleared to 0 when an exception becomes active. This bit is only writable via DAP. 0_B **Not Pending**, SVCall is not pending 1_B **Pending**, SVCall is pending
RES	14:0	r	**Reserved**

Table 166 RESET of CPU_SHCSR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

9.5 Instruction Set Summary

This chapter provides the Instruction set. **Table 167** shows the instructions and their cycle counts. The cycle counts are based on a system with zero wait states.

Within the assembler syntax, depending on the operation, the <op2> field can be replaced with one of the following options:

- a simple register
- an immediate shifted register
- a register shifted register
- an immediate value

For brevity, not all load and store addressing modes are shown.

Table 167 uses the following abbreviations in the cycles column:

- P for the number of cycles required for a pipeline refill.
- B for the number of cycles required to perform the barrier operation.
- N for the number of registers in the register list to be loaded or stored, including PC or LR.
- W for the number of cycles spent waiting for an appropriate event.

Table 167 Instruction Set Summary

Operation	Description	Mnemonic	Cycles (without wait states)
Move	Register	MOV Rd, Rm	1
Add	Add	ADD Rd, Rn, <op2>	1
	Add with carry	ADCS Rd, Rn, Rm	1
ADR	Address to Register	ADR Rd, <label>	1
Subtract	Subtract	SUB Rd, Rn, <op2>	1
	Subtract with carry	SBCS Rd, Rn, Rm	1
	Reverse	RSBS Rd, Rn, #0	1
Multiply	Multiply, 32-bit result	MULS Rd, Rn, Rm	1
Compare	Compare	CMP Rn, <op2>	1
	Negative	CMN Rn, Rm	1
Logical	AND bitwise	ANDS Rd, Rn, <op2>	1
	Exclusive OR	EORS Rd, Rn, Rm	1
	OR	ORRS Rd, Rn, Rm	1
	Bit clear	BICS Rd, Rn, <op2>	1
	Move NOT bitwise	MVNS Rd, Rm	1
	AND test	TST Rn, Rm	1
Shift	Logical shift left	LSLS Rd, Rn, #<imm>	1
	Logical shift left	LSLS Rd, Rn, Rs	1
	Logical shift right	LSRS Rd, Rn, #<imm>	1
	Logical shift right	LSRS Rd, Rn, Rs	1
	Arithmetic shift right	ASRS Rd, Rn, #<imm>	1
	Arithmetic shift right	ASRS Rd, Rn, Rs	1

Table 167 Instruction Set Summary (cont'd)

Operation	Description	Mnemonic	Cycles (without wait states)
Rotate	Rotate right	ROR Rd, Rn, Rs	1
Load	Word	LDR Rt, [Rn, <op2>]	2[1]
	Halfword	LDRH Rt, [Rn, <op2>]	2[1]
	Byte	LDRB Rt, [Rn, <op2>]	2[1]
	Signed halfword	LDRSH Rt, [Rn, <op2>]	2[1]
	Signed byte	LDRSB Rt, [Rn, <op2>]	2[1]
	Register from PC relative address	LDR Rt, label	2[1]
	Multiple register, increment after	LDM Rn, {<reglist>}	1 + N
Store	Word	STR Rt, [Rn, <op2>]	2[1]
	Halfword	STRH Rt, [Rn, <op2>]	2[1]
	Byte	STRB Rt, [Rn, <op2>]	2[1]
	Multiple register, increment after	STM Rn, {<reglist>}	1 + N
Push	Push registers onto stack	PUSH {<reglist>}	1 + N
Pop	Pop registers from stack	POP {<reglist>}	1 + N
Branch	Conditional	B <cc> <label>	1 or 1 + P[2]
	Unconditional	B <label>	1 + P
	With link	BL <label>	1 + P
	Indirect	BX Rm	1 + P
	Indirect with link	BLX Rm	1 + P
State change	Supervisor call	SVC #<imm>	–
	Disable interrupts	CPSID i	1 or 2
	Enable interrupts	CPSIE i	1 or 2
	Move to general register from special register	MRS Rd, <specreg>	1 or 2
	Move to special regsiter from general register	MSR <specreg>, Rn	1 or 2
	Breakpoint	BKPT #<imm>	–
Extend	Signed halfword to word	SXTH Rd, Rm	1
	Signed byte to word	SXTB Rd, Rm	1
	Unsigned halfword	UXTH Rd, Rm	1
	Unsigned byte	UXTB Rd, Rm	1
Bit field	Clear	BICS Rd, Rn, Rm	1
Reverse	Bytes in word	REV Rd, Rm	1
	Bytes in both halfwords	REV16 Rd, Rm	1
	Signed bottom halfword	REVSH Rd, Rm	1
	Subtract	RSBS Rd, Rn, #0	1

Table 167 Instruction Set Summary (cont'd)

Operation	Description	Mnemonic	Cycles (without wait states)
Hint	Send event	SEV	1
	Wait for event	WFE	1 + W
	Wait for interrupt	WFI	1 + W
	No operation	NOP	1
Barriers	Instruction synchronization	ISB	1 + B
	Data memory	DMB	1 + B
	Data synchronization	DSB	1 + B

1) Neighboring load and store single instructions can pipeline their address and data phases. This enables these instructions to complete in a singleexecution cycle.

2) Conditional branch completes in a single cycle if the branch is not taken.

10 Address Space Organization

The TLE985xQX manipulates operands in the following memory spaces:

- Up to 96 KB (product variant dependent) of Flash memory in code space
- 24 KB Boot ROM memory in code space (used for boot code and IP storage)
- 4 KB RAM memory in Arm® Cortex®-M0 code region (RAM can be fetched, read/written as program memory)
- Special function registers (SFRs) in peripheral linear address space

The on-chip memory modules available in the TLE985xQX are:

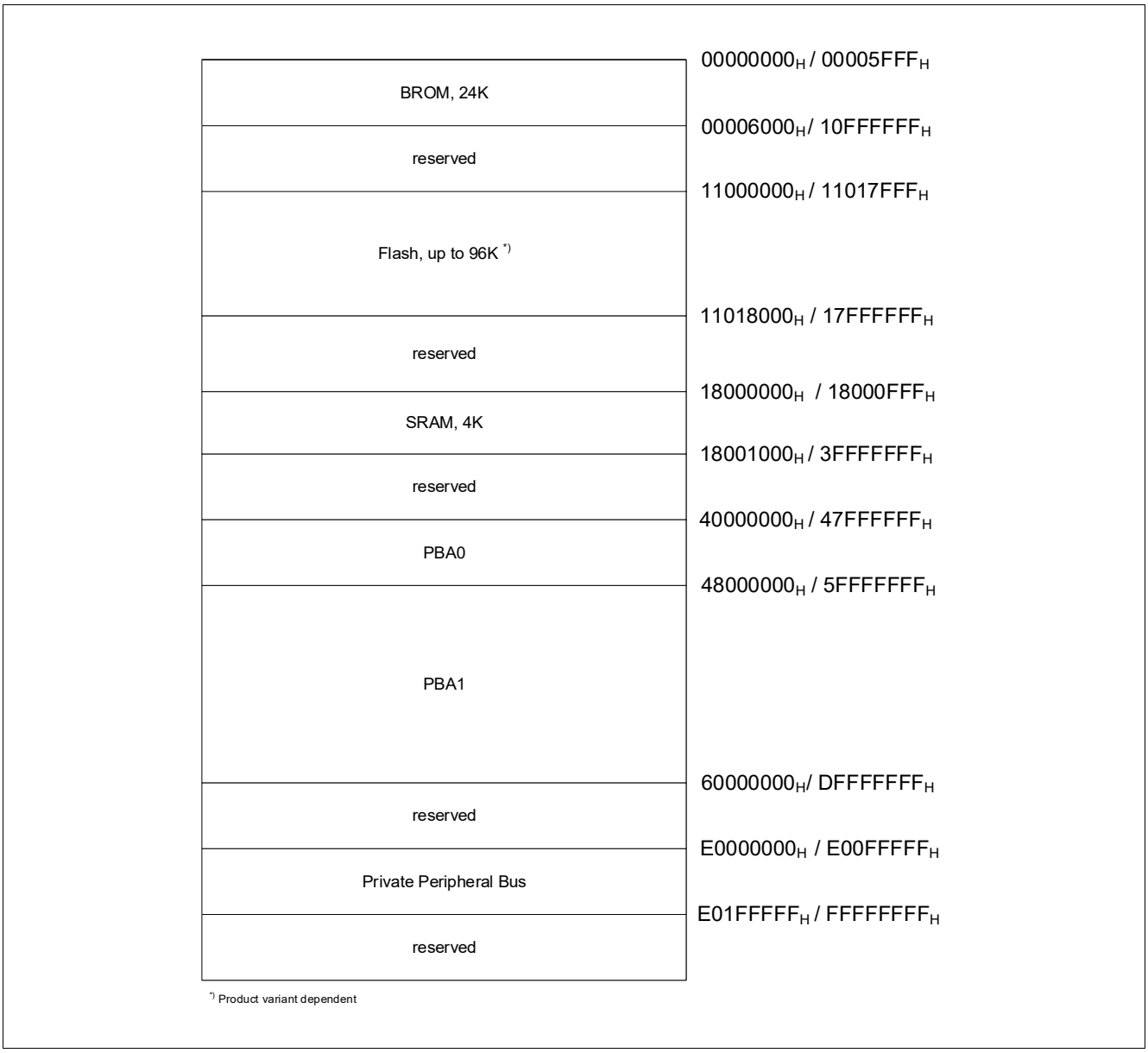


Figure 46 TLE985xQX Memory Map

Each module provides, beside the physical memory implementation, standard AHB-Lite interface and Error Correction Code (ECC) logic if needed.

Table 168 Memory Map

Start (hex)	End (hex)	Size (hex)	Space Name	Usage
0000_0000	0000_5FFF	6000	Code/Data	BROM, 24 KB
0000_6000	10FF_FFFF		Reserved	Reserved
1100_0000	1101_7FFF	18000	Code/Data	NVM, Up to 96 KB[1)]
1101_8000	17FF_FFFF		Reserved	Reserved
1800_0000	1800_0FFF	1000	Code/Data	RAM, 4 KB
1800_1000	3FFF_FFFF			Reserved
4000_0000	47FF_FFFF	08000000	Peripheral 0	Peripheral 0 (PBA0)
4800_0000	5FFF_FFFF	08000000	Peripheral 1	Peripheral 1 (PBA1)
6000_0000	DFFF_FFFF		Reserved	reserved
E000_0000	E00F_FFFF	00100000	PPB, Private Peripheral Bus	CPU
E010_0000	EFFF_FFFF		Vendor specific 1	reserved
F000_0000	FFFF_FFFF		Vendor specific 2	reserved

1) Product variant dependent

Table 169 Peripheral Memory Map

Bus Structure	Modules	Start Address	End Address
Peripherals 0	Reserved	40000000_H	$40003FFF_H$
	ADC1	40004000_H	$40007FFF_H$
	CCU6	$4000C000_H$	$4000FFFF_H$
	GPT12	40010000_H	$40013FFF_H$
	Reserved	40014000_H	$40023FFF_H$
	HS	40024000_H	$40027FFF_H$
	Reserved	40028000_H	$40033FFF_H$
	DRV	40034000_H	$40037FFF_H$
	Reserved	40038000_H	$47FFFFFF_H$
Peripherals 1	Reserved	48000000_H	$48003FFF_H$
	T2	48004000_H	$48004FFF_H$
	T21	48005000_H	$48005FFF_H$
	Reserved	48006000_H	$48012FFF_H$
	MATH_DIV	48013000_H	$48013FFF_H$
	Reserved	48014000_H	$48017FFF_H$
	MF	48018000_H	$4801BFFF_H$
	ADC2	$4801C000_H$	$4801DFFF_H$
	LIN	$4801E000_H$	$4801FFFF_H$
	UART	48020000_H	$48021FFF_H$
	UART2	48021000_H	$48023FFF_H$
	SSC1	48024000_H	$48025FFF_H$
	SSC2	48026000_H	$48027FFF_H$
	PORT	48028000_H	$48029FFF_H$
	Reserved	$4802A000_H$	$50003FFF_H$
	PMU	50004000_H	$50004FFF_H$
	SCU	50005000_H	$50005FFF_H$
	SCUPM	50006000_H	$50006FFF_H$
	Reserved	50007000_H	$5FFFFFFF_H$

11 Memory Control Unit

11.1 Features

- Provides Memory access to ROM, RAM, NVM, Config Sector through AHB-Lite Interface
- MBIST for RAM
- MBIST for ROM
- NVM Configuration with Special Function Registers through AHB-Lite Interface
- Hardware Memory Protection Logic
- Stack overflow detection

11.2 Introduction

11.2.1 Block Diagram

The Memory Control Unit (MCU) is divided in the following sub-modules:

- NVM Memory module (embedded Flash Memory)
- RAM memory module
- BootROM memory module
- Memory protection Unit (MPU) module
- LMB (Local Memory Bus) interface logic.

A block diagram view of the MCU, together with the main interface signals, is shown in the **Figure 47**.

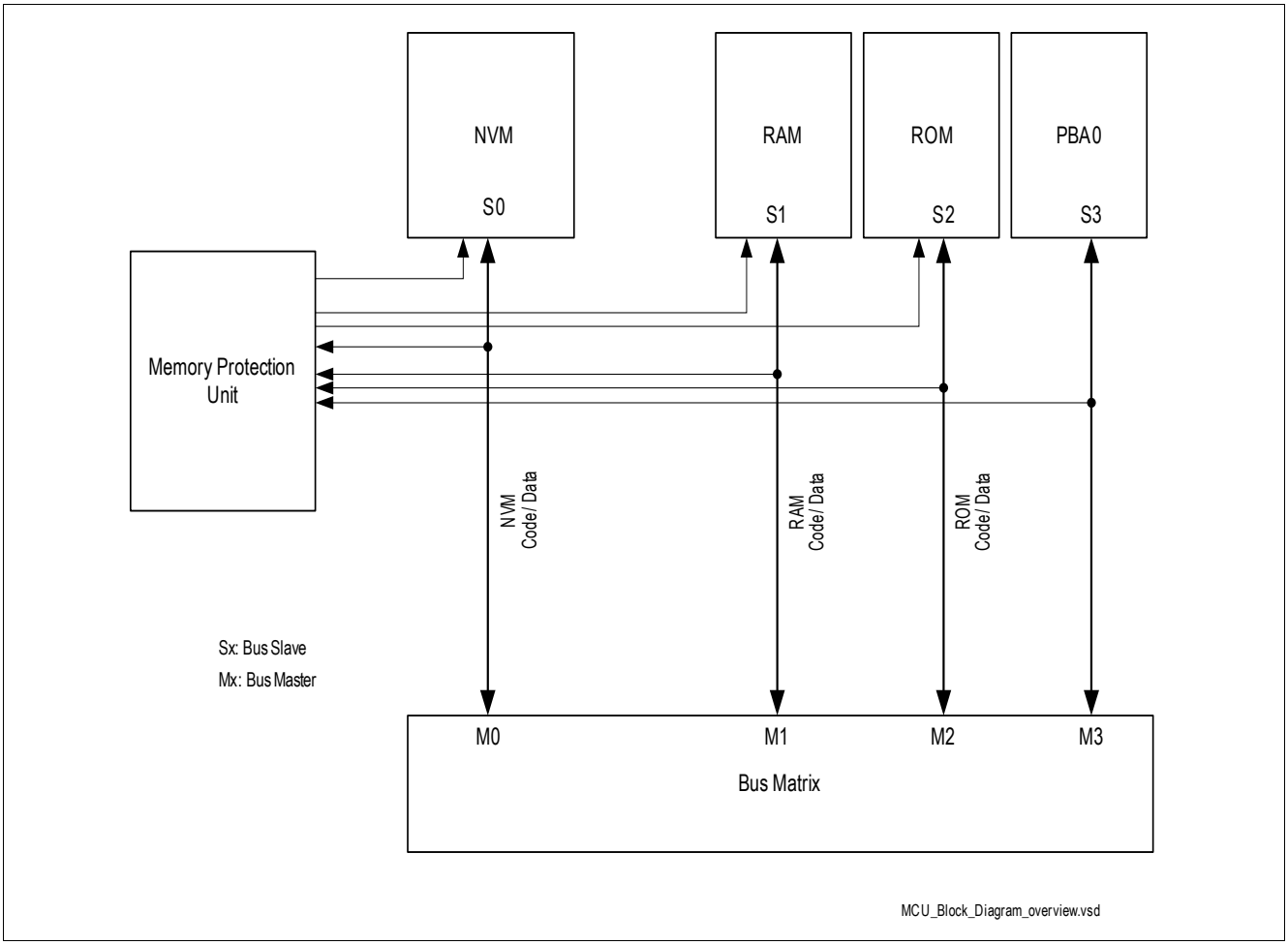


Figure 47 Memory Control Unit Block View

Functional Features for RAM

- 4 KB RAM
- Error correction code (ECC) for detection of single bit and double bit errors and dynamic correction of single bit errors
- Single byte access

As shown in the **Figure 47**, the MCU interface communicates with the external world, mainly the core, via 4 AHB-Lite interfaces, Data/Code access to the NVM, BootROM and RAM plus an access to the NVM internal registers. The AMBA bus matrix block decodes the access requests coming from the masters and forwards them to the target module interface together with the required sideband signals. The AMBA bus matrix block provides all the needed interface functions between the masters and the memory peripheral. It will generate proper HSEL signals, and multiplex the response coming from the modules. In addition, the AMBA bus matrix block takes care of forwarding the transfer according the a fixed priority policy described in the AMBA chapter.

Besides the AHB-Lite and sideband signals, the MCU has access to further Core specific signals, relevant for memory protection .

11.3 NVM Module (Flash Memory)

The Flash memory provides an embedded user-programmable non-volatile memory, allowing fast and reliable storage of user code and data.

Features

- In-System Programming via LIN (Flash mode) and SWD
- Error Correction Code (ECC) for detection of single Bit and double Bit errors and dynamic correction of single Bit errors on Data Block (Double words, 64 bits).
- Interrupt and signaling of double bit error by NMI, address of double bit error readable by FW API user routine
- Possibility of checking single bit error occurrence by ROM routines
- Program width of 128 Byte (page)
- Minimum erase width of 128 Byte (page)
- Integrated hardware support for EEPROM emulation
- 8 Byte read access
- Physical read access time: max. 75 ns
- Code read access acceleration integrated (read buffer)
- Page program time: typ. 3 ms
- Page erase (128 bytes) and sector erase (4K bytes) time: typ. 4 ms
- 4 individual protection passwords for NVM customer BSL region, code region, data linear region, and data mapped region
- Security option to protect read out via debug interface in application run mode
- Write/erase access to 100TP (e.g. option bytes) is possible via the debug interface

Note: The user has to ensure that no flash operations which change the content of the flash get interrupted at any time.

The clock for the NVM is supplied with the system frequency f_{sys}. Integrated firmware routines are provided to ease NVM, and other operations including EEPROM emulation.

The TLE985xQX NVM module provides physical implementation of the memory module as well as needed complementary features and interface towards the core.

The TLE985xQX NVM module consists of the memory cell array and all the control circuits and registers needed to access the array itself. The up to 96 Kbyte (product variant dependent) data module is mapped in the Arm® Cortex®-M0 code address range 11000000H - 11017FFFH (end address is product variant dependent) while the dedicated SFRs are mapped in the Arm® Cortex®-M0 system address range 58004000H - 58007FFFH.

Access of NVM module is granted through the AMBA matrix block that forwards to the memory modules AHB-Lite interfaces the requests generated by the masters according to the defined priority policy.

11.3.1 Definitions

This section defines the nomenclature and some abbreviations. the used flash memory is a non-volatile memory (“NVM”) based on a floating gate one-transistor cell. It is called “non-volatile” because the memory content is kept when the memory power supply is shut off.

11.3.1.1 General Definitions

Logical and Physical States

Erasing

The erased state of a cell is ´1´. Forcing an NVM cell to this state is called erasing. Erasing is possible with a granularity of a page (see below).

Writing

The written state of a cell is ´0´. Forcing an NVM cell to this state is called writing. Each bit can be individually written.

Programming

The combination of erasing and writing is called ‘programming’. Programming often means also writing a previously erased page.

The wording ‘write’ or ‘writing’ are also used for accessing special function registers and the assembly buffer. The meaning depends therefore on the context.

The above listed processes have certain limitations:

Retention: This is the time during which the data of a flash cell can be read reliably. The retention time is a statistical figure that depends on the operating conditions of the flash array (temperature profile) and the accesses to the flash array. With an increasing number of program/erase cycles (see endurance) the retention is lowered. Drain and gate disturbs decrease data retention as well.

Endurance: As described above, the data retention is reduced with an increasing number of program/erase cycles. A flash cell incurs one cycle whenever its page or sector is erased. This number is called “endurance”. As said for the retention, it is a statistical figure that depend on operating conditions and the use of the flash cells and on the required quality level.

Drain Disturb: Because of using a so called “one-transistor” flash cell each program access disturbs all pages of the same sector slightly. Over long these “drain disturbs” make 0 and 1 values indistinguishable and thus provoke read errors. This effect is again interrelated with the retention. A cell that incurred a high number of drain disturbs will have a lower retention. The physical sectors of the flash array are isolated from each other. So pages of a different sector do not incur a drain disturb. This effect must be therefore considered when the page erase feature is used or when re-programming an ready programmed page (implicitly causing an erase of the page before writing the new data).

Data Portions

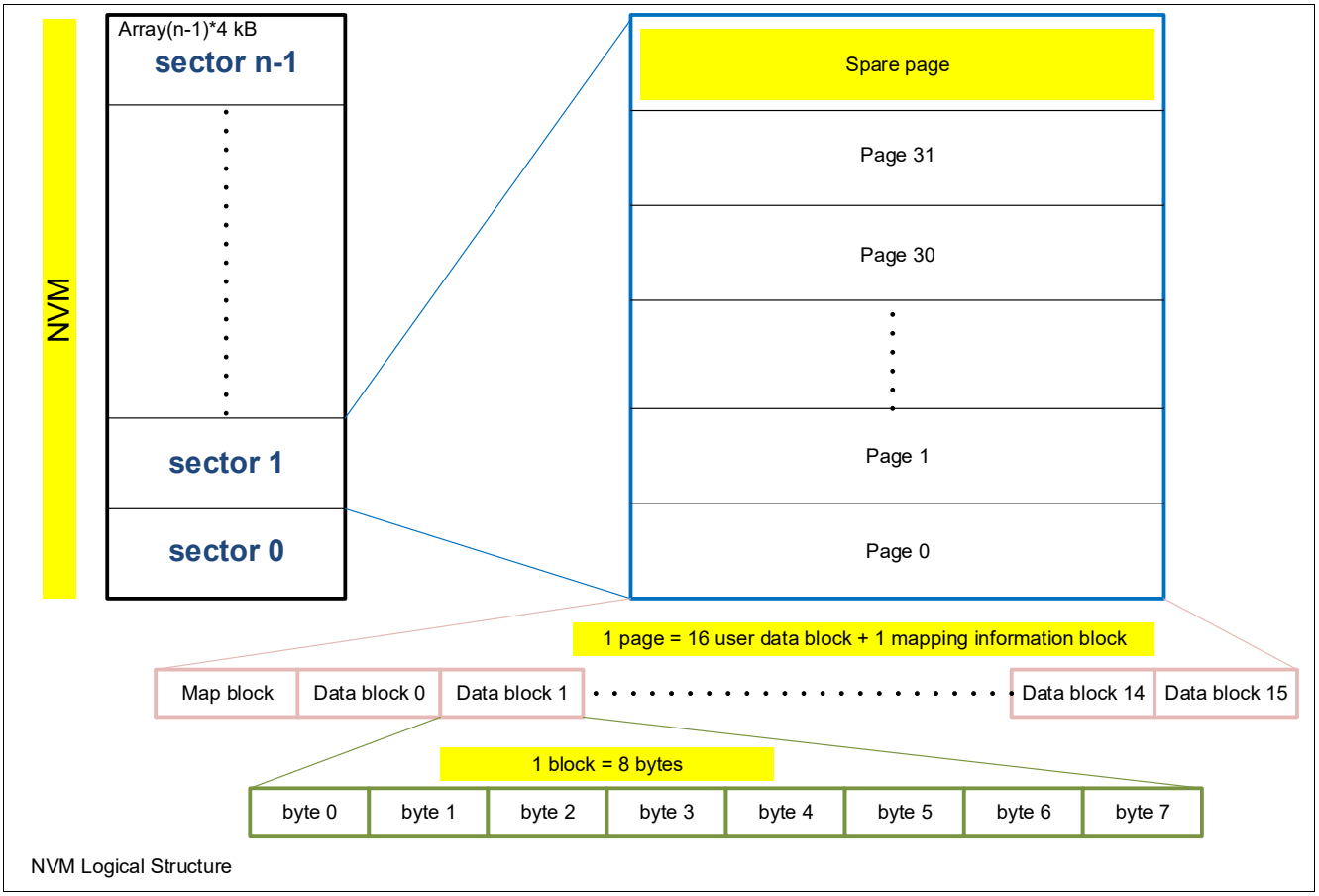


Figure 48 Logical Structure of the NVM Core

Doubleword

A doubleword consists of 64 bits. A doubleword represents the data size that is read from or written to the NVM core module within one access cycle.

Block

A block consists of one doubleword and its associated ECC data (64 bit data and 8 bit ECC). A block represents the smallest data portion that can be changed in the assembly buffer. Since the ECC protects 64 bits, when a byte is written to the assembly buffer automatically an NVM internal read of the complete block is triggered, the byte and the ECC are updated and the complete block is written back to the assembly buffer.

Mapblock

A map block consists of a module specific number of ECC-protected bits that hold the necessary information to map a physical page to a logical page.

Page

A page consists of 16 blocks and one map block.

Spare Page

A spare page is an additional page in a sector used in each programming routine to allow tearing-safe programming.

Sector

A sector consists of 32 logical and 33 physical page.

11.3.2 Functional Description

The main tasks of the NVM module are reading from the memory array, writing to the assembly buffer, enabling (tearing safe) programming of a single page, provide basic in-module functionality for code protection. The main features are listed following:

- Up to 96 KB (product variant dependent) memory size
- 3.5 ms write time per page
- 4.5 ms erase time per page
- Error correction and Error Detection code (ECC and EDC)
- In module memory protection logic

11.3.2.1 Basic Block Functions

Figure 49 shows a schematic block diagram of the NVM module

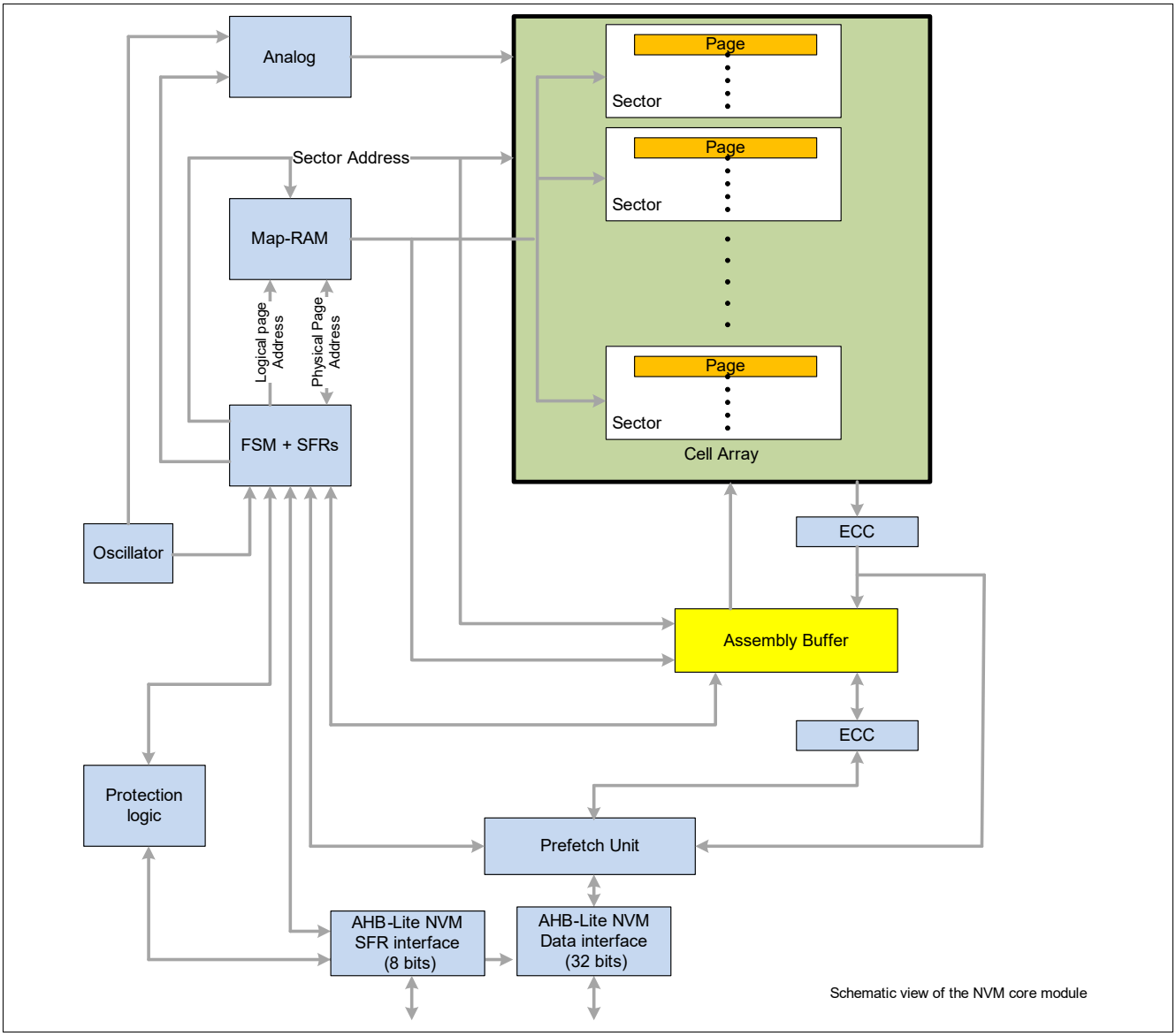


Figure 49 Schematic View of the NVM Core Module

11.3.2.2 Memory Cell Array

The non-volatile memory cells are organized in sectors, which consists of pages, which are structured in blocks and map block.

Page

Each page consists of 16 data blocks of 64 bits each and one map block. The map block stores the mapping information of the page in the sector. All blocks of a page are ECC-protected.

A page is the smallest granularity of data that can be changed (erased or written) within the cell array. One data block is the minimum granularity of data that can be read from the NVM module within memory read access.

Employing the integrated EEPROM emulation using the map RAM, the minimum granularity of data that can be changed in the NVM is one byte, while all other bytes in the page do not change.

Assembly Buffer

The assembly buffer is a RAM that can hold the content of one page including the mapblock.

Sector

A sector consists of 33 physical pages. 32 pages can be logically addressed during a memory access. One page is internally used as a spare page.

Map RAM

The map RAM is a static RAM that holds the mapping of a logical page addresses to physical page addresses for each mapped sector. It is completely handled by the NVM programming related BootROM routines.

FSM and SFR block

This block contains the special function registers (SFRs) of the NVM module.Beside memory reads and writes to the assembly all interactions of the BootROM software with the module take places through register accesses. The finite state machine (FSM) controls the actions (e.g. read, erase and write) of the NVM module.

Analog components

The module contains analog components to provide all the voltages necessary for erasing, writing and reading the non-volatile memory cells.

11.3.2.3 SFR Accesses

All SFRs can only be accessed through the NVM related BootROM routines, that is, the customer software cannot access the SFRs directly but has to use BootROM routines.

11.3.2.4 Memory Read

The NVM memory internally can be read with a minimum granularity of one block (64 data bits).

If the block is not within the memory address range of the NVM module, the module does not react at all and a different memory module may handle the access.

Memory read accesses are only possible while no FSM procedures (program, init, sleep or copy) is in progress. A memory read access while the FSM is busy is stalled as long as the FSM is busy and the access is carried out when the FSM is in idle mode again.

Since a read to the memory field takes a fixed time mostly independent of the system frequency, an optimized number of waitstates (3, 1 or, 0) is generated for different system frequencies.

Furthermore, a module internal read buffer holds the block read last. An access to an address within this block does not trigger a new reading from the memory field but is directly served from the read buffer.

11.3.2.5 Memory Write

Data is not written to the memory array directly, but to the assembly buffer and then copied into the cell array by the write sequence.

Memory writes are handled through the BootROM software, which at first copies the existing content of a page to the assembly buffer, allows the user to modify the content of the assembly buffer and afterwards executes the programming of the data to the memory field followed by a verification step.

11.3.2.6 Timing

The target timing of the hardware sequences excluding the software overhead is shown below:

- Erase: typ. 4 ms per page
- Write: typ. 3 ms per page
- Program (= Erase+ Write): typ. 7 ms per page

The disturb handling routine, when enabled, with a probability of approximately 0.1% adds additional typ. 7 ms to a page write or program operation.

11.3.2.7 Verify

The data programmed by the BootROM function is verified by the BootROM routine itself. The programmed data in the cell array is compared with the data still available in the assembly buffer. This is done using suitable hard-read levels. These hard-read levels provide a margin compared to the normal read level to ensure that the data is actually programmed with suitably distinct levels for written and erased bits.

11.3.2.8 Tearing-Safe Programming

The mapping mechanism of the NVM module is used like a log-structured file system: When a page is programmed in the sector the old values are not physically overwritten, but a different physical page (spare page) is programmed in the same sector in fact. If the programming fails (e.g. because of power loss during the erase or write procedure), the old values are still present in the sector. The BootROM routines therefore can program a single page in a tearing-safe way.

When an erase or write procedure to the memory field was interrupted by a power-down, this is identified during the reconstruction of the map-RAM content after the next reset. In this case, a special routine in the BootROM (called Service Algorithm) is automatically started, identifies this tearing case of respective logical page and repairs the NVM state, ensuring that either the old or the new data (or both) are fully valid.

11.3.2.9 Disturb Handling

Due to the implementation of the cell array, while writing a page into the cell array all other pages within the same sector are slightly written (disturbed) too. If some pages of a sector are changed often and other pages of the same sector only rarely, these rarely programmed pages may be disturbed too often and loose their data.

If the disturbs for a page exceed a specific value (this happens only when a different page in the same sector is programmed), the page has to be reprogrammed (refreshed). A dedicate option of the programming routines provided with the BootROM make sure that the pages are refreshed in time.

As mentioned, the refreshing of a page - when actually triggered - will double the overall programming time.

11.3.2.10 ECC and EDC

The NVM module provides all needed logic for proper error correction and detection logic. Since the block is the smallest data portion used for accessing the array in read and write, the ECC and EDC are performed at

block level. Requirement is to provide a single bit ECC and 2 bits EDC per block, that is 1 bit correction over 64 data bits.

Since the ECC protects 64bits, when a byte is written to the assembly buffer automatically an NVM internal read of the complete block is triggered, the byte and the ECC are updated and the complete block is written back to the assembly buffer.

11.3.2.11 Code and Data Access through the AHB-Lite Interface

The system provides access to the data stored in the NVM cell array through an AHB Lite interface. Whenever the core needs to fetch instructions or read data from or write data into the NVM module, a proper AHB Lite compliant access request is forwarded by the bus matrix block into the module.

11.4 BootROM Module

The TLE985xQX BootROM module provides physical implementation of the memory module as well as needed complementary features and interface towards the core.

The module provides proper access through a 32-bit AHB-Lite data interface multiplexed on Arm® Cortex®-M0 system bus for code/data access.

The BootROM module in TLE985xQX has a capacity of 24 Kbyte, organized with words of 32 bits.

The BootROM content consists basically of three parts, used for:

- Startup and boot SW
- Boot Strap Loader routines
- User routines

11.4.1 BootROM Addressing

The BootROM, as visible from the memory map, is mapped starting at the address range 00000000H - 00005FFFH.After any reset, the device hardware-controlled start address is 00000000H. At this location, the default VTOR to be used shall be stored.

11.4.2 BootROM Firmware Program Structure

The BootROM firmware provides basic functionality required to be executed after reset and routines for specific operation, such as:

- Startup routines, which is the main control firmware in the BootROM executed after every reset. This routine checks which kind of reset was issued and accordingly performs different kinds of operation to proper configure the device.
- Bootstrap loader, which provides basic functionality for code and data upload via LIN or UART into the RAM or NVM module.
- User routines, which provide functions for proper NVM operation handling and other useful ready-to -use routines designed for the customer.

11.5 RAM Module

The TLE985xQX RAM module provides physical implementation of the memory module as well as needed complementary features and interface towards the core.

The module provides proper access through a 32-bit AHB-Lite data interface multiplexed on Arm® Cortex®-M0 system bus for code/data access.

The RAM module in TLE985xQX has a capacity of 4 Kbyte, organized with words of 32 bits.

The module supports 1 bit error correction and 2 bits error detection per 32-bit word (actually requiring 7 bits parity per word). When an ECC error occurs, the corresponding status flag in the register EDCSTAT will be set. A double bit error can be configured via the interrupt enable bit in register EDCCON to trigger an exception.

11.5.1 RAM Addressing

The RAM, as visible from the memory map, is mapped at the address range 18000000H - 18000FFF. The module is mapped in the code area of the Arm® Cortex®-M0 map regions and can be used as program memory for code fetching as well as data storing.

11.6 Memory protection Unit (MPU)

The target of the memory protection scheme is to prevent unauthorized read out of critical data and user IPs from the BootROM and NVM as well as to prevent accidental memory data modification.

The TLE985xQX protection scheme is divided in 2 parts interacting togetherTLE985xQX

The first memory protection scheme is firmware based and involves the blocking of all external access to the device. More information on the firmware based protection scheme can be found in **Chapter 11.6.3**

The second memory protection scheme is hardware based; The “source” address, from which a memory read instruction is fetched, and the “target” address, where addressed data are stored, are checked by the Memory Protection Unit (MPU) to determine if the access must be blocked. Read instructions executed from an unsafe memory address (e.g. RAM) that target the BootROM or NVM are blocked when the respective protection mode is enabled. The hardware protection scheme is further described in Section **Chapter 11.6.2**.

11.6.1 BootROM

The TLE985xQX shall provide the following protection regions:

- BootROM region
- Customer BSL NVM regionBootROM
- Code Linear NVM region
- Data Linear NVM region
- Data Mapped NVM region[1)]

The protection scheme implemented for the NVM memory module shall support 4 different protection regions. On each region the protection feature can be enabled or disabled independently according to the mechanism and limitation further explained in the **Chapter 11.6.2.2**BootROM

The **Figure 50** shows the NVM memory regions supported by the protection mechanism.

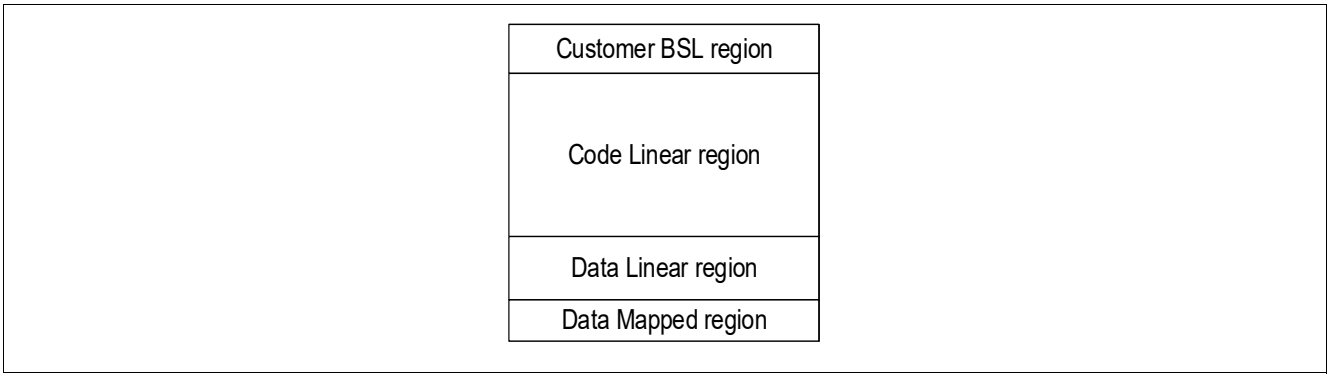


Figure 50 BootROM Protection Regions

11.6.2 Hardware Protection Mode

The hardware protection mode controls the access right on each memory or memory region available. Every access to any memory is checked against the memory protection settings and accordingly executed or rejected.

For the TLE985xQX, the BootROM protection mode is always enabled (hardware default) and it can never be disabled. The NVM protection modes can instead be enabled separately for Customer BSL, Code Linear, Data

1) Data mapped NVM region is identical to the terminology non-linear NVM region

Linear and Data Mapped regions. While the BootROM protection mode is enabled, the NVM protection mode may be enabled as well to further prevent code read out.

NVM has privilged region protection. Customer BSL region is considered to have the highest privilege, followed by Code region, then Data region. The higher privilge region can always read lower privilege regions regardless protection.

Therefore, regardless the protection mode enabling, the following data reading accesses shall always be possible:

- Data reading instructions executed from the BootROM targeting BootROM itself or the RAM
- Data reading instructions executed from the Customer BSL NVM region targeting Customer BSL NVM region itself, Code Linear NVM region, Data Linear NVM region, Data Mapped NVM region or RAM
- Data reading instructions executed from the Code Linear NVM region targeting Code Linear NVM region itself, Data Linear NVM region, Data Mapped NVM region or RAM
- Data reading instructions executed from the Data (Linear or Mapped) NVM region targeting RAM
- Data reading instructions executed from the RAM targeting RAM itself

Unauthorized data reading instructions will be detected and consequently blocked.

11.6.2.1 BootROM Protection Mode

The BootROM data read protection modes shall be enabled by default and consequently the following accesses shall be restricted:

- Data reading instructions executed from the NVM, or RAM targeting BootROM

Figure 51 shows all the data reading instructions authorized when only the BootROM data read protection is enabled (NVM protection disabled).

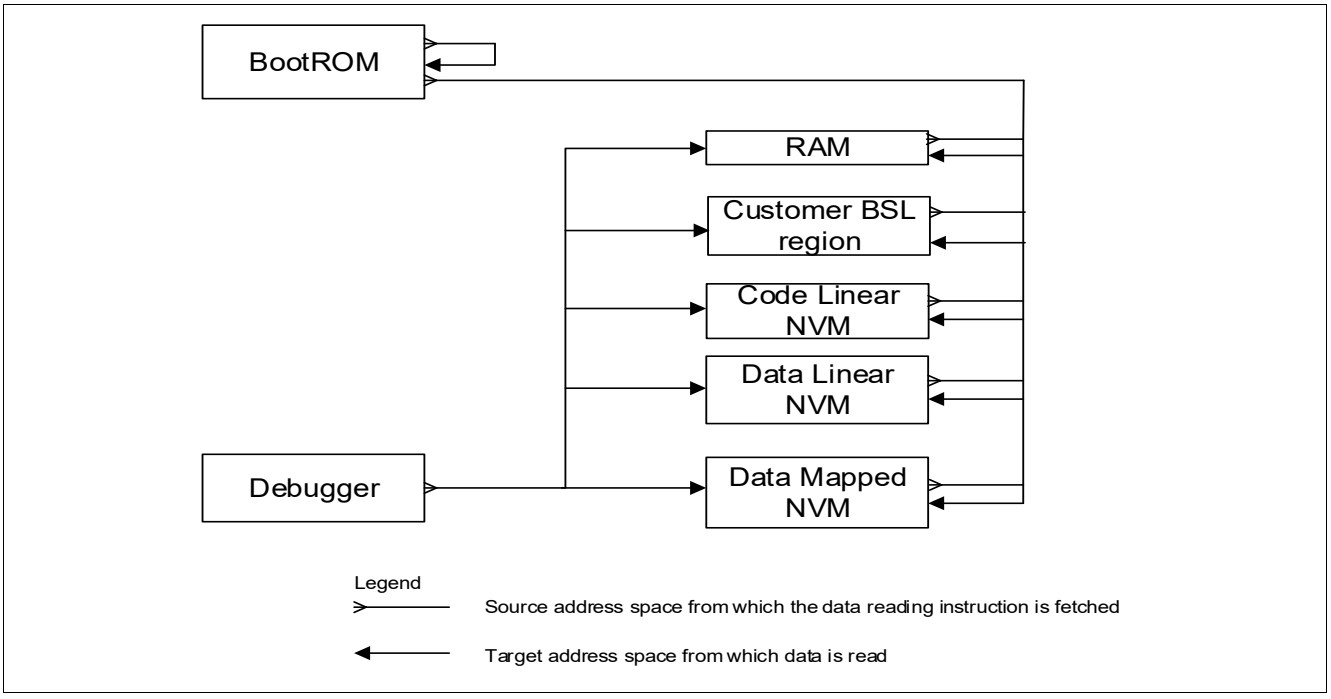


Figure 51 BootROM Data Read Protection Mode enabled

If the BootROM data read protection mode is enabled without enabling of any NVM protection mode

- Data reading instructions executed from NVM or RAM can target itself or one another

- Data reading instructions executed from the BootROM can target itself, NVM or RAM
- Data reading access issued by the debugger can target NVM or RAM

In addition, to avoid an indirect leak of information by hacking through the debugger, breakpoints set and step through features shall be disabled on the BootROM. In case debugger issues such a command, the command is suspended till the moment in which the code execution leaves the read protected region (BootROM). More information about protection against debugger activity can be found in **Chapter 11.7**

11.6.2.2 NVM Protection Modes

The NVM address space is divided into the for supported NVM regions: Customer BSL, Code Linear, Data Linear and Data Mapped region.

The Customer BSL region is supposed to be used for special user code that might not be changed over device life time. Since this region is anyhow meant to host user executable code, the region is linearly mapped even if, to distinguish it from standard user code region, it is named “Customer BSL”.

The Code Linear region is supposed to be used for user standard application code while the Data Linear and Data Mapped region is meant to be used for data storage even if code execution is not prevented.

The protection on each of the region is individually controlled by the setting of the NVM_PROT_STS register bits.Further details regarding the NVM region protection enable/disable are described in the **Chapter 11.6.2.2.5**

11.6.2.2.1 BootROM Protection Mode

The Customer BSL Region protection can be controlled via proper dedicated password as described in the **Chapter 11.6.2.2.5**.

When its write protection is enabled, any operation capable to change the NVM values stored in this region shall be blocked. For example, neither a program nor an erase can be executed.

In case the memory protection unit (MPU) and NVM control logic detect that the target address belongs to this region and that write protection is set, a proper alarm signal is forwarded to the NVM module to prevent the NVM state machine from accepting any program or erase command. This prevents inadvertent destruction of stored data when protection is set.

When Customer BSL region read protection is enabled, the following accesses shall be restricted:

- Data reading instructions executed from any other memory region (BootROM, RAM, Code Linear NVM, Data Linear NVM and Data Mapped NVM) RAMtargeting the Customer BSL region
- Data reading accesses triggered by debugger targeting the Customer BSL region

Figure 52 shows all the data reading instructions authorized when both the BootROM and Customer BSL Region read protections are enabled.

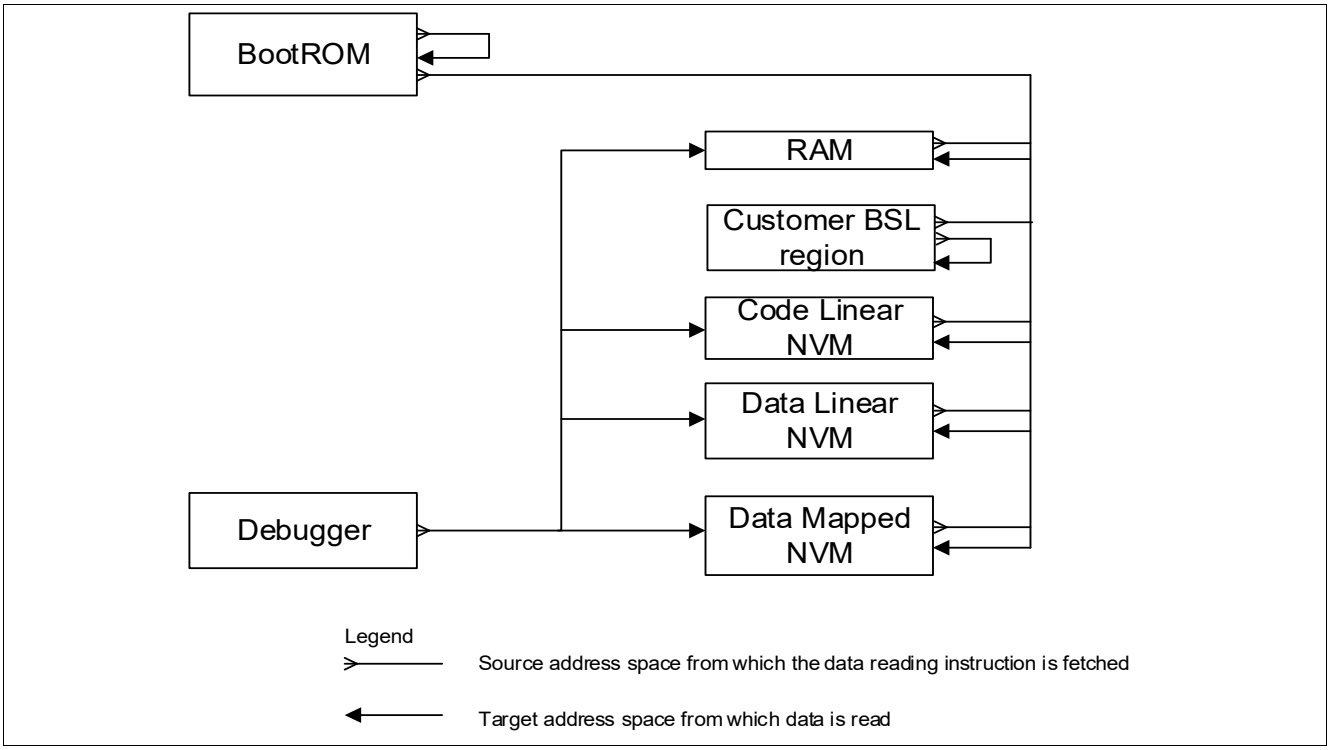


Figure 52 BootROM Protection Mode enabled

If the BootROM and the Customer BSL protection modes are enabled:

- Data reading instructions executed from the Code Linear NVM, Data Linear NVM, Data Mapped NVM or RAM can target itself or one another
- Data reading instructions executed from the BootROM can target itself, Code Linear NVM, Data Linear NVM, Data Mapped NVM or RAM
- Data reading instructions executed from the BootROM can target itself, Code Linear NVM, Data Linear NVM, Data Mapped NVM or RAM

11.6.2.2.2 BootROMProtection Mode

The NVM Code Linear protection can be controlled via proper dedicated password or via the NVMPROT_STS register as described in the **Chapter 11.6.2.2.5**.

When its write protection is enabled, any operation capable to change the NVM values stored in this region shall be blocked. For example, neither a program nor an erase can be executed.

Regarding write protection, the 100TP pages are considered to be part of the Code Linear NVM. For this reason, in case the write protection in this region is set, even the 100TP program is blocked.

In case the MPU and NVM control logic detect that the target address belongs to this region and that write protection is set, a proper alarm signal is forwarded to the NVM module to prevent the NVM state machine from accepting any program or erase command. This prevents inadvertent destruction of stored data while protection is set.

When NVM Code Linear read protection is enabled, the following accesses shall be restricted:

- Data reading instructions executed from other memory region (BootROM, RAM, Data Linear NVM or Data Mapped NVM) RAMtargeting the NVM Code Linear region

- Data reading accesses triggered by debugger targeting the NVM Code Linear region

Figure 53 shows all the data reading instructions authorized when the BootROM, the Customer BSL region and NVM Code Linear read protections are enabled.

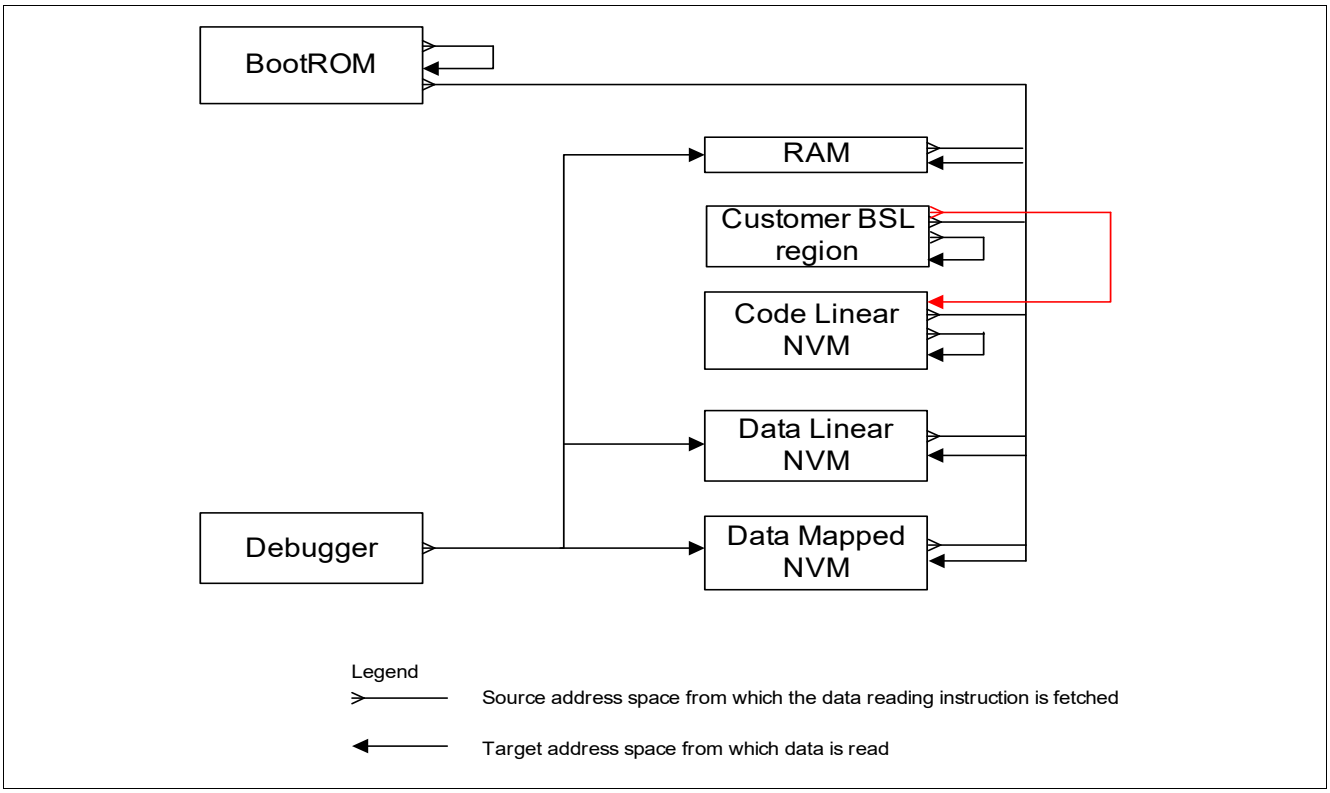


Figure 53 BootROMProtection Mode enabled

If the BootROM, the Customer BSL and the NVM Code Linear protection modes are enabled:

- Data reading instructions executed from the Data Linear NVM, Data Mapped NVM or RAM can target itself or one another
- Data reading instructions executed from the BootROM can target itself, Data Linear NVM, Data Mapped NVM or RAM
- Data reading instructions executed from the BootROM can target itself, Code Linear NVM, Data Linear NVM, Data Mapped NVM or RAM
- Data reading instructions executed from the BootROM can target itself, Data Linear NVM, Data Mapped NVM or RAM

11.6.2.2.3 BootROMProtection Mode

The NVM Data Linear protection can be controlled via proper dedicated Password or via the NVMPROT_STS register as described in the **Chapter 11.6.2.2.5**.

When its write protection is enabled, any operation capable to change the NVM values stored in this region shall be blocked. For example, neither a program nor an erase can be executed.

In case the MPU and NVM control logic detect that the target address belongs to this region and that write protection is set, a proper alarm signal is forwarded to the NVM module to prevent the NVM state machine

from accepting any program or erase command. This prevents inadvertent destruction of stored data while protection is set.

When NVM Data Linear read protection is enabled, the following accesses shall be restricted:

- Data reading instructions executed from BootROM, RAM, Data Mapped NVM RAMtargeting the NVM Data Linear region
- Data reading accesses triggered by debugger targeting the NVM Data Linear region

Hardware supports code execution from data region. Data reading instruction executed from data region is always allowed to target itself.

Figure 55 shows all the data reading instructions authorized when the BootROM, the Customer BSL region, NVM Code Linear, NVM Data Linear read protections are enabled.

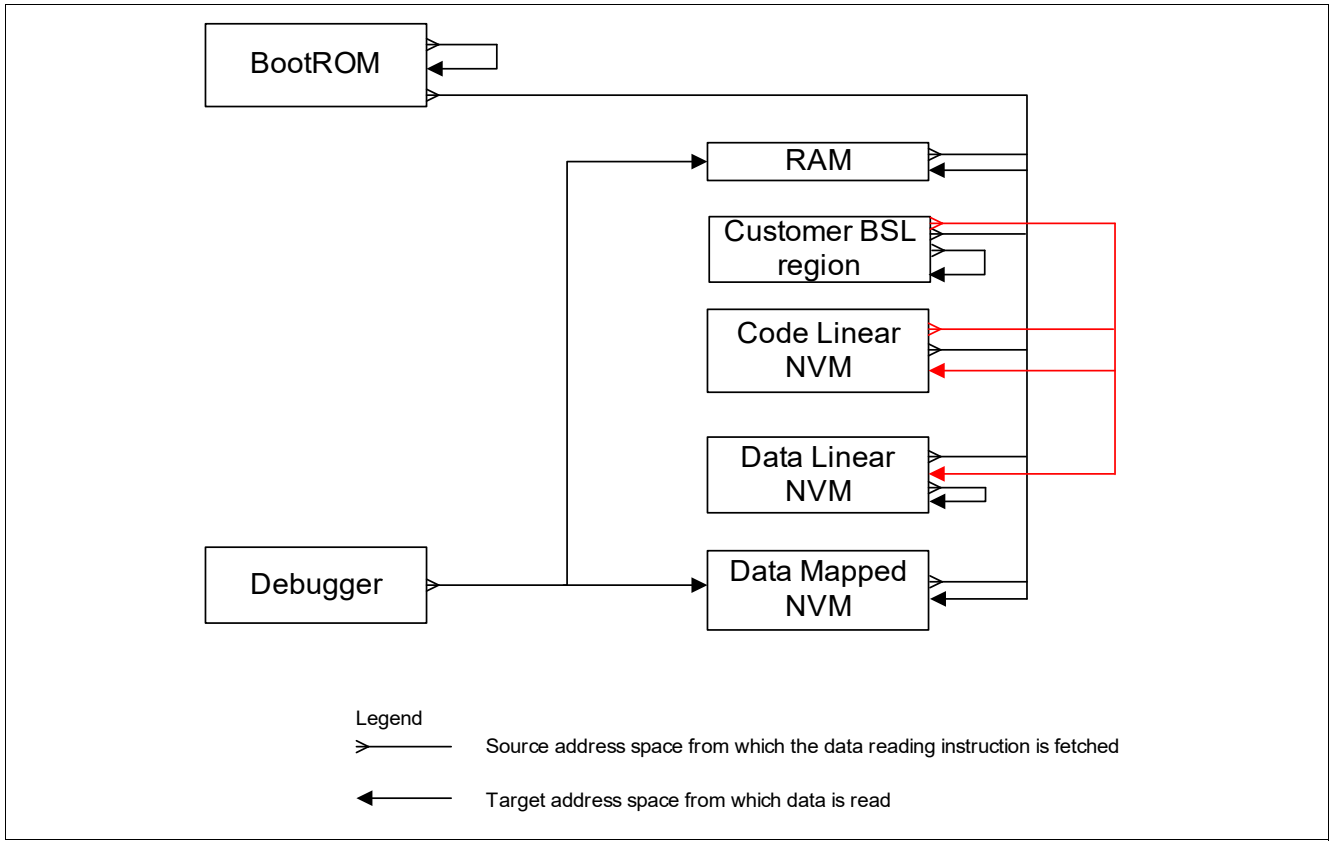


Figure 54 BootROM Protection Mode enabled

If the BootROM, the Customer BSL, the NVM Code Linear and the NVM Data Linear protection modes are enabled:

- Data reading instructions executed from the Data Linear NVM can target itself, Data Mapped NVM or RAM
- Data reading instructions executed from the Data Mapped NVM or RAM can target itself or one another
- Data reading instructions executed from the BootROM can target itself, Data Mapped NVM or RAM
- Data reading instructions executed from the BootROM can target itself, Code Linear NVM, Data Linear NVM, Data Mapped NVM or RAM
- Data reading instructions executed from the BootROM can target itself, Data Linear NVM, Data Mapped NVM or RAM

11.6.2.2.4 BootROMProtection Mode

The NVM Data Mapped protection can be controlled via proper dedicated Password or via the NVMPROT_STS register as described in the **Chapter 11.6.2.2.5**.

When its write protection is enabled, any operation capable to change the NVM values stored in this region shall be blocked. For example, neither a program nor an erase can be executed.

In case the MPU and NVM control logic detect that the target address belongs to this region and that write protection is set, a proper alarm signal is forwarded to the NVM module to prevent the NVM state machine from accepting any program or erase command (including fast invalidation). This prevents inadvertent destruction of stored data while protection is set.

When NVM Data Mapped read protection is enabled, the following accesses shall be restricted:

- Data reading instructions executed from BootROM, RAMand Data Linear NVM RAMtargeting the NVM Data Mapped region
- Data reading accesses triggered by debugger targeting the NVM Data Mapped region

Hardware supports code execution from data region, data reading instruction executed from data region is always allowed to target itself.

Figure 55 shows all the data reading instructions authorized when the BootROM, the Customer BSL region, NVM Code Linear, NVM Data Linear and NVM Data Mapped read protections are enabled.

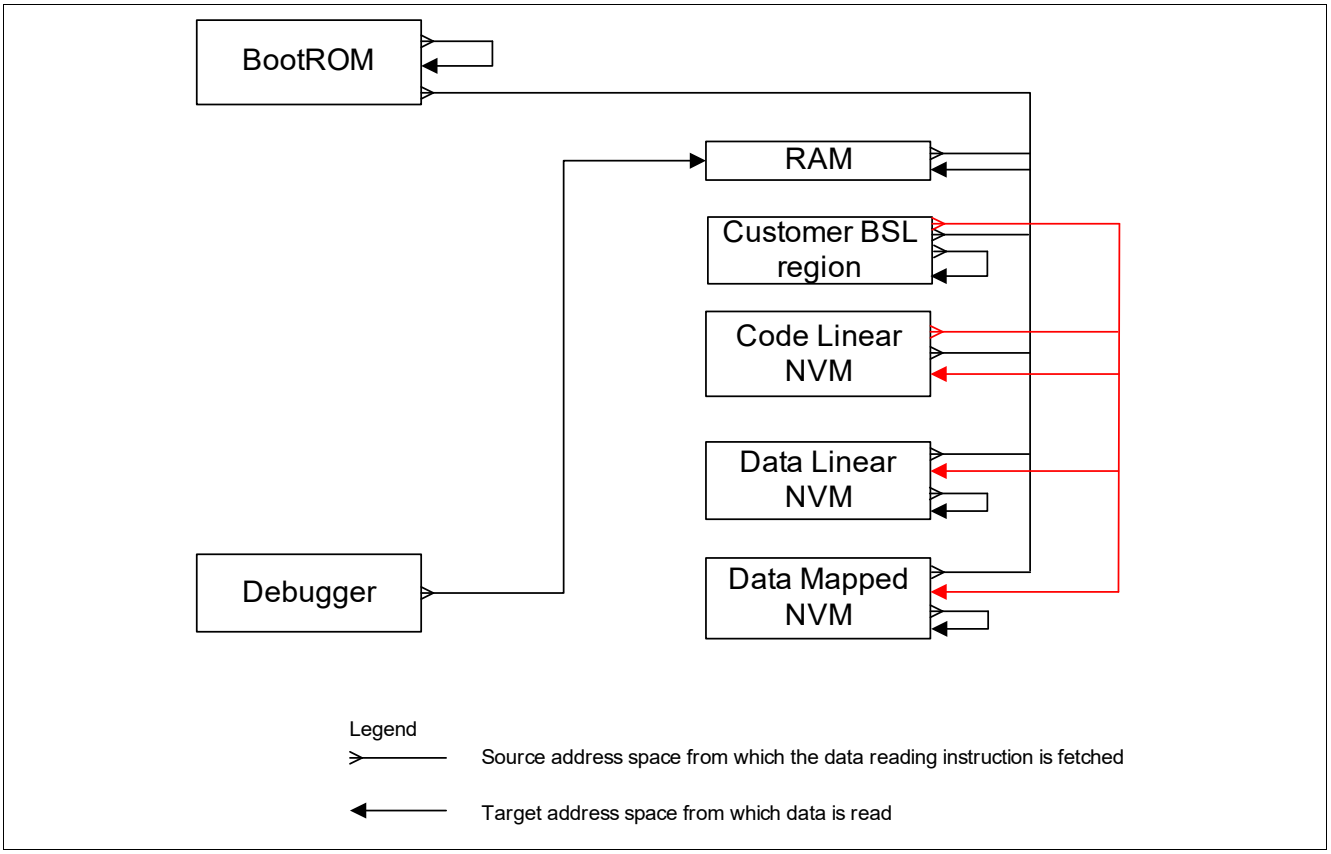


Figure 55 BootROM Protection Mode enabled

If the BootROM, the Customer BSL, the NVM Code Linear, NVM Data Linear and the NVM Data Mapped protection modes are enabled:

- Data reading instructions executed from Data Linear NVM can target itself or RAM

- Data reading instructions executed from Data Mapped NVM can target itself or RAM
- Data reading instructions executed from the BootROM can target itself or RAM
- Data reading instructions executed from the BootROM can target itself, Code Linear NVM, Data Linear NVM, Data Mapped NVM or RAM
- Data reading instructions executed from the BootROM can target itself, Data Linear NVM, Data Mapped NVM or RAM
- Data reading instructions executed from RAM can target itself

11.6.2.2.5 NVM Protection mode control

The read and write protection on the different regions are controlled via the register NVM_PROT_STS.

The value of this register can be changed in 2 different ways.

Protection Password

The first method is based on a region specific protection password. After the complete code has been programmed into the Customer BSL and Linear NVM regions, the protection scheme can be enabled by calling the BSL command: password set. The BSL command programs a user provided password into the reserved space if no password installed before. Upon the next reset, the BootROM startup routine will read out the stored user-defined password. If its value is valid (bit 29:0 neither 00000000_H nor $3FFFFFFF_H$), the user password is taken as programmed and the protection on the specific region is set. The read and program protection modes are set writing the related bits of the NVM_PROT_STS[1] register according to the information stored into the 2 most significant bits of the password. The format of the password is shown in the **Figure 56**.

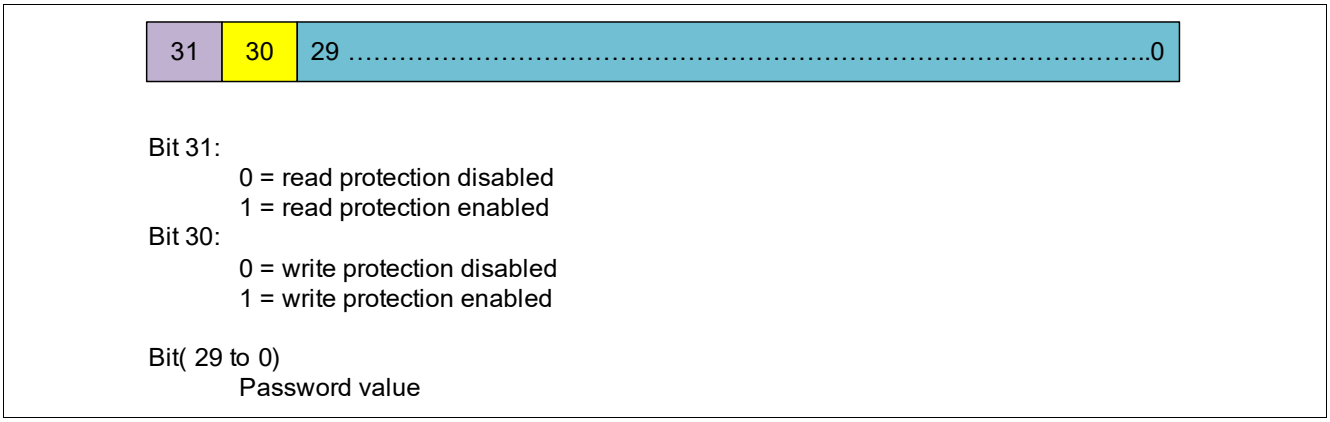


Figure 56 BootROM

In order to prevent hacking the password by repetitive trials, there is no BSL command or USER API available to clear password.

There shall be a password for each region.

1) For a complete description of the NVM_PROT_STS register, refer to the TLE985xQX User Manual

Temporary Protection

The hardware memory protection mechanism is controlled by the values of the NVM_ PROT_STS register bits. When user set a protection via password, the BootROM startup sequence enables proper protection modes by writing the related bit of the NVM_PROT_STS register.

Even if user enables write protection on a defined region at startup using the dedicated password, during the application code execution there might be the need to temporarily disable the write protection to store some new code/data.

For example, user might want to set by default at startup the write protection on the Data Linear NVM or Data Mapped NVM region to avoid accidental data loss. Nevertheless, during application code execution, there might be the need to update some of the data stored in this region. For this reason, the TLE985xQX provides the user the possibility to change the protection status writing directly the NVM_PROT_STS bits. The changes in the active protection scheme obtained via direct access to the register are anyhow temporary and the default protections controlled by password status will be automatically restored at the next reset (next BootROM startup sequence execution).

There is no user API availble to set/clear read protection. Two reasons:

- due to privileged NVM region protection, higher privilge region can always read lower privilege regions, there is no application scenario to change read protection at run time.
- the user API: read protection clear is considered to be a risk, which could be used by hacker to disable read protection to read out IP.

Anyhow, to safeguard against accidental access by user on this register, its access is controlled depending on boot mode, memory regions protections status and source address.

The user APIs shall:

- individually set/clear write protection on each memory protection region(apart from customer BSL region)
- Freely set/clear write protection as long as no valid password for the target region has been installed.
- In case a valid password for the target region is installed, the routine has to take the current valid password as input. If the provided password matches the current valid installed one, the target NVM_PROT_STS bits can be freely changed. In case, instead, the password provided as input does not match the current installed one, the NVM_PROT_STS target bits shall not be changed.(the hacker is still possible to get password by repetitive trials, but even they know, there is no way to remove password or read protection, the IP protection is still considered to be valid)

The above reported feature and routines applies in general for all the different memory protection regions. Exceptions:

- Customer BSL protection region shall be controllable only via password.

11.6.3 Firmware protection mode

The firmware protection scheme is the second leg of the overall memory protection concept.

In particular, the BootROM code provides following features:

- Each BootROM routine provided by the firmware for the NVM data handling (e.g Program or Erase routines) shall check the address to identify which region is targeted and accordingly check the relevant bit of the NVM_PROT_STS register. In case the write protection for the target region is not set, the operation is executed. In case, instead, the write protection for the target region is set, the routine exits and reports a proper error.

- In case read protection is enabled on selected protection region (Customer BSL, Code Linear NVM, Data Linear NVM or Data Mapped NVM), all provided feature to download code into the selected region shall be blocked (for example all BSL modes available to download code into the selected regiom).

The firmware protection features are provided to complete the protection scheme. The first implemented feature is to ease the detection of any BootROM routine fails due to the protection setting. In fact, in case a BootROM routine is called with write protection enabled, the routine would not affect the NVM content due to the hardware protection scheme. In such a case, the BootROM based protection feature would recognize in firmware the protection settings and stop the routine providing a proper fail indication to the user code.

The second firmware based protection feature is instead needed to make the read protection mechanism provided by hardware effective. In fact, the feature for code download could be used for hacking even if the read protection is set on a region (but not the write protection). It would then be possible to read out the code/data by downloading a proper code into the same region. In fact, according to the hardware protection scheme, a code running from a selected region can always address itself. So, the Firmware will block all the boot options such that it is not possible to load and execute any external code, but only to execute user code starting at address pointed by the standard reset handler routine address stored at 11000004_H.BootROM

11.7 Core Protection Mode

The **Chapter 11.6.2** and **Chapter 11.6.3** describe the protection against accidental write or malicious read memory access implemented in hardware and firmware. The hardware implements a check of all direct access to the each memory region (even from debugger) granting access only when the target region is not protected. The firmware, instead, blocks BSL NVM download in case selected region read protection is installed to prevent installing any malicious software into the selected region that removes the protection and reads out the user code.

Without any further feature, there would still be the possibility to use the debugger to leak information about user code. In fact, even if the read out of the memory content via debugger is blocked when accessing a read protected region, it is still possible to use the other debugger features (e.g. step through, breakpoints, watchpoints, code profiling) to perform a reverse engineering of executed code.

For this reason, a further level of protection is implemented between the MCU and the Core.

In particular, the debugger features are disabled according to the current program counter and the installed passwords.

By default, when no password is installed, the debug features are disabled while executing from the BootROM thus avoiding any code profiling.

If read protection is applied to any NVM region, the SWD debugger connection is blocked and no debug commands can be sent to the debug unit anymore. This is done internally, if any RD_EN bits of NVM_PROT_STS is disabled, SWD data bus is forced to 0.

12 Interrupt System

12.1 Features

- 23 interrupt nodes for on-chip peripherals
- 8 NMI nodes for critical system events
- Maximum flexibility (resp. priority and node grouping) for all interrupt nodes

12.2 Introduction

12.2.1 Overview

The TLE985xQX supports 24 interrupt vectors with 4 priority levels. 21 of these interrupt vectors are assigned to the on-chip peripherals: GPT12, SSC1, SSC2, CCU6, High-Side Switch, WAKEUP, Bridge Driver, Charge Pump, Differential Unit, Math Divider, GPIOs, MONs, CSA and A/D Converter are each assigned to one dedicated interrupt vector; while UART1 and Timer2 or UART2, External Interrupt 2 and Timer21 share interrupt vectors. Two vectors are dedicated for External Interrupt 0 and 1.

A non-maskable interrupt (NMI) with the highest priority is shared by the following:

- Watchdog Timer, warning before overflow
- MI_CLK Watchdog Timer overflow event
- PLL, loss of lock
- Oscillator watchdog detection for too low oscillation of f_{OSC}
- Flash map error
- Uncorrectable ECC error on Flash and RAM
- VSUP supply prewarning when any supply voltage drops below or exceeds any threshold or when PMU temperature exceeds a certain limit.
- Overtemperature prewarning when system temperature exceeds a certain limit.
- Stack overflow

Figure 74 gives the corresponding overview for the NMI sources. The table below shows the available interrupt vectors.

Table 170 Interrupt Vector Table

Service Request	Node ID	Description
GPT1	0	GPTimer 1 Interrupt
GPT2	1	GPTimer 2 Interrupt
MU	2	MU interrupt / ADC2, VBG interrupt
ADC1	3	ADC10 Bit interrupt
CCU0	4	CCU6 node 0 interrupt
CCU1	5	CCU6 node 1 interrupt
CCU2	6	CCU6 node 2 interrupt
CCU3	7	CCU6 node 3 interrupt

Table 170 Interrupt Vector Table (cont'd)

Service Request	Node ID	Description
SSC1	8	SSC1 interrupt (receive, transmit, error)
SSC2	9	SSC2 interrupt (receive, transmit, error)
UART1	10	UART1 interrupt (receive, transmit), Timer2, LIN sync, LIN
UART2	11	UART2 interrupt (receive, transmit), Timer21, External Interrupt (EINT2)
EXINT0	12	External interrupt (EINT0), wake-up
EXINT1	13	External interrupt (EINT1)
WAKEUP	14	Wake-up interrupt (generated by a wake-up event)
Math Div	15	Hardware Divider Unit Interrupt
rfu	16	Reserved for future use
CP	17	Charge Pump
BDRV	18	Bridge Driver
HS	19	High Side Interrupt
CSA	20	Current Sense Amplifier Overcurrent Measurement
DU	21	Differential Unit - DPP1
MONx	22	MONx Interrupt
Port 2.x	23	Port 2.x - DPP1

Table 171 NMI Interrupt Table

Service Request	Node	Description
Watchdog Timer NMI	NMI	Watchdog Timer overflow
MI_CLK Watchdog Timer NMI	NMI	MI_CLK Watchdog Timer Overflow
PLL NMI	NMI	PLL Loss-of-Lock
Overtemperature NMI	NMI	System Overtemperature
Oscillator Watchdog NMI	NMI	Oscillator Watchdog
NVM Map Error NMI	NMI	NVM Map Error
ECC Error NMI	NMI	RAM / NVM Uncorrectable ECC Error
Supply Prewarning NMI	NMI	Supply Prewarning
Stack overflow	NMI	Stack Overflow

12.3 Functional Description

12.3.1 Interrupt Node Assignment

12.3.1.1 Interrupt Node 0 and 1 - GPT12 Timer Module

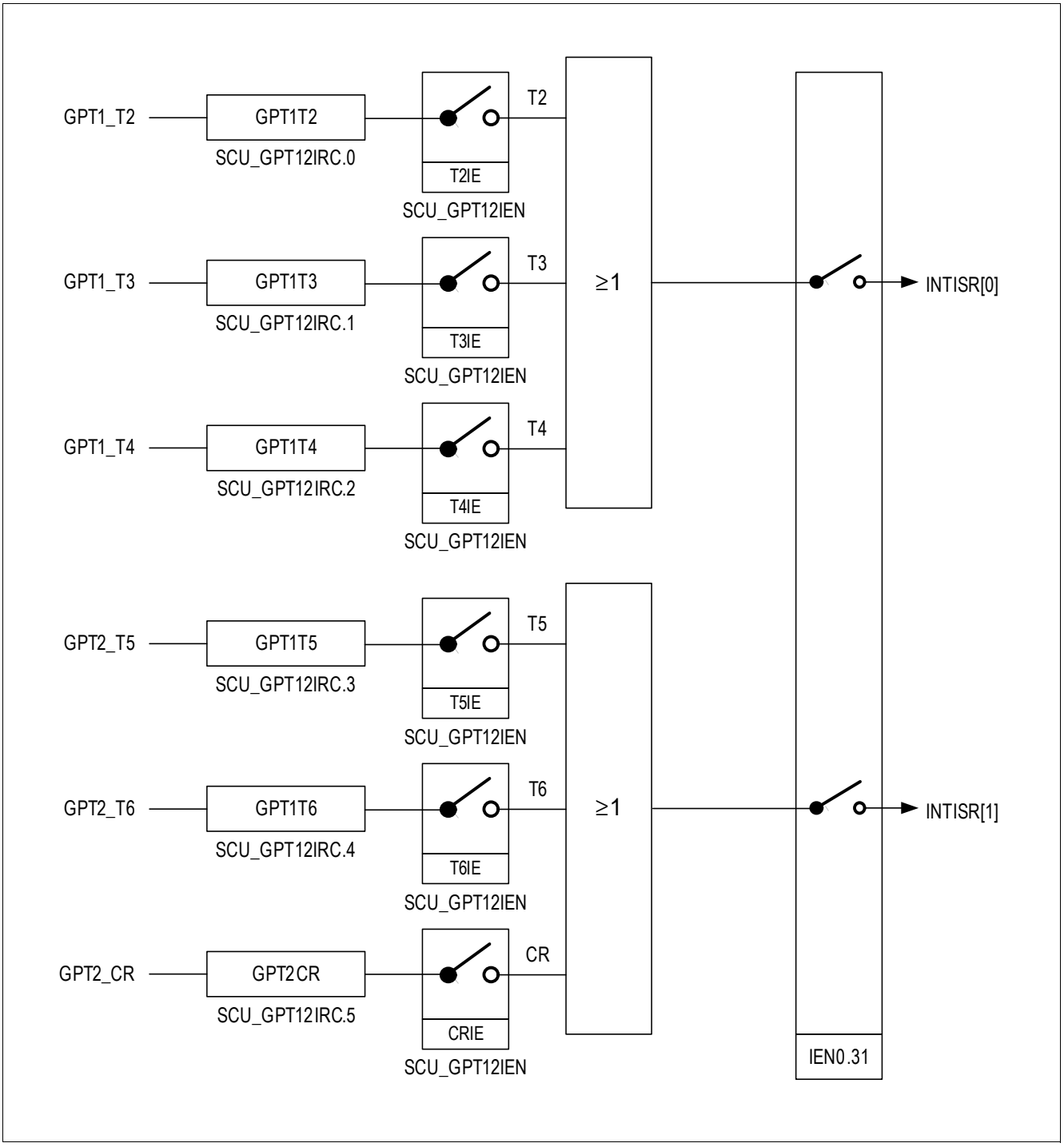


Figure 57 Interrupt Request Sources 0 and 1 (GPT12)

12.3.1.2 Interrupt Node 2 - Measurement Unit

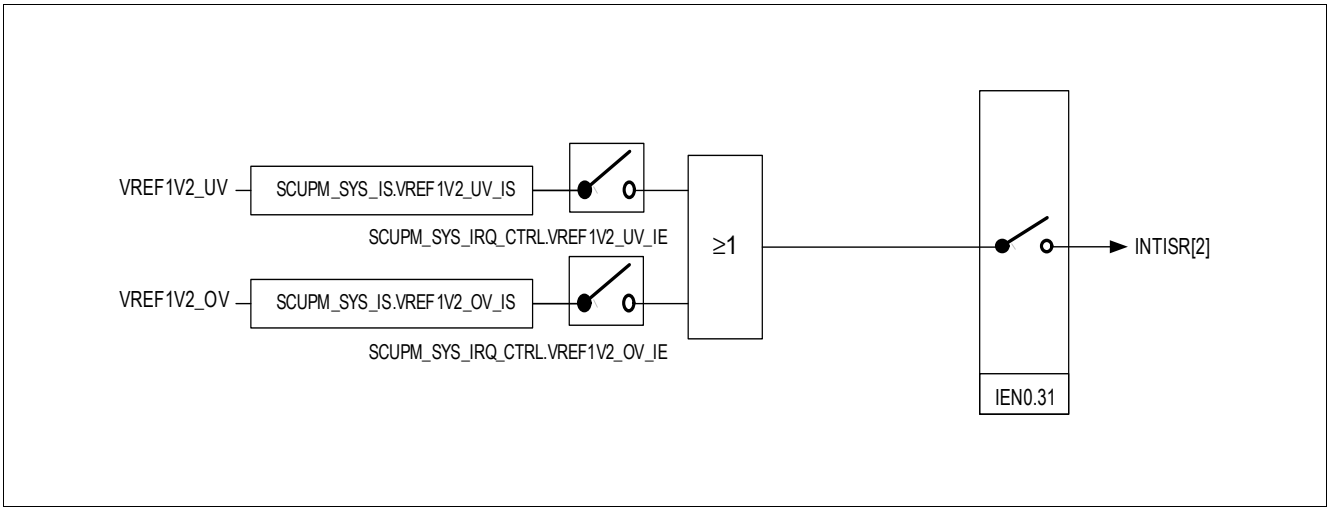


Figure 58 Interrupt Request Sources 2 (MU)

12.3.1.3 Interrupt Node 3 - ADC10

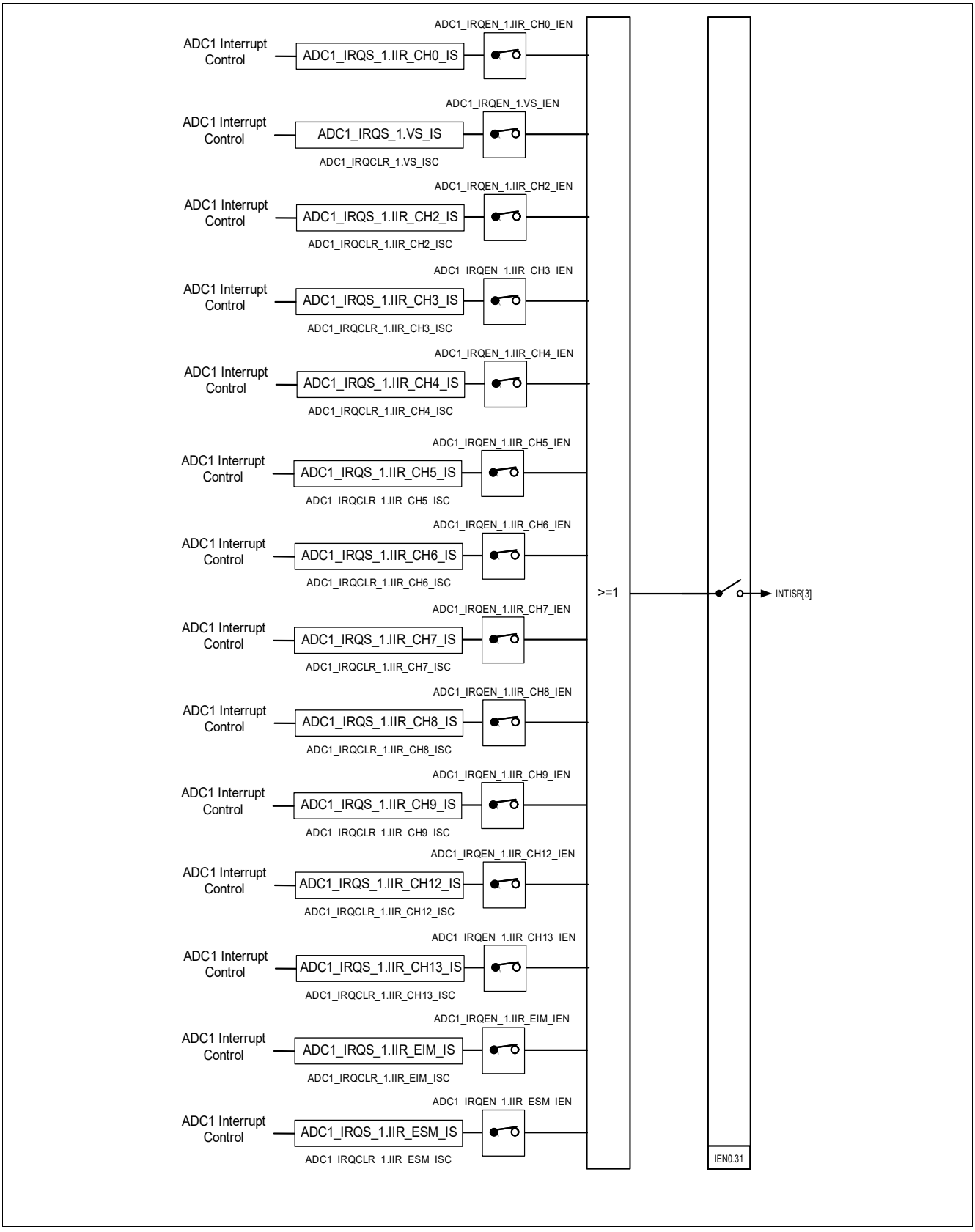


Figure 59 Interrupt Request Sources 3 (ADC10)

12.3.1.4 Interrupt Node 4, 5, 6, 7 - CCU6

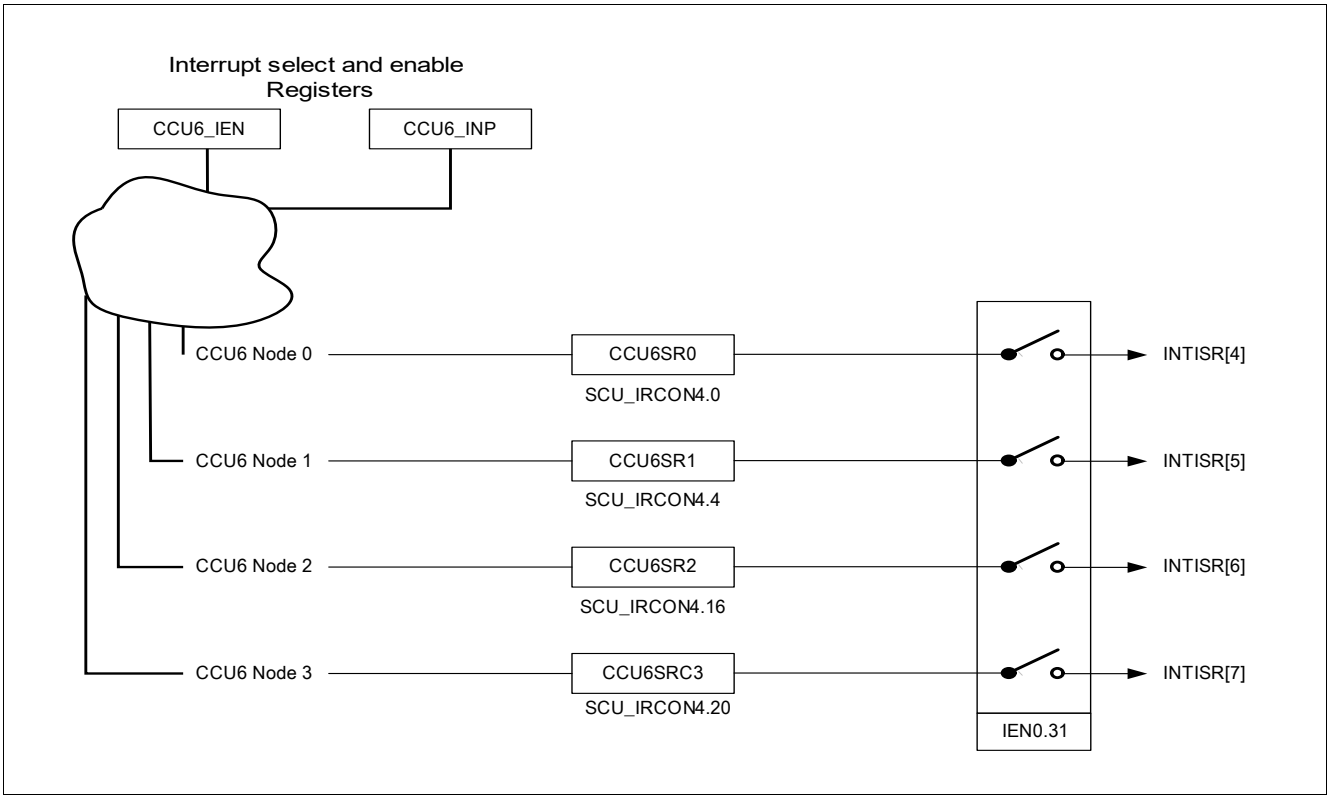


Figure 60 Interrupt Request Sources 4, 5, 6, 7 (CCU6)

12.3.1.5 Interrupt Node 8 and 9 - SSC

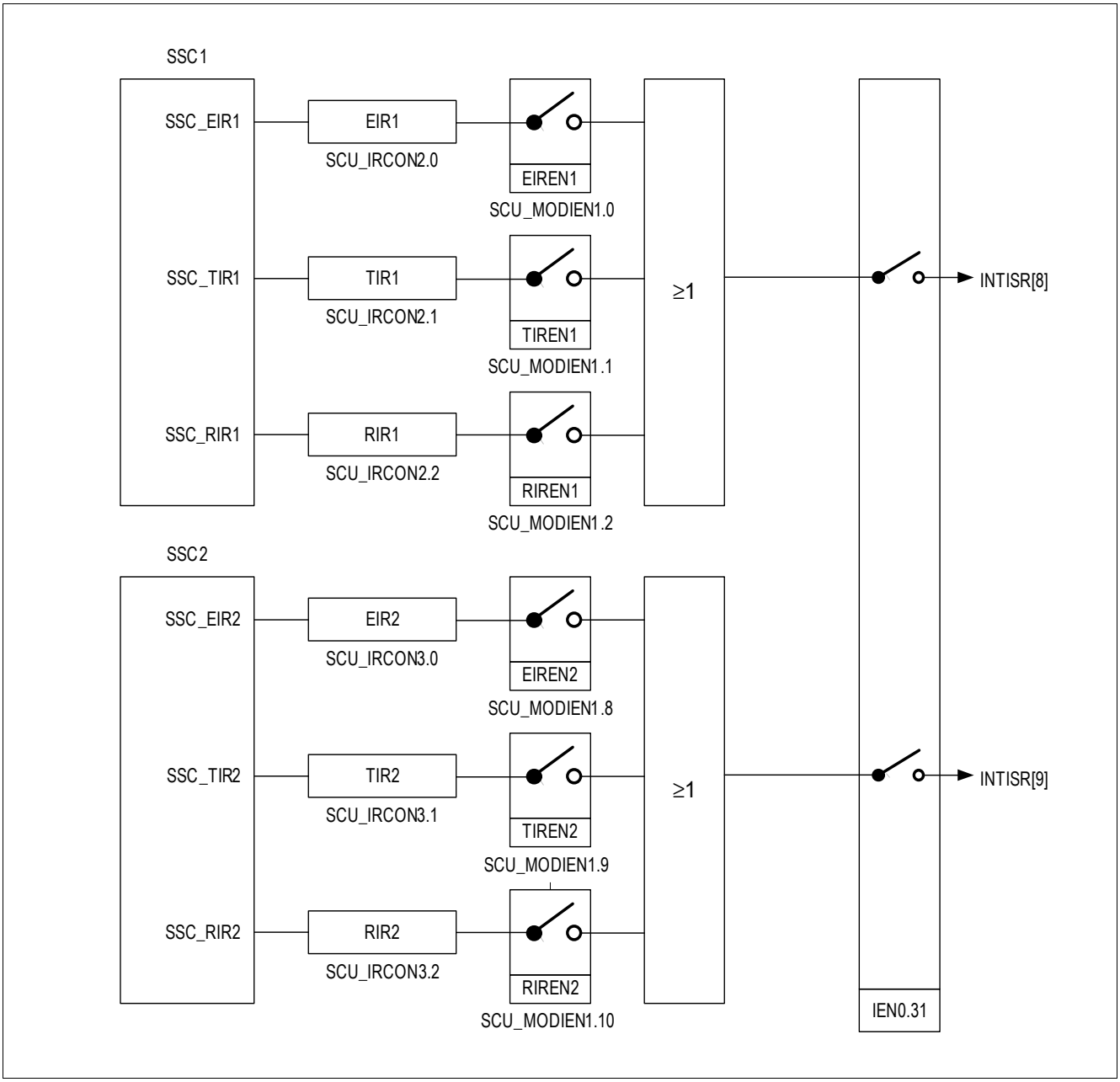


Figure 61 Interrupt Request Sources 8 and 9 (SSC)

12.3.1.6 Interrupt Node 10 - UART1

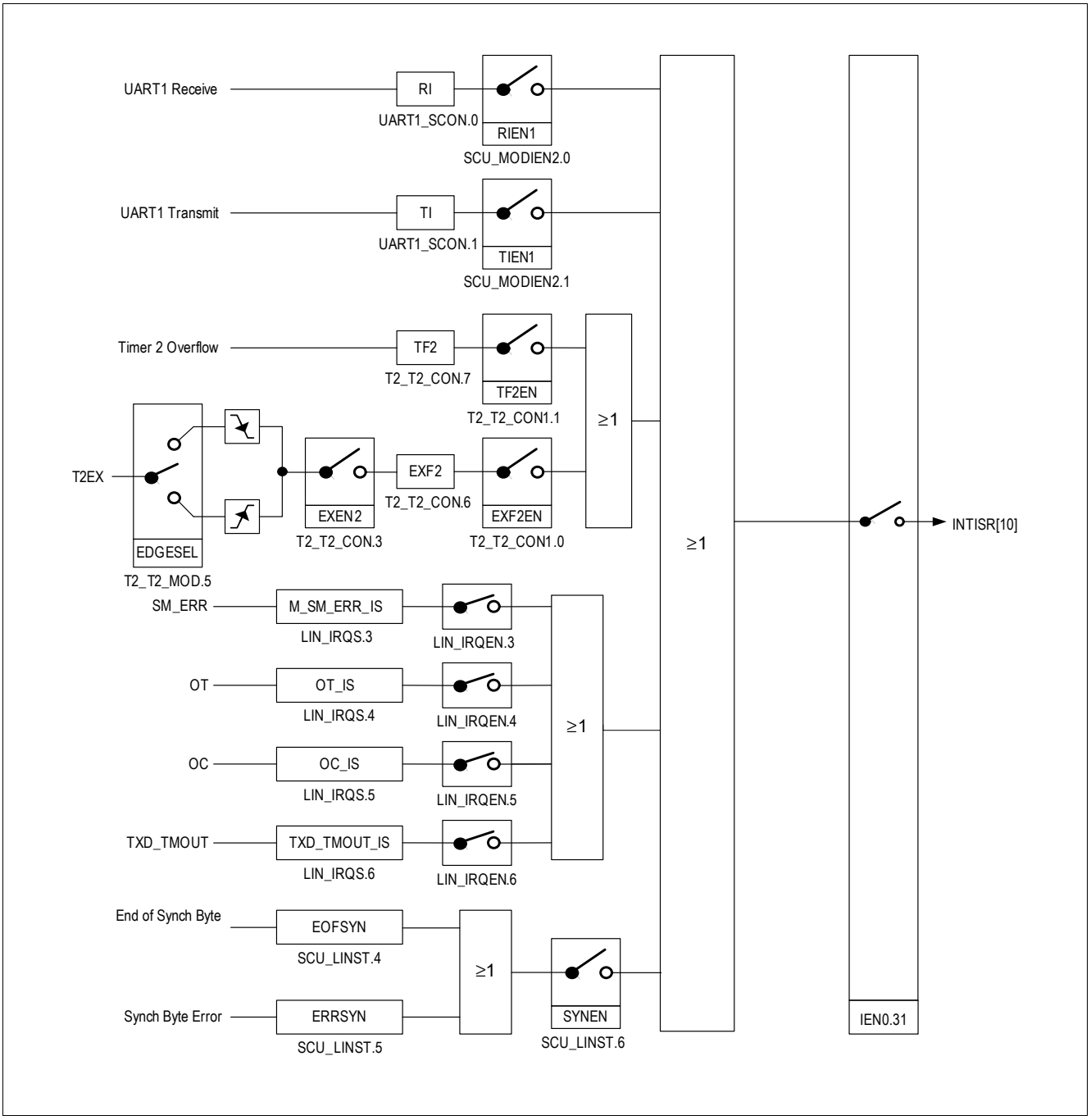


Figure 62 Interrupt Request Source 10 (UART1)

12.3.1.7 Interrupt Node 11 - UART2

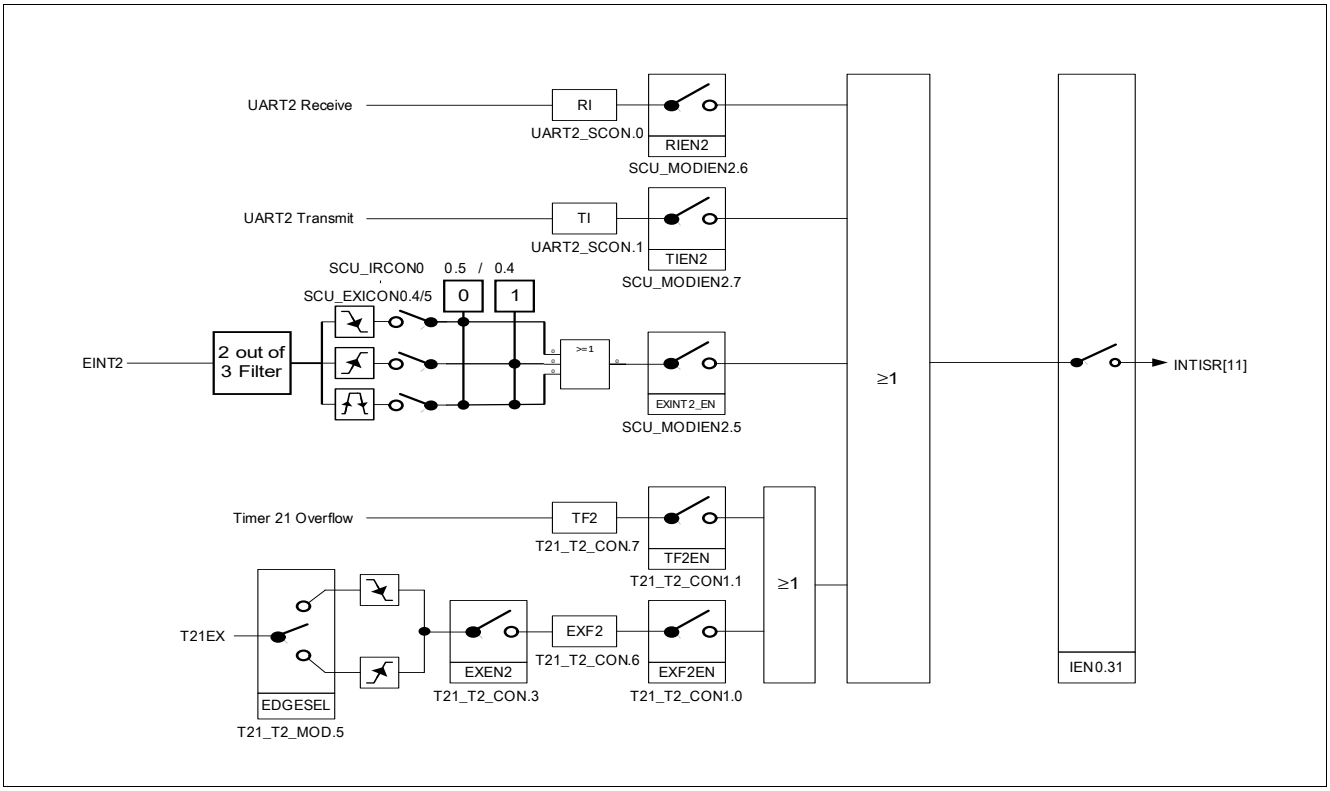


Figure 63 Interrupt Request Source 11 (UART2)

12.3.1.8 Interrupt Node 12 and 13 - Interrupt

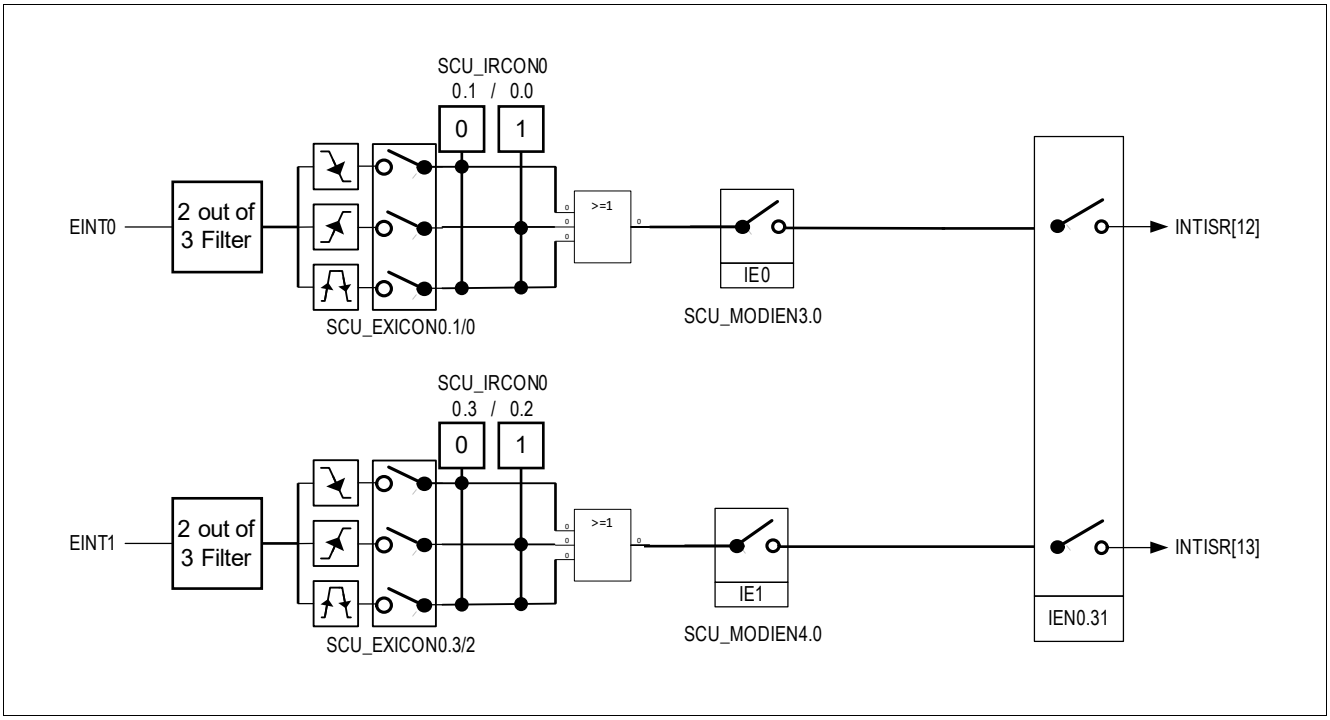


Figure 64 Interrupt Request Sources 12 and 13 (External Interrupt)

12.3.1.9 Interrupt Node 14

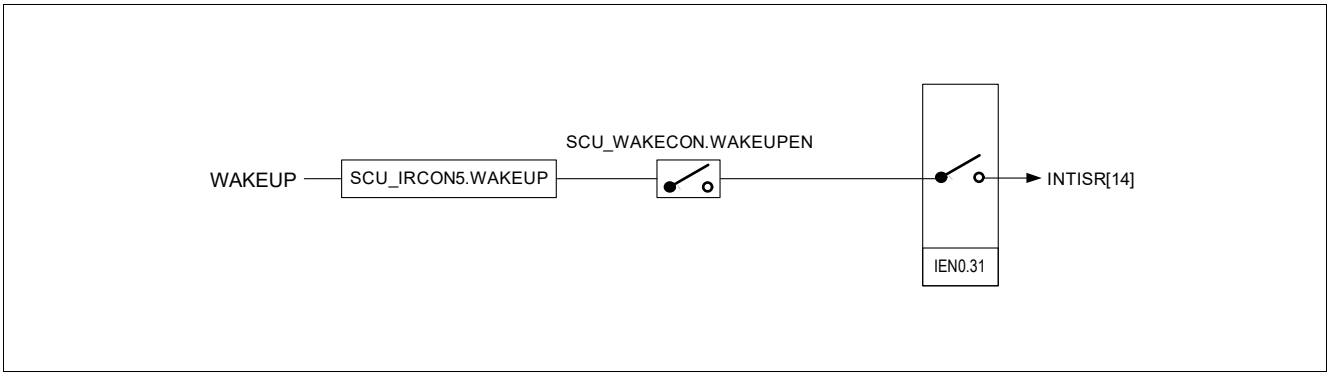


Figure 65 Interrupt Request Sources 14(Wakeup)

12.3.1.10 Interrupt Node 15

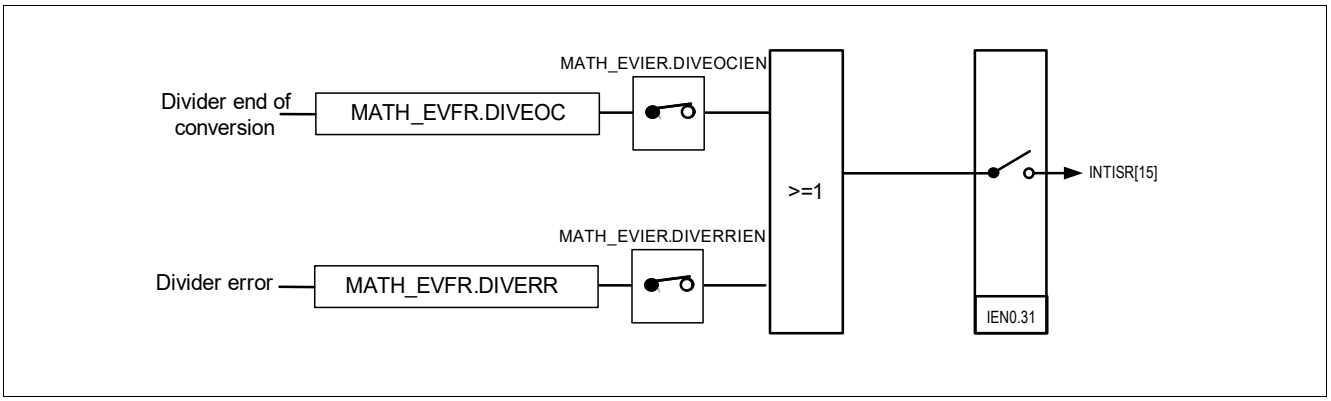


Figure 66 Interrupt Request Sources 15 (Divider Unit)

12.3.1.11 Interrupt Node 17 and 18 - Bridge Driver / Charge Pump

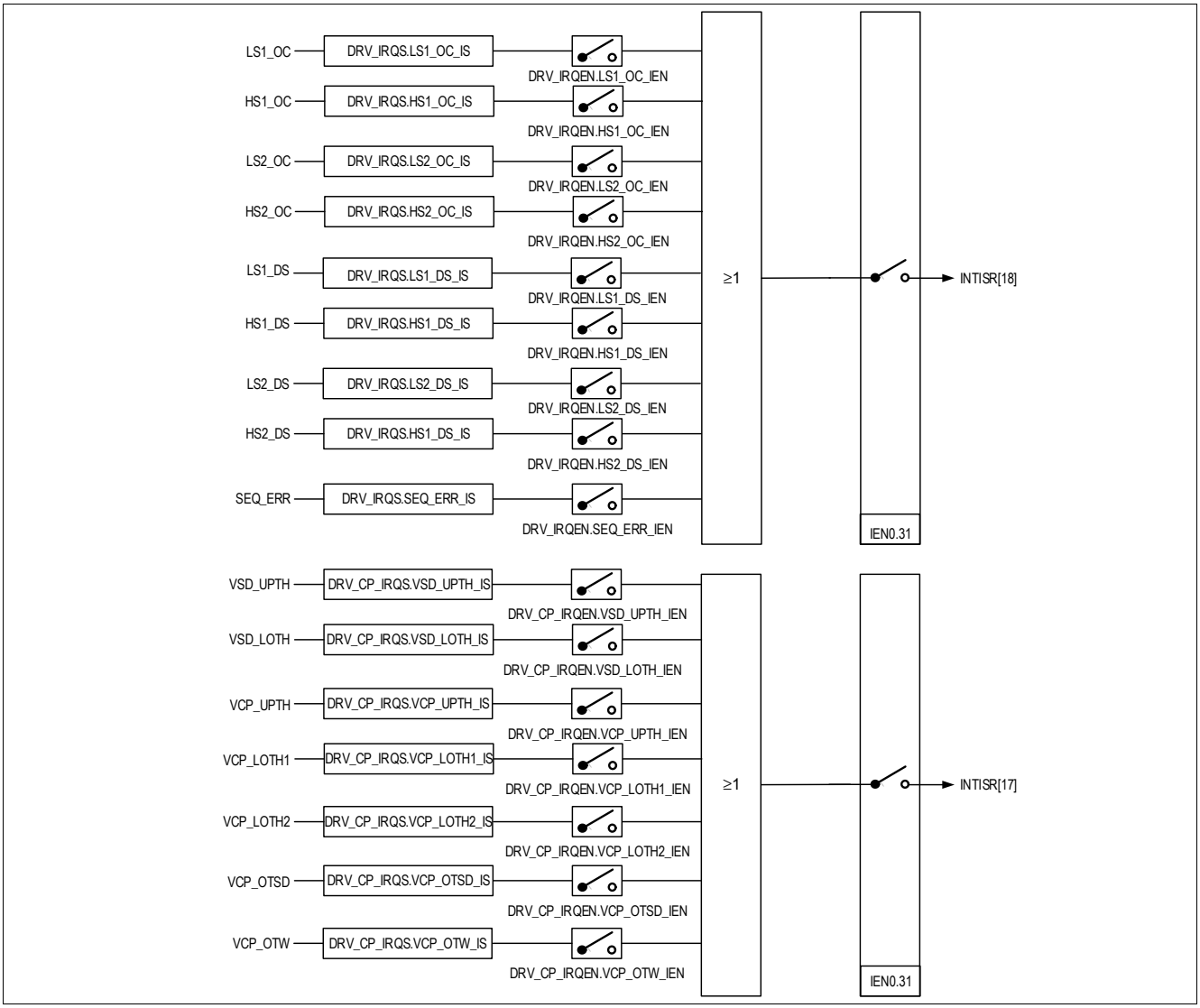


Figure 67 Interrupt Request Sources 17,18 (Charge Pump / Bridge Driver)

12.3.1.12 Interrupt Node 19 - HS

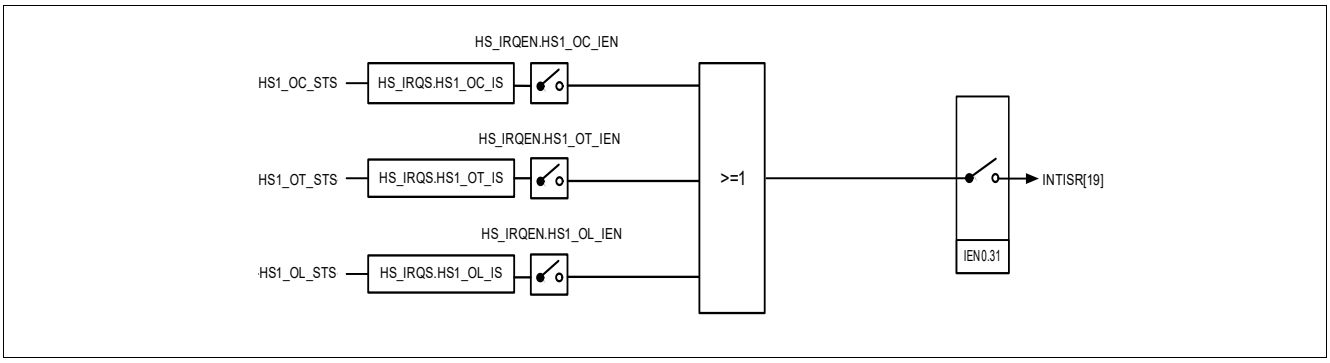


Figure 68 Interrupt Request Sources 19 (HS)

12.3.1.13 Interrupt Node 20 - Current Sense Amplifier

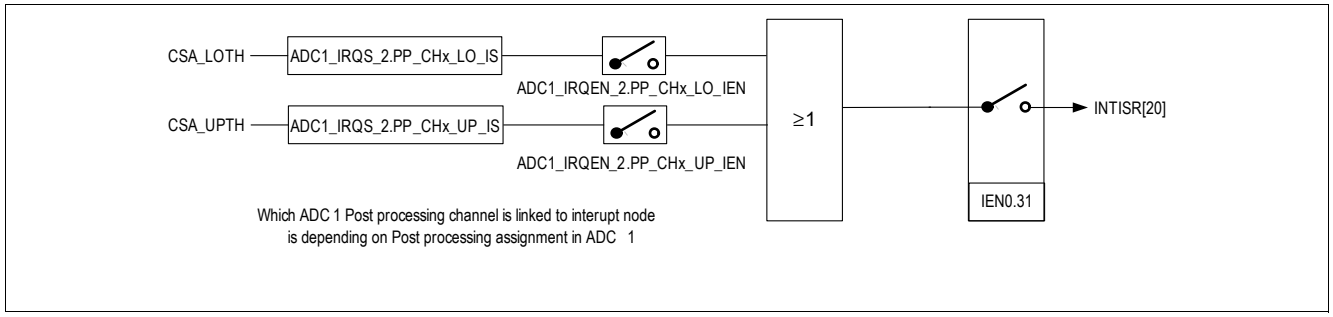


Figure 69 Interrupt Request Sources 20 (Current Sense Amplifier)

12.3.1.14 Interrupt Node 21 - DPP1 Differential Unit

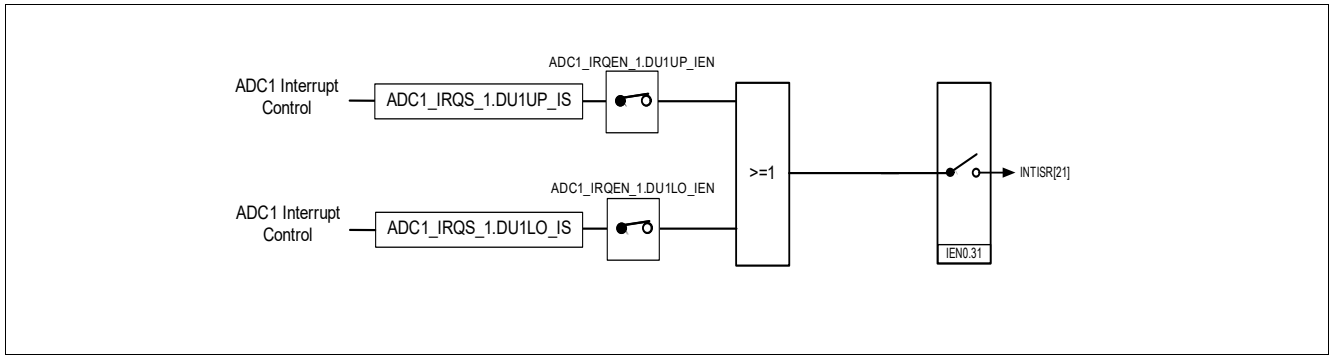


Figure 70 Interrupt Request Sources 21 (DPP1 - Diff Unit)

12.3.1.15 Interrupt Node 22 - MONx

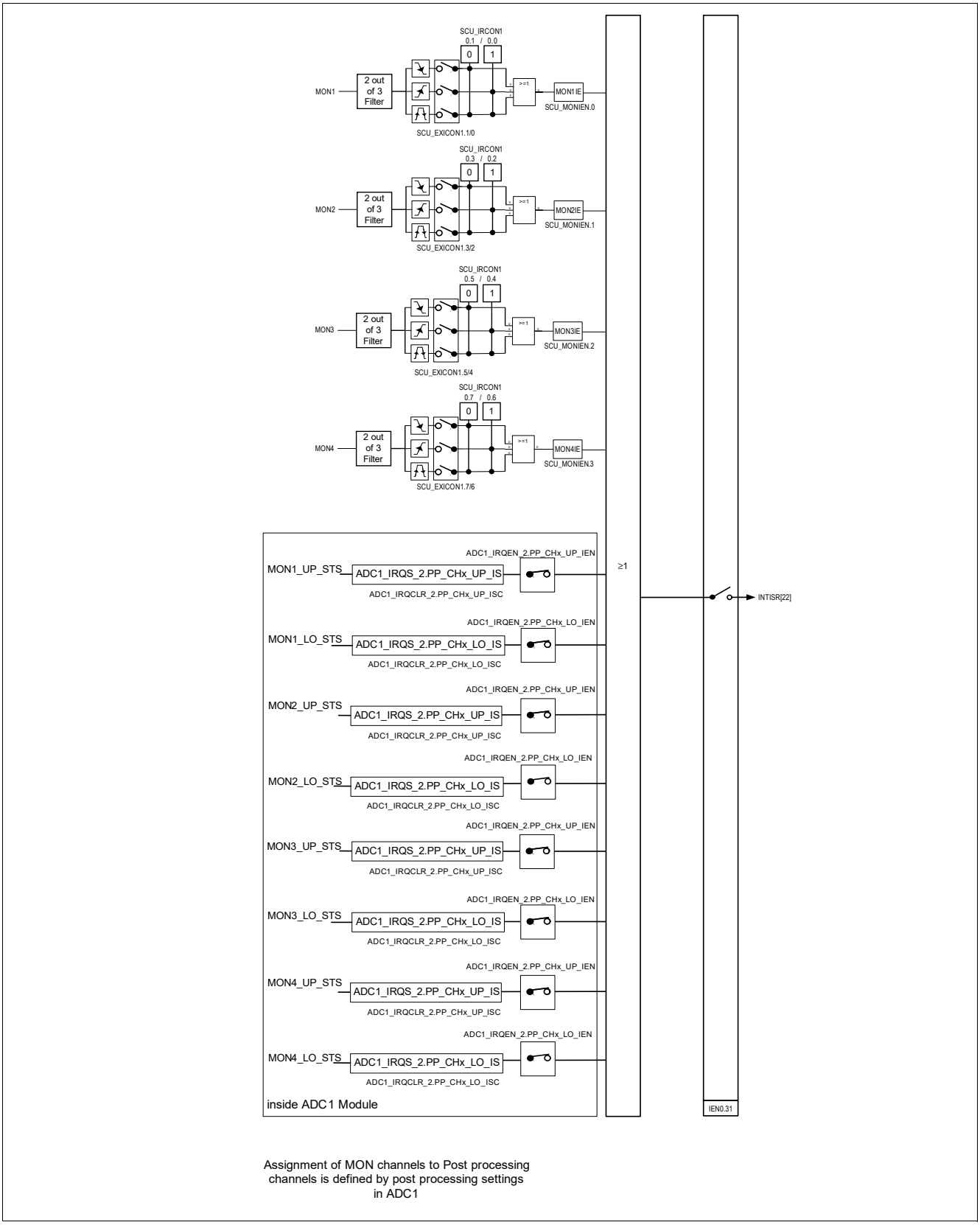


Figure 71 Interrupt Request Sources 22(MON1..4)

12.3.1.16 Interrupt Node 23 - Port2.x

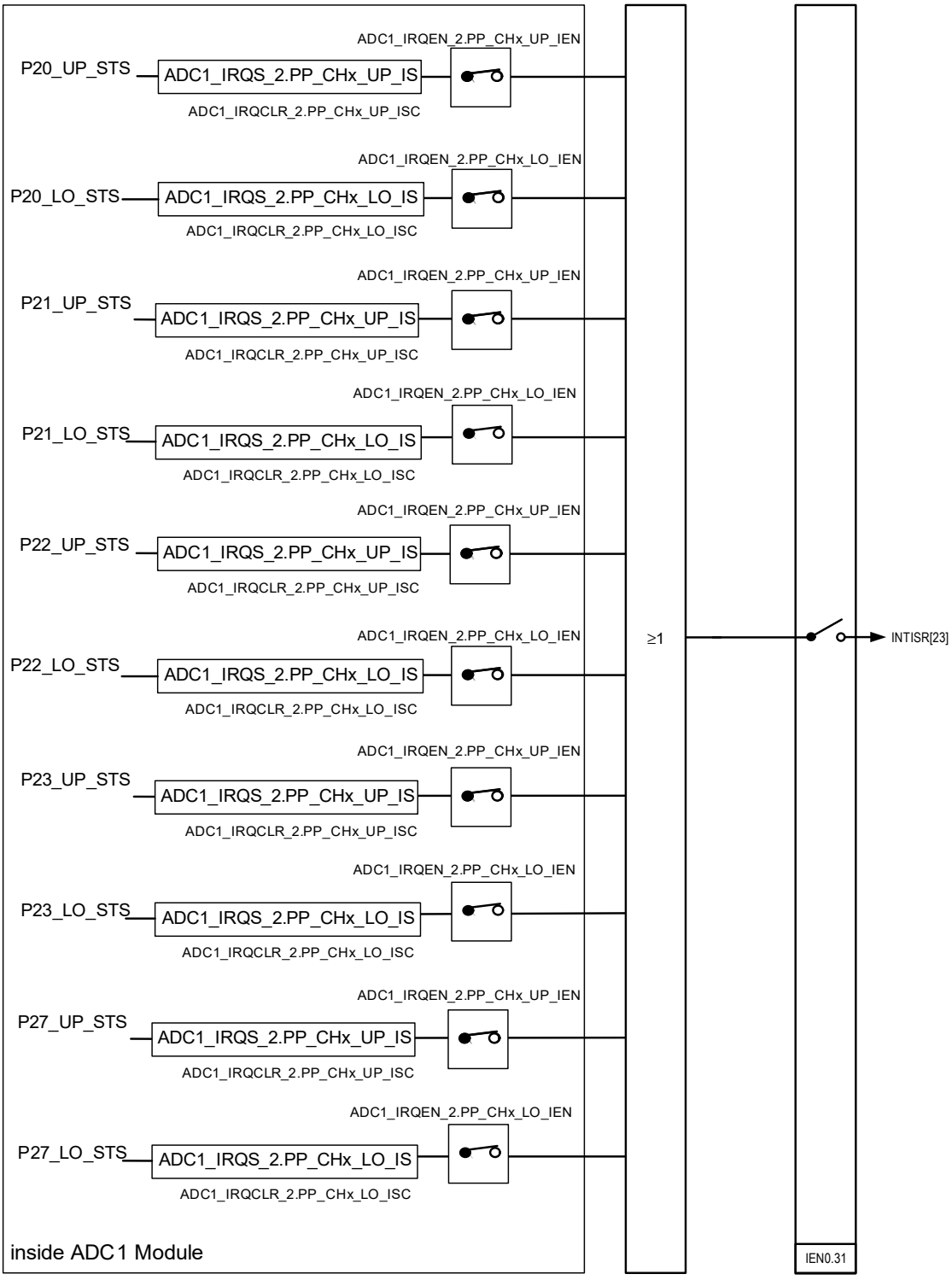


Figure 72 Interrupt Request Sources 23 (Port 2.x)

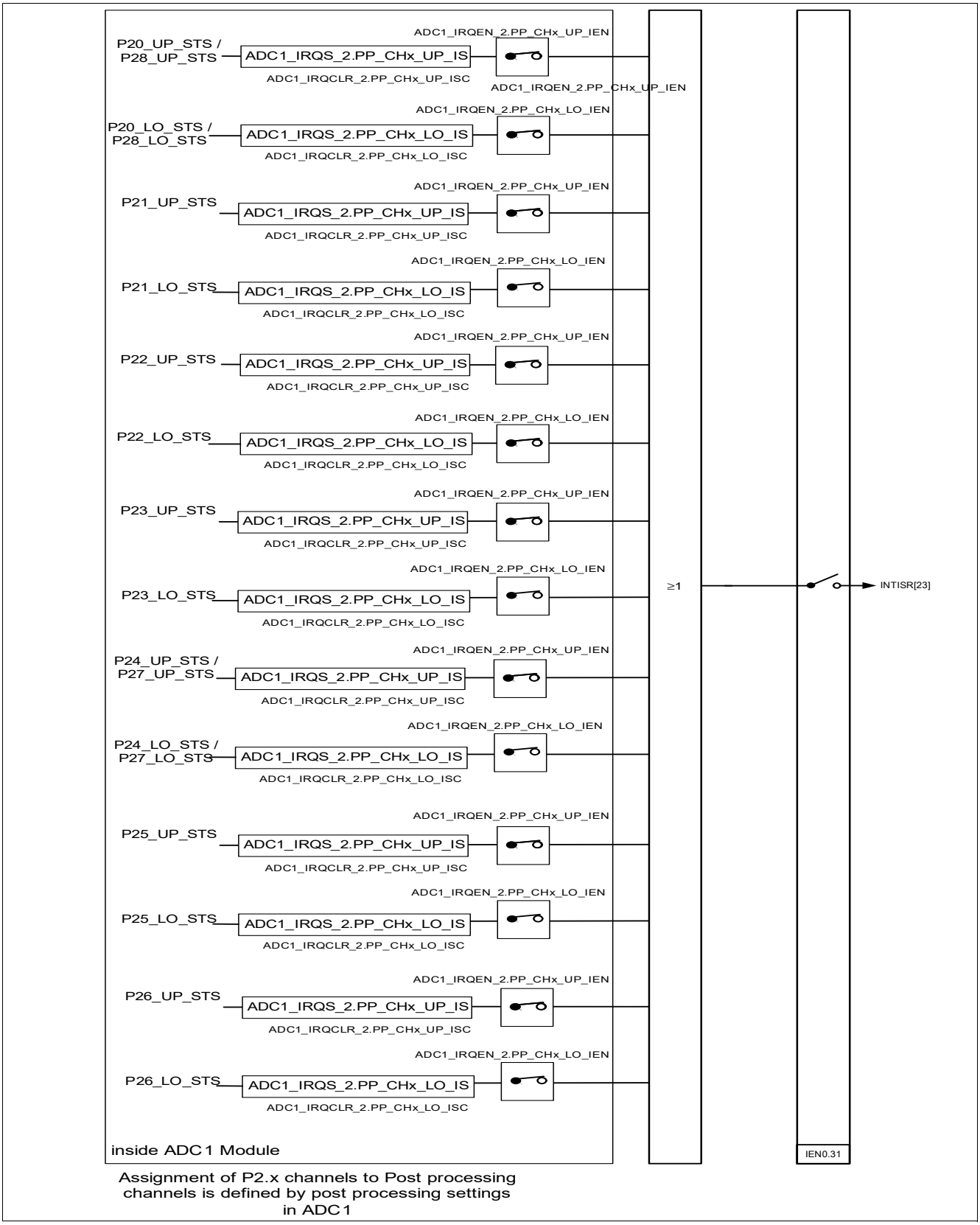


Figure 73 Interrupt Request Sources 23 (Port 2.x)

12.3.1.17 Non-Maskable Interrupt Request Source (NMI)

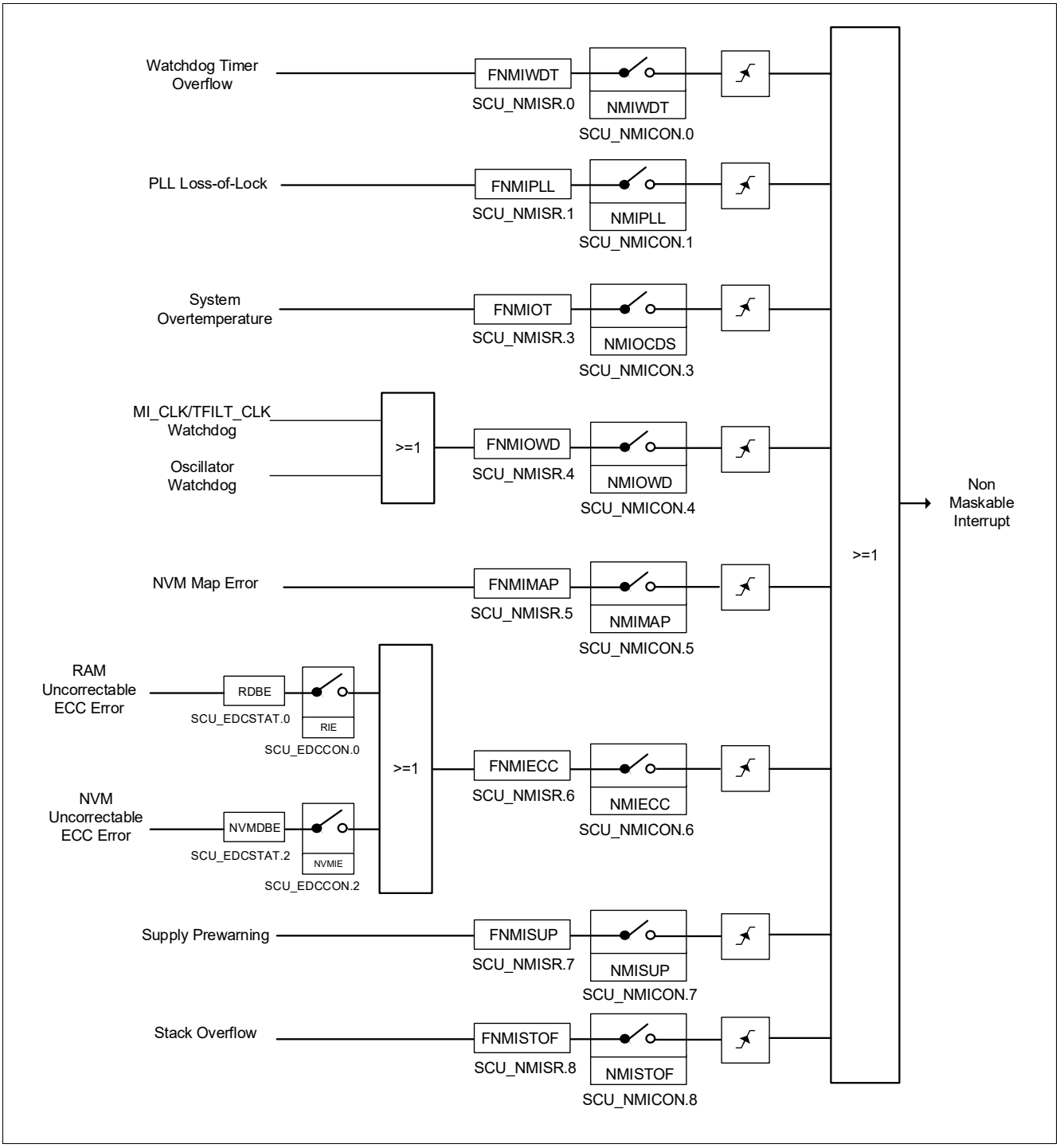


Figure 74 Non-Maskable Interrupt Request Source

12.3.1.18 Interrupt Flags Overview

Table 172 All Interrupt Flags and Enable

Service Request	Node ID	Level/Edge Sensitive	Duration	SFR Flag	Interrupt Enable
None Maskable Interrupts					
Watchdog NMI - FNMIWDT					
Watchdog Timer	NMI	edge	set until cleared by software	**SCU_WDTCON**.WDTPR	**SCU_WDTCON**.WDTEN
PLL NMI - FNMIPLL					
PLL Loss of Lock	NMI	level	set until cleared by software	**SCU_PLL_CON**.LOCK	N/A
OT NMI - FNMIOT					
SYS_OTWARN	NMI	edge	set until cleared by software	**SCUPM_SYS_IS**.SYS_OTWARN_IS	**SCUPM_SYS_IRQ_CTRL**.SYS_OTWARN_IE
SYS_OT	NMI	edge	set until cleared by software	**SCUPM_SYS_IS**.SYS_OT_IS	**SCUPM_SYS_IRQ_CTRL**.SYS_OT_IE
Clock Watchdog NMI - FNMIOWD					
CLKWDT	NMI	level	set until cleared by software	**SCU_APCLK_STS**.APCLK2STS **SCU_APCLK_STS**.APCLK1STS	**SCU_APCLK_CTRL**.CLKWDT_IE
OSCWDT	NMI	level	set until cleared by software	**SCU_OSC_CON**.OSC2L	N/A
NVM MAP NMI - FNMIMAP					
NVM Map Error	NMI	level	set until cleared by software	N/A	N/A
ECC Error NMI - FNMIECC					
RAM Uncorrectable	NMI	level	set until cleared by software	**SCU_EDCSTAT**.RDBE	**SCU_EDCCON**.RIE
NVM Uncorrectable	NMI	level	set until cleared by software	**SCU_EDCSTAT**.NVMDBE	**SCU_EDCCON**.NVMIE
Supply NMI - FNMISUP					
PREWARN_SUP VBAT_UV	NMI	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CH0_LO_IS (if PP channel is mapped to VBAT_SENSE)	**ADC1_IRQEN_2**.PP_CH0_LO_IEN (if PP channel is mapped to VBAT_SENSE)

Table 172 All Interrupt Flags and Enable (cont'd)

Service Request	Node ID	Level/Edge Sensitive	Duration	SFR Flag	Interrupt Enable
PREWARN_SUPVS_UV	NMI	edge	set until cleared by software	**SCUPM_SYS_SUPPLY_IRQ_STS**.VS_UV_IS	**SCUPM_SYS_SUPPLY_IRQ_CTRL**.VS_UV_IE
PREWARN_SUPVS_ADC10_UV	NMI	edge	set until cleared by software	**ADC1_IRQS_2**.VS_LO_IS	**ADC1_IRQEN_2**.VS_LO_IEN
PREWARN_SUPVDD5V_UV	NMI	edge	set until cleared by software	**SCUPM_SYS_SUPPLY_IRQ_STS**.VDD5V_UV_IS	**SCUPM_SYS_SUPPLY_IRQ_CTRL**.VDD5V_UV_IE
PREWARN_SUPVDD1V5_UV	NMI	edge	set until cleared by software	**SCUPM_SYS_SUPPLY_IRQ_STS**.VDD1V5_UV_IS	**SCUPM_SYS_SUPPLY_IRQ_CTRL**.VDD1V5_UV_IE
PREWARN_SUPVDDEXT_UV	NMI	edge	set until cleared by software	**SCUPM_SYS_SUPPLY_IRQ_STS**.VDDEXT_UV_IS	**SCUPM_SYS_SUPPLY_IRQ_CTRL**.VDDEXT_UV_IE
PREWARN_SUPVBAT_OV	NMI	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CH0_UP_IS (if PP channel is mapped to VBAT_SENSE)	**ADC1_IRQEN_2**.PP_CH0_UP_IEN (if PP channel is mapped to VBAT_SENSE)
PREWARN_SUPVS_OV	NMI	edge	set until cleared by software	**SCUPM_SYS_SUPPLY_IRQ_STS**.VS_OV_IS	**SCUPM_SYS_SUPPLY_IRQ_CTRL**.VS_OV_IE
PREWARN_SUPVS_ADC10_OV	NMI	edge	set until cleared by software	**ADC1_IRQS_2**.VS_UP_IS	**ADC1_IRQEN_2**.VS_UP_IEN
PREWARN_SUPVDD5V_OV	NMI	edge	set until cleared by software	**SCUPM_SYS_SUPPLY_IRQ_STS**.VDD5V_OV_IE	**SCUPM_SYS_SUPPLY_IRQ_CTRL**.VDD5V_OV_IE
PREWARN_SUPVDD1V5_OV	NMI	edge	set until cleared by software	**SCUPM_SYS_SUPPLY_IRQ_STS**.VDD1V5_OV_IS	**SCUPM_SYS_SUPPLY_IRQ_CTRL**.VDD1V5_OV_IE
PREWARN_SUPVDDEXT_OV	NMI	edge	set until cleared by software	**SCUPM_SYS_SUPPLY_IRQ_STS**.VDDEXT_OV_IS	**SCUPM_SYS_SUPPLY_IRQ_CTRL**.VDDEXT_OV_IE
PMU_OVERTEMP	NMI	level	set until cleared by software	**PMU_SUPPLY_STS**.PMU_OVERTEMP	**PMU_OT_CTRL**.PMU_OT_INT_EN
VDDP_OVERVOLT	NMI	level	set until cleared by software	**PMU_SUPPLY_STS**.PMU_5V_OVERVOLT	**PMU_SUPPLY_STS**.PMU_5V_FAIL_EN
VDDP_OVERLOAD	NMI	level	set until cleared by software	**PMU_SUPPLY_STS**.PMU_5V_OVERLOAD	**PMU_SUPPLY_STS**.PMU_5V_FAIL_EN

Table 172 All Interrupt Flags and Enable (cont'd)

Service Request	Node ID	Level/Edge Sensitive	Duration	SFR Flag	Interrupt Enable
VDDC_OVERVOLT	NMI	level	set until cleared by software	**PMU_SUPPLY_STS**.PMU_1V5_OVERVOLT	**PMU_SUPPLY_STS**.PMU_1V5_FAIL_EN
VDDC_OVERLOAD	NMI	level	set until cleared by software	**PMU_SUPPLY_STS**.PMU_1V5_OVERLOAD	**PMU_SUPPLY_STS**.PMU_1V5_FAIL_EN
VDDEXT_OVERTEMP	NMI	level	set until cleared by software	**PMU_VDDEXT_CTRL**.VDDEXT_OT_IS	**PMU_VDDEXT_CTRL**.VDDEXT_FAIL_EN
VDDEXT_UNDERVOLT	NMI	level	set until cleared by software	**PMU_VDDEXT_CTRL**.VDDEXT_UV_IS	**PMU_VDDEXT_CTRL**.VDDEXT_FAIL_EN
Stack NMI - FNMISTOF					
Stack Overflow	NMI	edge	set until cleared by software	**SCU_STACK_OVF_STS**.STOF_STS	**SCU_STACK_OVF_CTRL**.STOF_EN
INTISR<0/1> → GPT12					
GPT12-T2	0	level	2 per_clk cycles	GPT1_T2: **SCU_GPT12IRC**.GPT1T2	**SCU_GPT12IEN**.T2IE
GPT12-T3	0	level	2 per_clk cycles	GPT1_T3: **SCU_GPT12IRC**.GPT1T3	**SCU_GPT12IEN**.T3IE
GPT12-T4	0	level	2 per_clk cycles	GTP1_T4: **SCU_GPT12IRC**.GPT1T4	**SCU_GPT12IEN**.T4IE
GPT12-T5	1	level	2 per_clk cycles	GPT2_T5: **SCU_GPT12IRC**.GPT2T5	**SCU_GPT12IEN**.T5IE
GPT12-T6	1	level	2 per_clk cycles	GPT2_T6: **SCU_GPT12IRC**.GPT2T6	**SCU_GPT12IEN**.T6IE
GPT12-CR	1	level	2 per_clk cycles	GPT2_CR: **SCU_GPT12IRC**.GPT2CR	**SCU_GPT12IEN**.CRIE
INTISR<2> → MU					
VREF1V2_UV	2	level	set until cleared by software	VREF1V2_UV: **SCUPM_SYS_IS**.VREF1V2_UV_IS	**SCUPM_SYS_IRQ_CTRL**.VREF1V2_UV_IE
VREF1V2_OV	2	level	set until cleared by software	VREF1V2_OV: **SCUPM_SYS_IS**.VREF1V2_OV_IS	**SCUPM_SYS_IRQ_CTRL**.VREF1V2_OV_IE
INTISR<3> → ADC 10 Bit					
ADC10CH0	3	edge	set until cleared by software	ADC1_Interrupt_Control: **ADC1_IRQS_1**.IIR_CH0_IS	**ADC1_IRQEN_1**.IIR_CH0_IEN

Table 172 All Interrupt Flags and Enable (cont'd)

Service Request	Node ID	Level/Edge Sensitive	Duration	SFR Flag	Interrupt Enable
ADC10CH1	3	edge	set until cleared by software	ADC1_Interrupt_Control: **ADC1_IRQS_1**.IIR_VS_IS	**ADC1_IRQEN_1**.IIR_VS_IEN
ADC10-CH2	3	edge	set until cleared by software	ADC1_Interrupt_Control: **ADC1_IRQS_1**.IIR_CH2_IS	**ADC1_IRQEN_1**.IIR_CH2_IEN
ADC10-CH3	3	edge	set until cleared by software	ADC1_Interrupt_Control: **ADC1_IRQS_1**.IIR_CH3_IS	**ADC1_IRQEN_1**.IIR_CH3_IEN
ADC10-CH4	3	edge	set until cleared by software	ADC1_Interrupt_Control: **ADC1_IRQS_1**.IIR_CH4_IS	**ADC1_IRQEN_1**.IIR_CH4_IEN
ADC10-CH5	3	edge	set until cleared by software	ADC1_Interrupt_Control: **ADC1_IRQS_1**.IIR_CH5_IS	**ADC1_IRQEN_1**.IIR_CH5_IEN
ADC10-CH6	3	edge	set until cleared by software	ADC1_Interrupt_Control: **ADC1_IRQS_1**.IIR_CH6_IS	**ADC1_IRQEN_1**.IIR_CH6_IEN
ADC10-CH7	3	edge	set until cleared by software	ADC1_Interrupt_Control: **ADC1_IRQS_1**.IIR_CH7_IS	**ADC1_IRQEN_1**.IIR_CH7_IEN
ADC10-CH8	3	edge	set until cleared by software	ADC1_Interrupt_Control: **ADC1_IRQS_1**.IIR_CH8_IS	**ADC1_IRQEN_1**.IIR_CH8_IEN
ADC10-CH9	3	edge	set until cleared by software	ADC1_Interrupt_Control: **ADC1_IRQS_1**.IIR_CH9_IS	**ADC1_IRQEN_1**.IIR_CH9_IEN
ADC10-CH10	3	edge	set until cleared by software	ADC1_Interrupt_Control: **ADC1_IRQS_1**.IIR_CH10_IS	**ADC1_IRQEN_1**.IIR_CH10_IEN
ADC10-CH11	3	edge	set until cleared by software	ADC1_Interrupt_Control: **ADC1_IRQS_1**.IIR_CH11_IS	**ADC1_IRQEN_1**.IIR_CH11_IEN
ADC10-CH12	3	edge	set until cleared by software	ADC1_Interrupt_Control: **ADC1_IRQS_1**.IIR_CH12_IS	**ADC1_IRQEN_1**.IIR_CH12_IEN
ADC10-CH13	3	edge	set until cleared by software	ADC1_Interrupt_Control: **ADC1_IRQS_1**.IIR_CH13_IS	**ADC1_IRQEN_1**.IIR_CH13_IEN
ADC10-ESM	3	edge	set until cleared by software	ADC1_Interrupt_Control: **ADC1_IRQS_1**.ESM_IS	**ADC1_IRQEN_1**.IIR_ESM_IEN

Table 172 All Interrupt Flags and Enable (cont'd)

Service Request	Node ID	Level/Edge Sensitive	Duration	SFR Flag	Interrupt Enable
ADC10-EIM	3	edge	set until cleared by software	ADC1_Interrupt_Control: **ADC1_IRQS_1**.IIR_EIM_IS	**ADC1_IRQEN_1**.IIR_EIM_IEN
INTISR<4,5,6,7> → CCU6					
CCU0[1)]	4	level	2 per_clk cycles	CCU6 Node 0: **SCU_IRCON4**.CCU6SR0	**CCU6_IEN**/**CCU6_INP**
CCU1[1)]	5	level	2 per_clk cycles	CCU6 Node 1: **SCU_IRCON4**.CCU6SR1	**CCU6_IEN**/**CCU6_INP**
CCU2[1)]	6	level	2 per_clk cycles	CCU6 Node 2: **SCU_IRCON4**.CCU6SR2	**CCU6_IEN**/**CCU6_INP**
CCU3[1)]	7	level	2 per_clk cycles	CCU6 Node 3: **SCU_IRCON4**.CCU6SR3	**CCU6_IEN**/**CCU6_INP**
INTISR<8,9> → SSC1/SSC2					
SSC1	8	level	2 per_clk cycles	SSC1.SSC_EIR1: **SCU_IRCON2**.EIR1	**SCU_MODIEN1**.EIREN1
SSC1	8	level	2 per_clk cycles	SSC1.SSC_TIR1: **SCU_IRCON2**.TIR1	**SCU_MODIEN1**.TIREN1
SSC1	8	level	2 per_clk cycles	SSC1.SSC_RIR1: **SCU_IRCON2**.RIR1	**SCU_MODIEN1**.RIREN1
SSC2	9	level	2 per_clk cycles	SSC2.SSC_EIR1: **SCU_IRCON3**.EIR2	**SCU_MODIEN1**.EIREN2
SSC2	9	level	2 per_clk cycles	SSC2.SSC_TIR1: **SCU_IRCON3**.TIR2	**SCU_MODIEN1**.TIREN2
SSC2	9	level	2 per_clk cycles	SSC2.SSC_RIR1: **SCU_IRCON3**.RIR2	**SCU_MODIEN1**.RIREN2
INTISR<10,11> → UART1/UART2					
UART1 Receive	10	level	copy of RI bit, set until cleared by software	UART1: **UART_SCON**.RI	**SCU_MODIEN2**.RIEN1
UART1 Transmit	10	level	copy of TI bit, set until cleared by software	UART1: **UART_SCON**.TI	**SCU_MODIEN2**.TIEN1
LIN sync byte error	10	level	set until cleared by software	Synch Byte Error: **SCU_LINST**.ERRSYN	**SCU_LINST**.SYNEN
LIN end of sync byte	10	level	set until cleared by software	End of Synch Byte: **SCU_LINST**.EOFSYN	**SCU_LINST**.SYNEN

Table 172 All Interrupt Flags and Enable (cont'd)

Service Request	Node ID	Level/Edge Sensitive	Duration	SFR Flag	Interrupt Enable
Timer 2	10	edge	set until cleared by software	Timer2 Overflow: **T2_CON**.TF2	**T2_CON1**.TF2EN
T2EX	10	edge	set until cleared by software	T2EX: **T2_CON**.EXF2	**T2_CON1**.EXF2EN
LIN OT	10	edge	set until cleared by software	LIN: **LIN_IRQS**.OT_IS **SCUPM_SYS_IS**.LIN_FAIL_IS	**LIN_IRQEN**.OT_IEN
LIN OC	10	level	set until cleared by software	LIN: **LIN_IRQS**.OC_IS **SCUPM_SYS_IS**.LIN_FAIL_IS	**LIN_IRQEN**.OC_IEN
TXD_TMOUT	10	edge	set until cleared by software	LIN: **LIN_IRQS**.TXD_TMOUT_IS	**LIN_IRQEN**.TXD_TMOUT_IEN
M_SM_ERR	10	edge	set until cleared by software	LIN: **LIN_IRQS**.M_SM_ERR_IS	**LIN_IRQEN**.M_SM_ERR_IEN
T21EX	10	edge	set until cleared by software	T21EX: **T2_CON**.EXF2	**T2_CON1**.EXF2EN
UART2 Receive	11	level	copy of RI bit, set until cleared by software	UART2: **UART_SCON**.RI	**SCU_MODIEN2**.RIEN2
UART2 Transmit	11	level	copy of TI bit, set until cleared by software	UART2: **UART_SCON**.TI	**SCU_MODIEN2**.TIEN2
exint2	11	edge	set until cleared by software	EINT2: **SCU_IRCON0**.EXINT2R/F	**SCU_MODIEN2**.EXINT2_EN
Timer 21	11	edge	set until cleared by software	Timer21 Overflow: **T2_CON**.TF2	**T2_CON1**.TF2EN
T21EX	11	edge	set until cleared by software	T21EX: **T2_CON**.EXF2	**T2_CON1**.EXF2EN
INTISR<12,13> → EXTINT0/EXTINT1					
exint0	12	edge		EINT0: **SCU_IRCON0**.EXINT0R/F	**SCU_MODIEN3**.IE0

Table 172 All Interrupt Flags and Enable (cont'd)

Service Request	Node ID	Level/Edge Sensitive	Duration	SFR Flag	Interrupt Enable
exint1	13	edge		EINT1: **SCU_IRCON0**.EXINT1R/F	**SCU_MODIEN4**.IE1
INTISR<14> → Wakeup					
wakeup	14	edge		Wake: **SCU_IRCON5**.WAKEUP	**SCU_WAKECON**.WAKEUPEN
INTISR<15> → Math Divider					
DIVERR	15_0			Math Div: **MATH_EVFR**.DIVERR	**MATH_EVIER**.DIVERRIEN
DIVEOC	15_1			Math Div: **MATH_EVFR**.DIVEOC	**MATH_EVIER**.DIVEOCIEN
INTISR<17> → Charge Pump					
VSD_UPTH	17_0	edge	set until cleared by software	**BDRV_CP_IRQS**.VSD_UPTH_IS	**BDRV_CP_IRQEN**.VSD_UPTH_IEN
VSD_LOTH	17_1	edge	set until cleared by software	**BDRV_CP_IRQS**.VSD_LOTH_IS	**BDRV_CP_IRQEN**.VSD_LOTH_IEN
VCP_UPTH	17_2	edge	set until cleared by software	**BDRV_CP_IRQS**.VCP_UPTH_IS	**BDRV_CP_IRQEN**.VCP_UPTH_IEN
VCP_LOTH1	17_3	edge	set until cleared by software	**BDRV_CP_IRQS**.VCP_LOTH1_IS	**BDRV_CP_IRQEN**.VCP_LOTH1_IEN
VCP_LOTH2	17_4	edge	set until cleared by software	**BDRV_CP_IRQS**.VCP_LOTH2_IS	**BDRV_CP_IRQEN**.VCP_LOTH2_IEN
VCP_OTSD	17_5	edge	set until cleared by software	**BDRV_CP_IRQS**.VCP_OTSD_IS	**BDRV_CP_IRQEN**.VCP_OTSD_IEN
VCP_OTW	17_6	edge	set until cleared by software	**BDRV_CP_IRQS**.VCP_OTW_IS	**BDRV_CP_IRQEN**.VCP_OTW_IEN
INTISR<18> → Bridge Driver					
HS1_OC	18_0	level	set until cleared by software	**BDRV_IRQS**.HS1_OC_IS	**BDRV_IRQEN**.HS1_OC_IEN
LS1_OC	18_1	level	set until cleared by software	**BDRV_IRQS**.LS1_OC_IS	**BDRV_IRQEN**.LS1_OC_IEN
HS2_OC	18_2	level	set until cleared by software	**BDRV_IRQS**.HS2_OC_IS	**BDRV_IRQEN**.HS2_OC_IEN

Table 172 All Interrupt Flags and Enable (cont'd)

Service Request	Node ID	Level/Edge Sensitive	Duration	SFR Flag	Interrupt Enable
LS2_OC	18_3	level	set until cleared by software	**BDRV_IRQS**.LS2_OC_IS	**BDRV_IRQEN**.LS2_OC_IEN
HS1_DS	18_4	edge	set until cleared by software	**BDRV_IRQS**.HS1_DS_IS	**BDRV_IRQEN**.HS1_DS_IEN
LS1_DS	18_5	edge	set until cleared by software	**BDRV_IRQS**.LS1_DS_IS	**BDRV_IRQEN**.LS1_DS_IEN
HS2_DS	18_6	edge	set until cleared by software	**BDRV_IRQS**.HS2_DS_IS	**BDRV_IRQEN**.HS2_DS_IEN
LS2_DS	18_7	edge	set until cleared by software	**BDRV_IRQS**.LS2_DS_IS	**BDRV_IRQEN**.LS2_DS_IEN
SEQ_ERR	18_8	level	set until cleared by software	**BDRV_IRQS**.SEQ_ERR_IS	**BDRV_IRQEN**.SEQ_ERR_IEN
INTISR<19> → HS					
HS_OC	19_0	level	set until cleared by software	**HS_IRQS**.HS1_OC_IS	**HS_IRQEN**.HS1_OC_IEN
HS_OT	19_1	edge	set until cleared by software	**HS_IRQS**.HS1_OT_IS	**HS_IRQEN**.HS1_OT_IEN
HS_OL	19_2	edge	set until cleared by software	**HS_IRQS**.HS1_OL_IS	**HS_IRQEN**.HS1_OL_IEN
INTISR<20> → CSA					
CSA_UPTH	20_0	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_UP_IS	**ADC1_IRQEN_2**.PP_CHx_UP_IEN
CSA_LOTH	20_1	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_LO_IS	**ADC1_IRQEN_2**.PP_CHx_LO_IEN
INTISR<21> → Differential Unit					
DU1_UP	21_0	edge	set until cleared by software	**ADC1_IRQS_1**.DU1UP_IS	**ADC1_IRQEN_1**.DU1UP_IEN
DU1_LO	21_1	edge	set until cleared by software	**ADC1_IRQS_1**.DU1LO_IS	**ADC1_IRQEN_1**.DU1LO_IEN

Table 172 All Interrupt Flags and Enable (cont'd)

Service Request	Node ID	Level/Edge Sensitive	Duration	SFR Flag	Interrupt Enable
INTISR<22> → MON					
WAKEUP	22_0	edge	set until cleared by software	**SCU_IRCON1**.MON1R/F	**SCU_MONIEN**.MON1IE
WAKEUP	22_1	edge	set until cleared by software	**SCU_IRCON1**.MON2R/F	**SCU_MONIEN**.MON2IE
WAKEUP	22_2	edge	set until cleared by software	**SCU_IRCON1**.MON3R/F	**SCU_MONIEN**.MON3IE
WAKEUP	22_3	edge	set until cleared by software	**SCU_IRCON1**.MON4R/F	**SCU_MONIEN**.MON4IE
MON1_UPTH	22_5	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_UP_IS	**ADC1_IRQEN_2**.PP_CHx_UP_IEN
MON1_LOTH	22_6	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_LO_IS	**ADC1_IRQEN_2**.PP_CHx_LO_IEN
MON2_UPTH	22_7	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_UP_IS	**ADC1_IRQEN_2**.PP_CHx_UP_IEN
MON2_LOTH	22_8	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_LO_IS	**ADC1_IRQEN_2**.PP_CHx_LO_IEN
MON3_UPTH	22_9	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_UP_IS	**ADC1_IRQEN_2**.PP_CHx_UP_IEN
MON3_LOTH	22_10	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_LO_IS	**ADC1_IRQEN_2**.PP_CHx_LO_IEN
MON4_UPTH	22_11	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_UP_IS	**ADC1_IRQEN_2**.PP_CHx_UP_IEN
MON4_LOTH	22_12	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_LO_IS	**ADC1_IRQEN_2**.PP_CHx_LO_IEN
INTISR<23> → P2.x					
P20_UPTH	23_0	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_UP_IS	**ADC1_IRQEN_2**.PP_CHx_UP_IEN

Table 172 All Interrupt Flags and Enable (cont'd)

Service Request	Node ID	Level/Edge Sensitive	Duration	SFR Flag	Interrupt Enable
P20_LOTH	23_1	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_LO_IS	**ADC1_IRQEN_2**.PP_CHx_LO_IEN
P21_UPTH	23_2	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_UP_IS	**ADC1_IRQEN_2**.PP_CHx_UP_IEN
P21_LOTH	23_3	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_LO_IS	**ADC1_IRQEN_2**.PP_CHx_LO_IEN
P22_UPTH	23_4	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_UP_IS	**ADC1_IRQEN_2**.PP_CHx_UP_IEN
P22_LOTH	23_5	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_LO_IS	**ADC1_IRQEN_2**.PP_CHx_LO_IEN
P23_UPTH	23_6	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_UP_IS	**ADC1_IRQEN_2**.PP_CHx_UP_IEN
P23_LOTH	23_7	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_LO_IS	**ADC1_IRQEN_2**.PP_CHx_LO_IEN
P27_UPTH	23_14	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_UP_IS	**ADC1_IRQEN_2**.PP_CHx_UP_IEN
P27_LOTH	23_15	edge	set until cleared by software	**ADC1_IRQS_2**.PP_CHx_LO_IS	**ADC1_IRQEN_2**.PP_CHx_LO_IEN

1) Each CCU6 interrupt can be assigned to any of the CCU6 interrupt nodes [3:0] via CCU6 registers INPL/INPH.

12.4 Interrupt Structure

An interrupt event source may be generated from the on-chip peripherals or from external. Detection of interrupt events is controlled by the respective on-chip peripherals. Interrupt status flags are available for determining which interrupt event has occurred, especially useful for an interrupt node which is shared by several event sources. Each interrupt node (except NMI) has a global enable/disable bit. In most cases, additional enable bits are provided for enabling/disabling particular interrupt events (provided for NMI events). No interrupt will be requested for any occurred event that has its interrupt enable bit disabled.

There is an interrupt masking bit EA available, which is used to globally enable or disable all interrupt requests (except NMI) to the core. Resetting bit EA to 0 only masks the pending interrupt requests from the core, but does not block the capture of incoming interrupt requests.

As displayed in **Figure 75**, the interrupt event will set the interrupt status flag which doubles as a pending interrupt request to the core. An active pending interrupt request will interrupt the core only if its corresponding interrupt node is enabled. Once an interrupt node is serviced (interrupt acknowledged), its pending interrupt request (represented by the interrupt status flag) may be automatically cleared by hardware (the core).

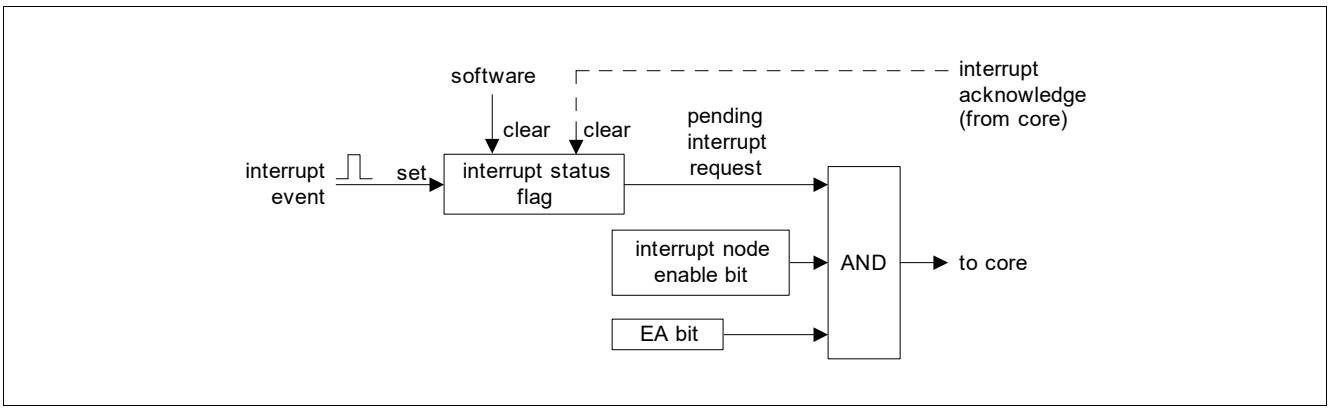


Figure 75 Interrupt Structure

For the TLE985xQX, interrupt sources like ADC10B, MU and Bridge Driver (each have a dedicated interrupt node) will have their respective interrupt status flags in the dedicated registers. This flags are not cleared by the core once their corresponding pending interrupt request is serviced. They have to cleared by software. For the UART which has its dedicated interrupt node, interrupt status flags RI and TI in register SCON will not be cleared by the core even when its pending interrupt request is serviced. The UART interrupt status flags (and hence the pending interrupt request) can only be cleared by software.

12.5 Interrupt Source and Vector

Each interrupt event source has an associated interrupt vector address for the interrupt node it belongs to. This vector is accessed to service the corresponding interrupt node request. The interrupt service of each interrupt node can be individually enabled or disabled via an enable bit. The assignment of the TLE985xQX interrupt sources to the interrupt vector address and the corresponding interrupt node enable bits are summarized in **Table 173**.

Table 173 Interrupt Vector Address

Interrupt Node	Assignment for TLE985xQX	Enable Bit	SFR
NMI	Watchdog Timer NMI	NMIWDT	SCU_NMICON
	PLL NMI	NMIPLL	
	Overtemperature NMI	NMIOT	
	Oscillator Watchdog NMI	NMIOWD	
	NVM Map Error NMI	NMIMAP	
	ECC Error NMI	NMIECC	
	Supply Prewarning NMI	NMISUP	
	Stack Overflow	NMISTOF	
INTISR[0]	GPT1_T2, GPT1_T3, GPT1_T4	GPT12	GPT12IEN
INTISR[1]	GPT2_T5, GPT2_T6, GPT2_CR	GPT12	GPT12IEN
INTISR[2]	MU/Timer3	EMU	SYS_IRQ_CTRL
INTISR[3]	ADC10		ADC1
INTISR[4]	CCU6 Node 0	CCU6SR0	IRCON3.0
INTISR[5]	CCU6 Node 1	CCU6SR1	IRCON3.4
INTISR[6]	CCU6 Node 2	CCU6SR2	IRCON4.0
INTISR[7]	CCU6 Node 3	CCU6SRC3	IRCON4.4
INTISR[8]	SSC1	EIREN	MODIEN1.0
		TIREN	MODIEN1.1
		RIREN	MODIEN1.2
INTISR[9]	SSC2	EIREN	MODIEN2.0
		TIREN	MODIEN2.1
		RIREN	MODIEN2.2
INTISR[10]	UART1 Receive	RIEN	MODIEN1.0
	UART1 Transmit	TIEN	MODIEN1.1
	Timer 2 Overflow	TF2EN	T2_T2CON1.1
	T2EX	EXF2EN	T2_T2CON1.0
	LIN_OT_STS	LIN_OT_IE	SYS_IRQ_CTRL
	LIN_OC_STS	LIN_OC_IE	SYS_IRQ_CTRL
	TXD_TMOUT	LIN_TMOUT_IE	SYS_IRQ_CTRL
	EOFSYN	SYNEN	SYNCST
	ERRSYN		

Table 173 Interrupt Vector Address (cont'd)

Interrupt Node	Assignment for TLE985xQX	Enable Bit	SFR
INTISR[11]	UART2 Receive	RIEN	MODIEN2.0
	UART2 Transmit	TIEN	MODIEN2.1
	Timer 21 Overflow	TF2EN/	T21T2CON1.1
	T21EX	EXF2EN	T21T2CON1.0
INTISR[12]	EINT0	IE0	MODIEN3
INTISR[13]	EINT1	IE1	MODIEN4
INTISR[14]	Wake	WAKECON	WAKEUPEN
INTISR[15]	Math Divider End Of Conversion	DIVEOCIEN	MATH_EVIER
	Math Divider Error	DIVERRIEN	MATH_EVIER
INTISR[17]	VSD Upper Threshold	VSD_UPTH_IE	DRV_CP_IRQEN
	VSD Lower Threshold	VSD_LOTH_IE	DRV_CP_IRQEN
	VCP Upper Threshold	VCP_UPTH_IE	DRV_CP_IRQEN
	VCP Lower Threshold 1	VCP_LOTH1_IE	DRV_CP_IRQEN
	VCP Lower Threshold 2	VCP_LOTH2_IE	DRV_CP_IRQEN
	VCP Overtemperature Shutdown	VCP_OTSD_IE	DRV_CP_IRQEN
	VCP Overtemperature Warning	VCP_OTW_IE	DRV_CP_IRQEN
INTISR[18]	Low Side Driver 1 Overcurrent	LS1_OC_IE	DRV_IRQEN
	High Side Driver 1 Overcurrent	HS1_OC_IE	DRV_IRQEN
	Low Side Driver 2 Overcurrent	LS2_OC_IE	DRV_IRQEN
	High Side Driver 2 Overcurrent	HS2_OC_IE	DRV_IRQEN
	Low Side Driver 1 Drain Source Monitoring	LS1_DS_IE	DRV_IRQEN
	High Side Driver 1 Drain Source Monitoring	HS1_DS_IE	DRV_IRQEN
	Low Side Driver 2 Drain Source Monitoring	LS2_DS_IE	DRV_IRQEN
	High Side Driver 2 Drain Source Monitoring	HS2_DS_IE	DRV_IRQEN
	Driver Sequence Error	SEQ_ERR_IE	DRV_IRQEN
INTISR[19]	High Side Switch Overcurrent	HS1_OC_IEN	HS_IRQEN
	High Side Switch Overtemperature	HS1_OT_IEN	HS_IRQEN
	High Side Switch Open Load	HS1_OL_IEN	HS_IRQEN
INTISR[20]	CSA Lower Threshold	PP_CHx_LO_IEN	ADC1_IRQEN_2
	CSA Upper Threshold	PP_CHx_UP_IEN	ADC1_IRQEN_2
INTISR[21]	DU Upper Threshold	DU1UP_IEN	ADC1_IRQEN_1
	DU Lower Threshold	DU1LO_IEN	ADC1_IRQEN_1
INTISR[22]	MON1	MON1IE	MONIEN
	MON2	MON2IE	MONIEN
	MON3	MON3IE	MONIEN
	MON4	MON4IE	MONIEN

Table 173 Interrupt Vector Address (cont'd)

Interrupt Node	Assignment for TLE985xQX	Enable Bit	SFR
INTISR[23]	Port2.1	P2_1_UP_IEN / P2_1_LO_IEN	ADC1_IRQEN_2
	Port2.2	P2_2_UP_IEN / P2_2_LO_IEN	ADC1_IRQEN_2
	Port2.3	P2_3_UP_IEN / P2_3_LO_IEN	ADC1_IRQEN_2
	Port2.7	P2_7_UP_IEN / P2_7_LO_IEN	ADC1_IRQEN_2

12.6 Interrupt Priority

An interrupt that is currently being serviced can only be interrupted by a higher-priority interrupt, but not by another interrupt of the same or lower priority. Hence, an interrupt of the highest priority cannot be interrupted by any other interrupt request.

If two or more requests of different priority levels are received simultaneously, the request with the highest priority is serviced first. If requests of the same priority are received simultaneously, an internal polling sequence determines which request is serviced first. Thus, within each priority level, there is a second priority structure determined by the polling sequence as shown in **Table 172**.

Table 174 Interrupt Node Table

Service Request	Node ID	Description
GPT1	0	GPT1 interrupt (T2-T4)
GPT2	1	GPT2 interrupt (T5-T6, CR)
MU	2	Measurement Unit / ADC2, VBG
ADC1	3	ADC 10 bit interrupt
CCU0	4	CCU6 node 0 interrupt
CCU1	5	CCU6 node 1 interrupt
CCU2	6	CCU6 node 2 interrupt
CCU3	7	CCU6 node 3 interrupt
SSC1	8	SSC1 interrupt (receive, transmit, error)
SSC2	9	SSC2 interrupt (receive, transmit, error)
UART1	10	UART1 (ASC-LIN) interrupt (receive, transmit), t2, linsync1, LIN
UART2	11	UART2 interrupt (receive, transmit), t21, linsync2, External interrupt (EINT2)
EXINT0	12	External interrupt (EINT0), wake-up
EXINT1	13	External interrupt (EINT1)
WAKEUP	14	Wake-up interrupt
Math Div	15	Hardware Divider Unit Interrupt
	16	rfu
CP	17	Charge Pump Interrupt
BDRV	18	Bridge Driver Interrupt
HS1	19	High Side Driver 1
OPA	20	Operational Amplifier
DU	21	Differential Unit - DPP1
MONx	22	MONx Interrupt - DPP1
Port 2.x	23	Port 2.x Interrupt - DPP1

The interrupt priority is configured in the corresponding NVIC control register:

Table 175

Register Short name	Register Long Name	Offset Address	Reset Value
CPU_NVIC_IPR0	Interrupt Priority Register 0	400_H	$0000\ 0000_H$
CPU_NVIC_IPR1	Interrupt Priority Register 1	404_H	$0000\ 0000_H$
CPU_NVIC_IPR2	Interrupt Priority Register 2	408_H	$0000\ 0000_H$
CPU_NVIC_IPR3	Interrupt Priority Register 3	$40C_H$	$0000\ 0000_H$
CPU_NVIC_IPR4	Interrupt Priority Register 4	410_H	$0000\ 0000_H$
CPU_NVIC_IPR5	Interrupt Priority Register 5	414_H	$0000\ 0000_H$

For further description see ARM_Architecture_v7n_Reference_Manual.

12.7 Interrupt Handling

See also ARM_Architecture_v7n_Reference_Manual. The most important Interrupt Registers are listed below. This registers are dedicated to the 16 available interrupt nodes. For all nodes which are a combination of several interrupt requests, the corresponding control and status registers are located in the System Control Unit (SCU) or the System Control Unit for the Power Modules (SCU_PM).

Table 176

Register Short name	Register Long Name	Offset Address	Reset Value
CPU_NVIC_ISER	Interrupt Set-Enable	000_H	$0000\ 0000_H$
CPU_NVIC_ICER	Interrupt Clear-Enable	080_H	$0000\ 0000_H$
CPU_NVIC_ISPR	Interrupt Set-Pending	100_H	$0000\ 0000_H$
CPU_NVIC_ICPR	Interrupt Clear-Pending	180_H	$0000\ 0000_H$

12.8 Interrupt Registers

Interrupt registers are used for interrupt node enable, external interrupt control, interrupt flags and interrupt priority setting.

Table 177 Register Address Space

Module	Base Address	End Address	Note
SCU	50005000_H	$50005FFF_H$	SCU

Table 178 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Interrupt Registers, Interrupt Node Enable Registers			
SCU_IEN0	Interrupt Enable Register 0	$01C_H$	$0000\ 0000_H$
SCU_NMICON	NMI Control Register	024_H	$0000\ 0000_H$
Interrupt Registers, External Interrupt Control Registers			
SCU_EXICON0	External Interrupt Control Register 0	028_H	$0000\ 0030_H$
SCU_EXICON1	External Interrupt Control Register 1	$02C_H$	$0000\ 0030_H$
SCU_WAKECON	Wake-Up Interrupt Control Register	078_H	$0000\ 0000_H$
Interrupt Registers, Interrupt Flag Registers			
SCU_SCON1	UART1 Control/Status Register	xxx_H	$0000\ 0000_H$
SCU_SCON2	UART2 Control/Status Register	xxx_H	$0000\ 0000_H$
SCU_IRCON0	Interrupt Request Register 0	004_H	$0000\ 0000_H$
SCU_IRCON0CLR	Interrupt Request 0 Clear Register	178_H	$0000\ 0000_H$
SCU_IRCON1	Interrupt Request Register 1	008_H	$0000\ 0000_H$
SCU_IRCON1CLR	Interrupt Request 1 Clear Register	$17C_H$	$0000\ 0000_H$
SCU_IRCON2	Interrupt Request Register 2	$00C_H$	$0000\ 0000_H$
SCU_IRCON2CLR	Interrupt Request 2 Clear Register	190_H	$0000\ 0000_H$
SCU_IRCON3	Interrupt Request Register 3	010_H	$0000\ 0000_H$
SCU_IRCON3CLR	Interrupt Request 3 Clear Register	194_H	$0000\ 0000_H$
SCU_IRCON4	Interrupt Request Register 4	014_H	$0000\ 0000_H$
SCU_IRCON4CLR	Interrupt Request 4 Clear Register	198_H	$0000\ 0000_H$
SCU_IRCON5	Interrupt Request Register 5	$07C_H$	$0000\ 0000_H$
SCU_IRCON5CLR	Interrupt Request 5 Clear Register	$19C_H$	$0000\ 0000_H$
SCU_NMISR	NMI Status Register	018_H	$0000\ 0000_H$
SCU_NMISRCLR	NMI Status Clear Register	000_H	$0000\ 0000_H$
SCU_GPT12IRC	Timer and Counter Control/Status Register	160_H	$0000\ 0000_H$
SCU_GPT12ICLR	Timer and Counter Control/Status Clear Register	180_H	$0000\ 0000_H$

The registers are addressed wordwise.

12.8.1 Interrupt Node Enable Registers

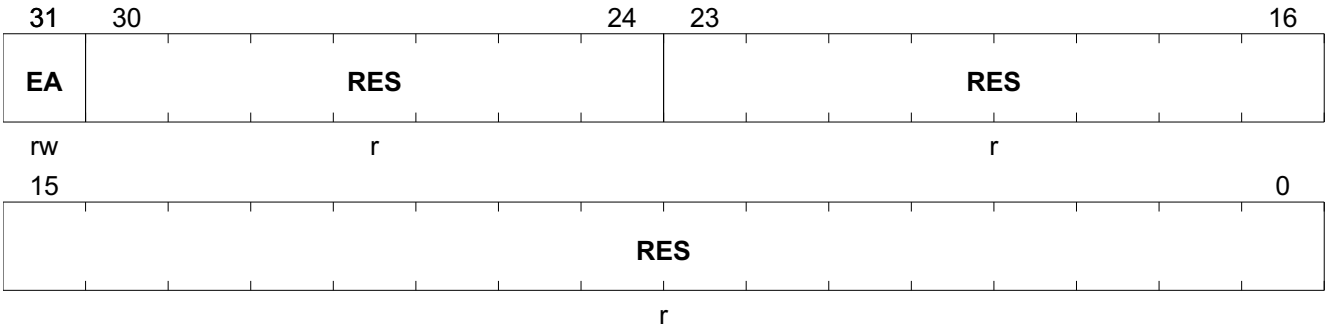
Register IEN0 contains the global interrupt masking bit (EA), which can be cleared to block all pending interrupt requests at once.

The NMI interrupt vector is shared by a number of sources, each of which can be enabled or disabled individually via register NMICON.

After reset, the enable bits in IEN0, IEN1 and NMICON are cleared to 0. This implies that all interrupt nodes are disabled by default.

Interrupt Enable Register 0

SCU_IEN0	Offset	Reset Value
Interrupt Enable Register 0	$01C_H$	see Table 179



Field	Bits	Type	Description
EA	31	rw	**Global Interrupt Mask** 0_B **Disable**, All pending interrupt requests (except NMI) are blocked from the core. 1_B **Enable**, Pending interrupt requests are not blocked from the core.
RES	30:24	r	**Reserved** Returns 0 if read; should be written with 0.
RES	23:0	r	**Reserved** Returns 0 if read; should be written with 0.

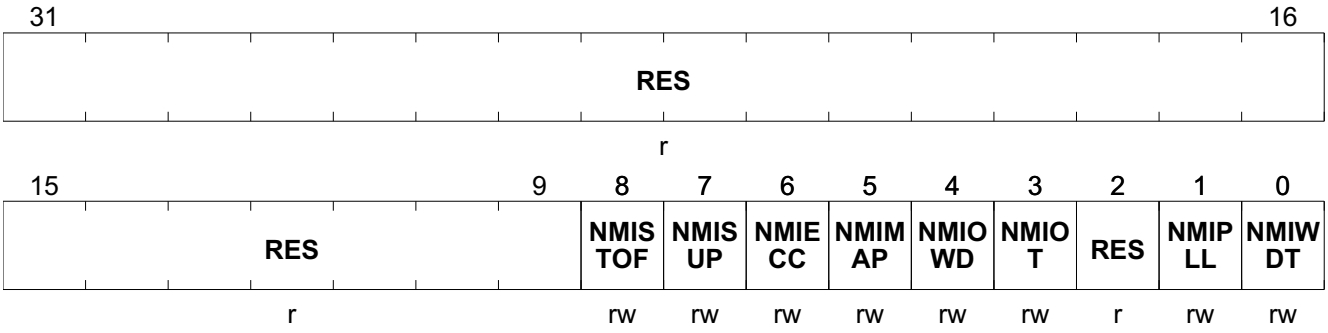
Table 179 RESET of SCU_IEN0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		

Interrupt System

NMI Control Register

SCU_NMICON	Offset	Reset Value
NMI Control Register	024_H	see Table 180



Field	Bits	Type	Description
RES	31:9	r	**Reserved** Returns 0 if read; should be written with 0.
NMISTOF	8	rw	**Stack Overflow NMI Enable** 0_B **disable**, Stack overflow NMI is disabled. 1_B **enable**, Stack overflow NMI is enabled.
NMISUP	7	rw	**Supply Prewarning NMI Enable** 0_B **disable**, Supply NMI is disabled. 1_B **enable**, Supply NMI is enabled.
NMIECC	6	rw	**ECC Error NMI Enable** 0_B **disable**, ECC Error NMI is disabled. 1_B **enable**, ECC Error NMI is enabled.
NMIMAP	5	rw	**NVM Map Error NMI Enable** 0_B **disable**, NVM Map Error NMI is disabled. 1_B **enable**, NVM Map Error NMI is enabled.
NMIOWD	4	rw	**Oscillator Watchdog NMI Enable** 0_B **disable**, Oscillator watchdog NMI is disabled. 1_B **enable**, Oscillator watchdog NMI is enabled.
NMIOT	3	rw	**NMI OT Enable** 0_B **disable**, NMI OT is disabled. 1_B **enable**, NMI OT is enabled.
RES	2	r	**Reserved** Returns 0 if read; should be written with 0.
NMIPLL	1	rw	**PLL Loss of Lock NMI Enable** 0_B **disable**, PLL Loss of Lock NMI is disabled. 1_B **enable**, PLL Loss of Lock NMI is enabled.
NMIWDT	0	rw	**Watchdog Timer NMI Enable** 0_B **disable**, WDT NMI is disabled. 1_B **enable**, WDT NMI is enabled.

Table 180 RESET of SCU_NMICON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

12.8.2 External Interrupt Control Registers

The external interrupts, EXT_INT[1:0], are driven into the XC8_EPOWER from the ports. External interrupts can be positive, negative or double edge triggered. Register EXICON0 specifies the active edge for the external interrupt. An active edge event detected in SCU will generate internally two CCLK cycle low pulse for detection by core.

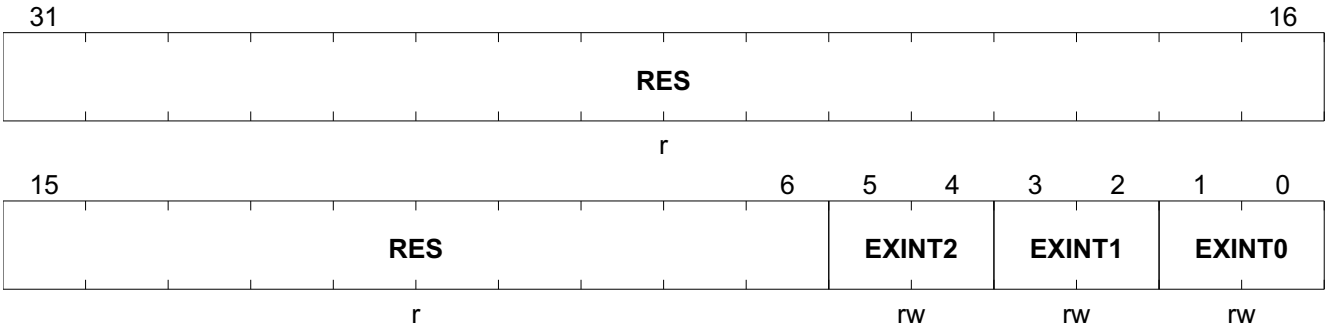
If the external interrupt is positive (negative) edge triggered, the external source must hold the request pin low (high) for at least one CCLK cycle, and then hold it high (low) for at least one CCLK cycle to ensure that the transition is recognized.

External interrupt 2 share the interrupt node with other interrupt sources. Therefore in addition to the corresponding interrupt node enable, external interrupt 2 may be disabled individually, and is disabled by default after reset.

Note: Several external interrupts support alternative input pin, selected via MODPISEL register in the SCU. When switching inputs, the active edge/level trigger select and the level on the associated pins should be considered to prevent unintentional interrupt generation.

External Interrupt Control Register 0

SCU_EXICON0	Offset	Reset Value
External Interrupt Control Register 0	**028_H**	**see Table 181**



Field	Bits	Type	Description
RES	31:6	r	**Reserved** Returns 0 if read; should be written with 0.
EXINT2	5:4	rw	**External Interrupt 2 Trigger Select** 00_B **Disable**, Interrupt disabled. 01_B **Rising Edge**, Interrupt on rising edge. 10_B **Falling Edge**, Interrupt on falling edge. 11_B **Both Edges**, Interrupt on both rising and falling edge.

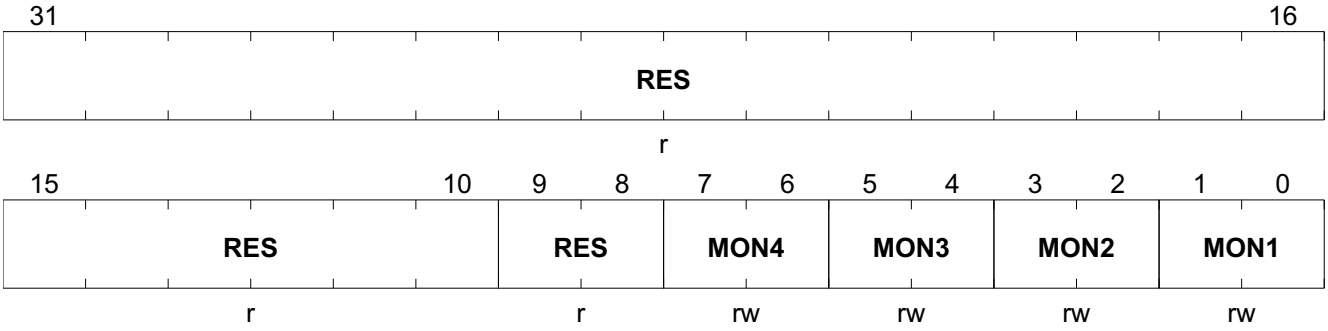
Field	Bits	Type	Description
EXINT1	3:2	rw	**External Interrupt 1 Trigger Select** 00_B **Disable**, Interrupt disabled. 01_B **Rising Edge**, Interrupt on rising edge. 10_B **Falling Edge**, Interrupt on falling edge. 11_B **Both Edges**, Interrupt on both rising and falling edge.
EXINT0	1:0	rw	**External Interrupt 0 Trigger Select** 00_B **Disable**, Interrupt disabled. 01_B **Rising Edge**, Interrupt on rising edge. 10_B **Falling Edge**, Interrupt on falling edge. 11_B **Both Edges**, Interrupt on both rising and falling edge.

Table 181 RESET of SCU_EXICON0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0030_H	RESET_TYPE_3		

External Interrupt Control Register 1

SCU_EXICON1	Offset	Reset Value
External Interrupt Control Register 1	**$02C_H$**	**see Table 182**



Field	Bits	Type	Description
RES	31:10	r	**Reserved** Returns 0 if read; should be written with 0.
RES	9:8	r	**Reserved** Returns 0 if read; should be written with 0.
MON4	7:6	rw	**MON4 Input Trigger Select** 00_B **Disable**, external interrupt MON is disabled. 01_B **Rising Edge**, Interrupt on rising edge. 10_B **Falling Edge**, Interrupt on falling edge. 11_B **Both Edges**, Interrupt on both rising and falling edge.

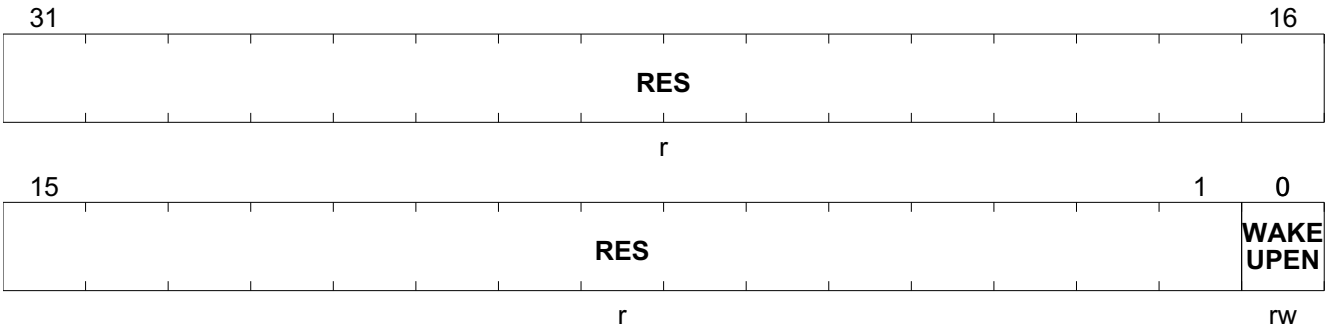
Field	Bits	Type	Description
MON3	5:4	rw	**MON3 Input Trigger Select** 00_B **Disable**, external interrupt MON is disabled. 01_B **Rising Edge**, Interrupt on rising edge. 10_B **Falling Edge**, Interrupt on falling edge. 11_B **Both Edges**, Interrupt on both rising and falling edge.
MON2	3:2	rw	**MON2 Input Trigger Select** 00_B **Disable**, external interrupt MON is disabled. 01_B **Rising Edge**, Interrupt on rising edge. 10_B **Falling Edge**, Interrupt on falling edge. 11_B **Both Edges**, Interrupt on both rising and falling edge.
MON1	1:0	rw	**MON1 Input Trigger Select** 00_B **Disable**, external interrupt MON is disabled. 01_B **Rising Edge**, Interrupt on rising edge. 10_B **Falling Edge**, Interrupt on falling edge. 11_B **Both Edges**, Interrupt on both rising and falling edge.

Table 182 RESET of SCU_EXICON1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Wake-Up Interrupt Control Register

SCU_WAKECON	Offset	Reset Value
Wakeup Interrupt Control Register	$0EC_H$	see Table 183



Field	Bits	Type	Description
RES	31:1	r	**Reserved** Returns 0 if read; should be written with 0.
WAKEUPEN	0	rw	**Wake-Up Interrupt Enable** 0_B **Disable**, wake-up interrupt is disabled. 1_B **Enable**, wake-up interrupt is enabled.

Table 183 RESET of SCU_WAKECON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

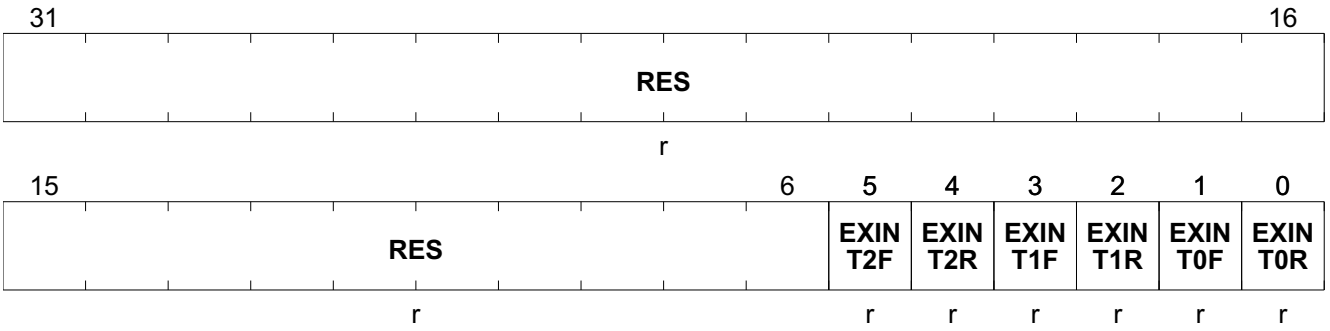
12.8.3 Interrupt Flag Registers

The interrupt flags for the different interrupt sources are located in several special function registers. This section describes the interrupt flags located in system registers or external interrupts belonging to system. Other interrupt flags located in respective module registers are described in the specific module chapter. For a complete listing of the interrupt flags and their assignment to SFRs, refer to **Table 172**.

In case of software and hardware access to a flag bit at the same time, hardware will have higher priority.

Interrupt Request Register 0

SCU_IRCON0	Offset	Reset Value
Interrupt Request Register 0	004_H	see Table 184



Field	Bits	Type	Description
RES	31:6	r	**Reserved** Returns 0 if read; should be written with 0.
EXINT2F	5	r	**Interrupt Flag for External Interrupt 2x on falling edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on falling edge event has not occurred. 1_B **Int**, Interrupt on falling edge event has occurred.
EXINT2R	4	r	**Interrupt Flag for External Interrupt 2x on rising edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on rising edge event has not occurred. 1_B **Int**, Interrupt on rising edge event has occurred.
EXINT1F	3	r	**Interrupt Flag for External Interrupt 1x on falling edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on falling edge event has not occurred. 1_B **Int**, Interrupt on falling edge event has occurred.

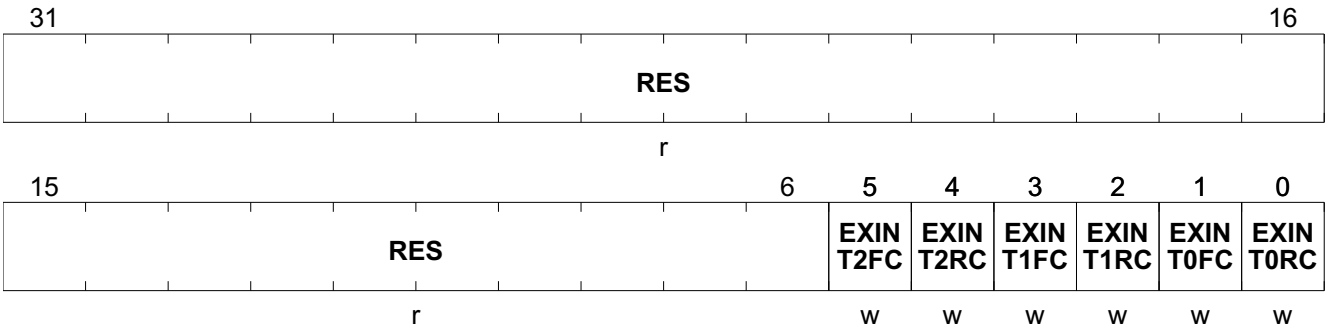
Field	Bits	Type	Description
EXINT1R	2	r	**Interrupt Flag for External Interrupt 1x on rising edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on rising edge event has not occurred. 1_B **Int**, Interrupt on rising edge event has occurred.
EXINT0F	1	r	**Interrupt Flag for External Interrupt 0x on falling edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on falling edge event has not occurred. 1_B **Int**, Interrupt on falling edge event has occurred.
EXINT0R	0	r	**Interrupt Flag for External Interrupt 0x on rising edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on rising edge event has not occurred. 1_B **Int**, Interrupt on rising edge event has occurred.

Table 184 RESET of SCU_IRCON0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Interrupt Request 0 Clear Register

SCU_IRCON0CLR	Offset	Reset Value
Interrupt Request 0 Clear Register	**178_H**	**see Table 185**



Field	Bits	Type	Description
RES	31:6	r	**Reserved** Returns 0 if read; should be written with 0.
EXINT2FC	5	w	**Interrupt Flag for External Interrupt 2x on falling edge** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared

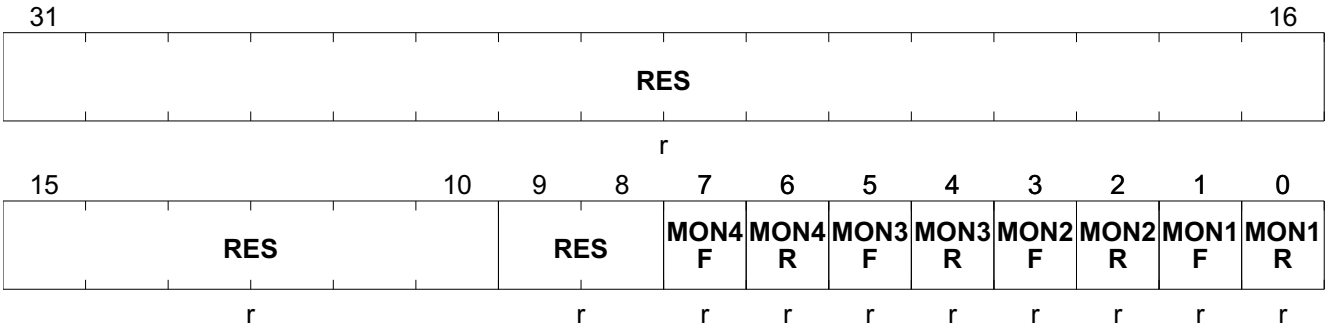
Field	Bits	Type	Description
EXINT2RC	4	w	**Interrupt Flag for External Interrupt 2x on rising edge** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared
EXINT1FC	3	w	**Interrupt Flag for External Interrupt 1x on falling edge** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared
EXINT1RC	2	w	**Interrupt Flag for External Interrupt 1x on rising edge** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared
EXINT0FC	1	w	**Interrupt Flag for External Interrupt 0x on falling edge** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared
EXINT0RC	0	w	**Interrupt Flag for External Interrupt 0x on rising edge** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared

Table 185 RESET of SCU_IRCON0CLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000_H	RESET_TYPE_3		

Interrupt Request Register 1

SCU_IRCON1	Offset	Reset Value
Interrupt Request Register 1	**008_H**	**see Table 186**



Field	Bits	Type	Description
RES	31:10	r	**Reserved** Returns 0 if read; should be written with 0.
RES	9:8	r	**Reserved** Returns 0 if read; should be written with 0.

Field	Bits	Type	Description
MON4F	7	r	**Interrupt Flag for MON4x on falling edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on falling edge event has not occurred. 1_B **Int**, Interrupt on falling edge event has occurred.
MON4R	6	r	**Interrupt Flag for MON4x on rising edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on rising edge event has not occurred. 1_B **Int**, Interrupt on rising edge event has occurred.
MON3F	5	r	**Interrupt Flag for MON3x on falling edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on falling edge event has not occurred. 1_B **Int**, Interrupt on falling edge event has occurred.
MON3R	4	r	**Interrupt Flag for MON3x on rising edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on rising edge event has not occurred. 1_B **Int**, Interrupt on rising edge event has occurred.
MON2F	3	r	**Interrupt Flag for MON2x on falling edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on falling edge event has not occurred. 1_B **Int**, Interrupt on falling edge event has occurred.
MON2R	2	r	**Interrupt Flag for MON2x on rising edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on rising edge event has not occurred. 1_B **Int**, Interrupt on rising edge event has occurred.
MON1F	1	r	**Interrupt Flag for MON1x on falling edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on falling edge event has not occurred. 1_B **Int**, Interrupt on falling edge event has occurred.

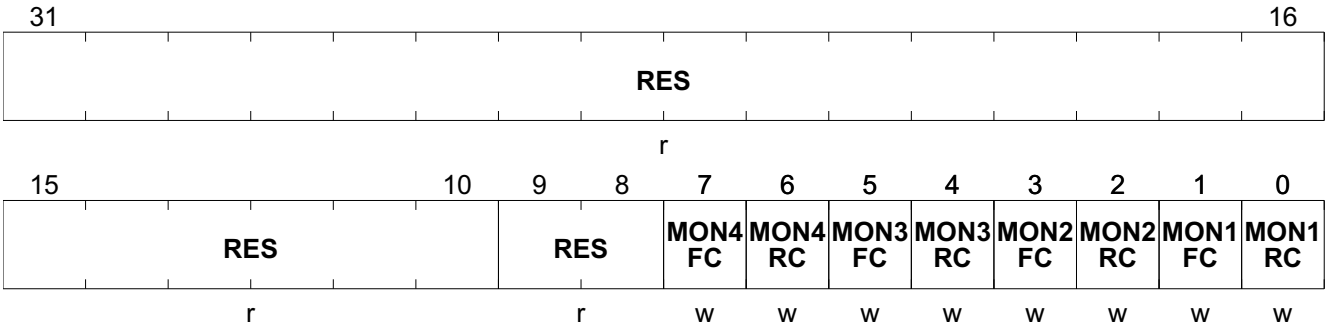
Field	Bits	Type	Description
MON1R	0	r	**Interrupt Flag for MON1x on rising edge** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt on rising edge event has not occurred. 1_B **Int**, Interrupt on rising edge event has occurred.

Table 186 RESET of SCU_IRCON1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Interrupt Request 1 Clear Register

SCU_IRCON1CLR	Offset	Reset Value
Interrupt Request 1 Clear Register	$17C_H$	see Table 187



Field	Bits	Type	Description
RES	31:10	r	**Reserved** Returns 0 if read; should be written with 0.
RES	9:8	r	**Reserved** Returns 0 if read; should be written with 0.
MON4FC	7	w	**Interrupt Flag for MON4x on falling edge** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared
MON4RC	6	w	**Interrupt Flag for MON4x on rising edge** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared.
MON3FC	5	w	**Interrupt Flag for MON3x on falling edge** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared
MON3RC	4	w	**Interrupt Flag for MON3x on rising edge** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared

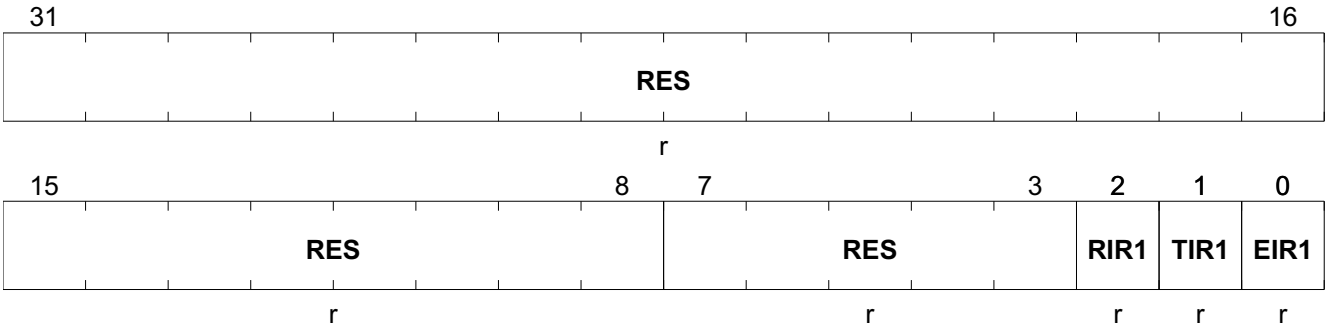
Field	Bits	Type	Description
MON2FC	3	w	**Interrupt Flag for MON2x on falling edge** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared
MON2RC	2	w	**Interrupt Flag for MON2x on rising edge** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared
MON1FC	1	w	**Interrupt Flag for MON1x on falling edge** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared
MON1RC	0	w	**Interrupt Flag for MON1x on rising edge** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared

Table 187 RESET of SCU_IRCON1CLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Interrupt Request Register 2

SCU_IRCON2	Offset	Reset Value
Interrupt Request Register 2	$00C_H$	see Table 188



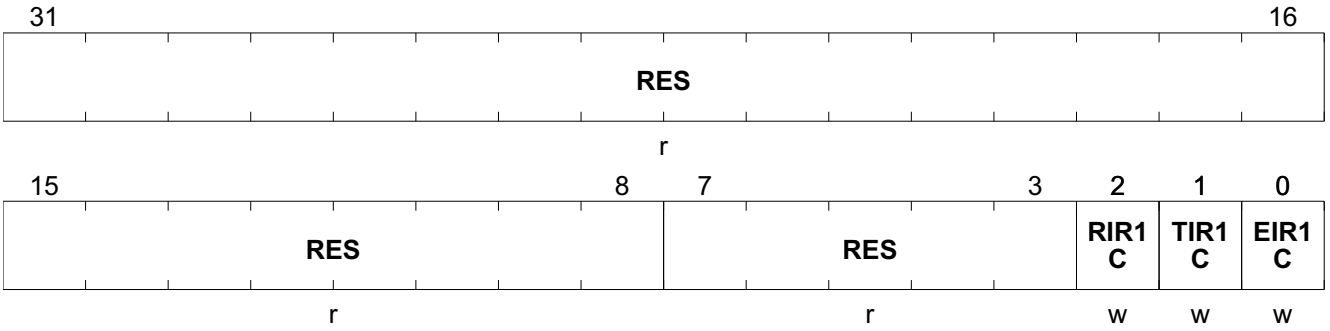
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
RES	7:3	r	**Reserved** Returns 0 if read; should be written with 0.
RIR1	2	r	**Receive Interrupt Flag for SSC1** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt event has not occurred. 1_B **Int**, Interrupt event has occurred.
TIR1	1	r	**Transmit Interrupt Flag for SSC1** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt event has not occurred. 1_B **Int**, Interrupt event has occurred.
EIR1	0	r	**Error Interrupt Flag for SSC1** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt event has not occurred. 1_B **Int**, Interrupt event has occurred.

Table 188 RESET of SCU_IRCON2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Interrupt Request 2 Clear Register

SCU_IRCON2CLR	Offset	Reset Value
Interrupt Request 2 Clear Register	190_H	see Table 189



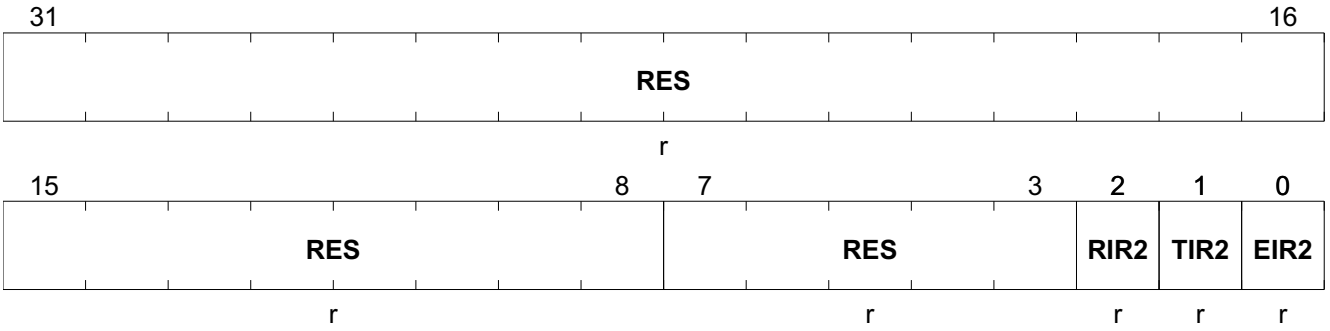
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
RES	7:3	r	**Reserved** Returns 0 if read; should be written with 0.
RIR1C	2	w	**Receive Interrupt Flag for SSC1** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared.
TIR1C	1	w	**Transmit Interrupt Flag for SSC1** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared.
EIR1C	0	w	**Error Interrupt Flag for SSC1** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared.

Table 189 RESET of SCU_IRCON2CLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000_H	RESET_TYPE_3		

Interrupt Request Register 3

SCU_IRCON3	Offset	Reset Value
Interrupt Request Register 3	010_H	see Table 190



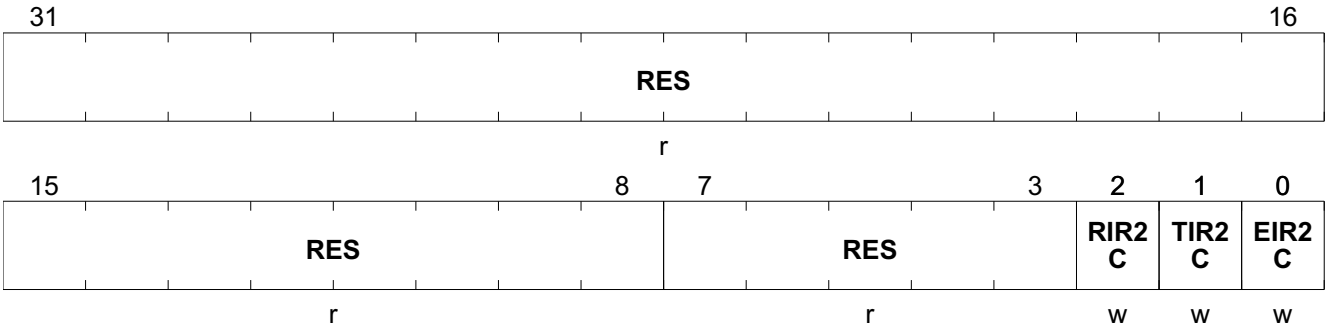
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
RES	7:3	r	**Reserved** Returns 0 if read; should be written with 0.
RIR2	2	r	**Receive Interrupt Flag for SSC2** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt event has not occurred. 1_B **Int**, Interrupt event has occurred.
TIR2	1	r	**Transmit Interrupt Flag for SSC2** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt event has not occurred. 1_B **Int**, Interrupt event has occurred.
EIR2	0	r	**Error Interrupt Flag for SSC2** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt event has not occurred. 1_B **Int**, Interrupt event has occurred.

Table 190 RESET of SCU_IRCON3

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000_H	RESET_TYPE_3		

Interrupt Request 3 Clear Register

SCU_IRCON3CLR	Offset	Reset Value
Interrupt Request 3 Clear Register	194_H	see Table 191



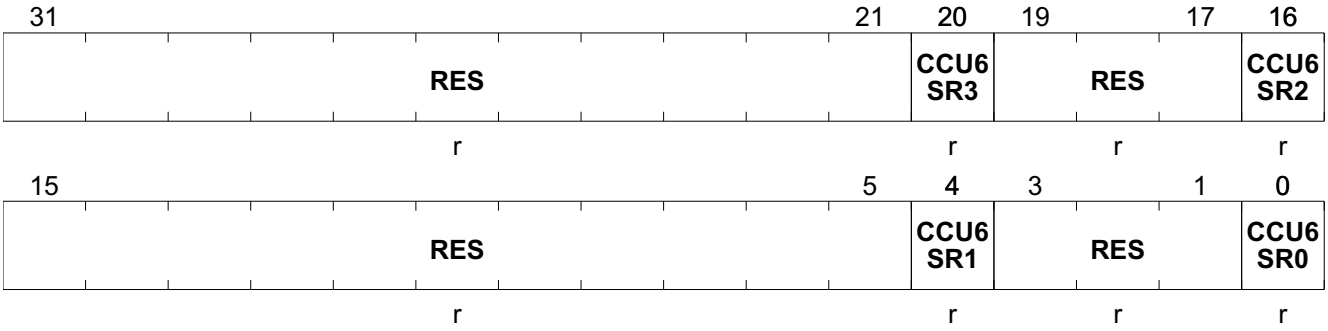
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
RES	7:3	r	**Reserved** Returns 0 if read; should be written with 0.
RIR2C	2	w	**Receive Interrupt Flag for SSC2** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared.
TIR2C	1	w	**Transmit Interrupt Flag for SSC2** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared.
EIR2C	0	w	**Error Interrupt Flag for SSC2** 0_B **No Clear**, Interrupt event is not cleared. 1_B **Clear**, Interrupt event is cleared.

Table 191 RESET of SCU_IRCON3CLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Interrupt Request Register 4

SCU_IRCON4	Offset	Reset Value
Interrupt Request Register 4	014_H	see Table 192



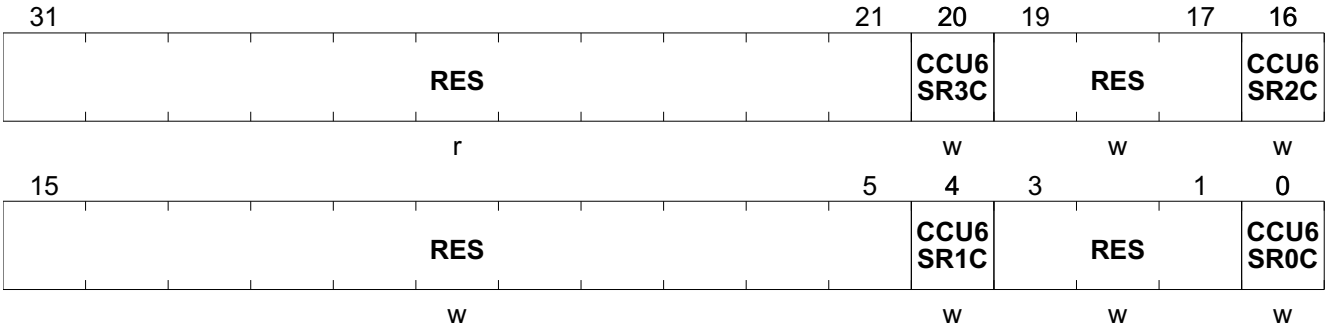
Field	Bits	Type	Description
RES	31:21	r	**Reserved** Returns 0 if read; should be written with 0.
CCU6SR3	20	r	**Interrupt Flag 1 for CCU6** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt event has not occurred. 1_B **Int**, Interrupt event has occurred.
RES	19:17	r	**Reserved** Returns 0 if read; should be written with 0.
CCU6SR2	16	r	**Interrupt Flag 1 for CCU6** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt event has not occurred. 1_B **Int**, Interrupt event has occurred.
RES	15:5	r	**Reserved** Returns 0 if read; should be written with 0.
CCU6SR1	4	r	**Interrupt Flag 1 for CCU6** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt event has not occurred. 1_B **Int**, Interrupt event has occurred.
RES	3:1	r	**Reserved** Returns 0 if read; should be written with 0.
CCU6SR0	0	r	**Interrupt Flag 1 for CCU6** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt event has not occurred. 1_B **Int**, Interrupt event has occurred.

Table 192 RESET of SCU_IRCON4

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Interrupt Request 4 Clear Register

SCU_IRCON4CLR	Offset	Reset Value
Interrupt Request 4 Clear Register	198_H	see Table 193



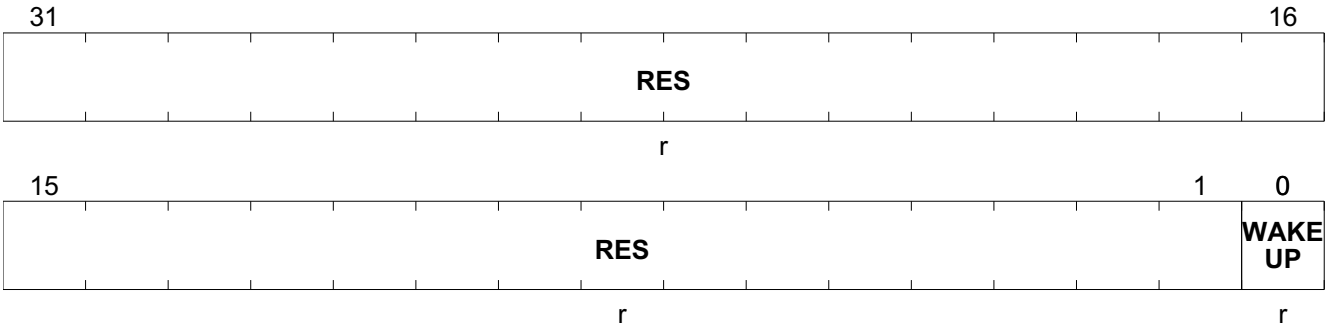
Field	Bits	Type	Description
RES	31:21	r	**Reserved** Returns 0 if read; should be written with 0.
CCU6SR3C	20	w	**Interrupt Flag 1 for CCU6** 0_B **No Clear**, Interrupt event is not cleared 1_B **Clear**, Interrupt event is cleared
RES	19:17	w	**Reserved** Returns 0 if read; should be written with 0.
CCU6SR2C	16	w	**Interrupt Flag 1 for CCU6** 0_B **No Clear**, Interrupt event is not cleared 1_B **Clear**, Interrupt event is cleared
RES	15:5	w	**Reserved** Returns 0 if read; should be written with 0.
CCU6SR1C	4	w	**Interrupt Flag 1 for CCU6** 0_B **No Clear**, Interrupt event is not cleared 1_B **Clear**, Interrupt event is cleared
RES	3:1	w	**Reserved** Returns 0 if read; should be written with 0.
CCU6SR0C	0	w	**Interrupt Flag 1 for CCU6** 0_B **No Clear**, Interrupt event is not cleared 1_B **Clear**, Interrupt event is cleared

Table 193 RESET of SCU_IRCON4CLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Interrupt Request Register 5

SCU_IRCON5	Offset	Reset Value
Interrupt Request Register 5	$0F0_H$	see Table 194



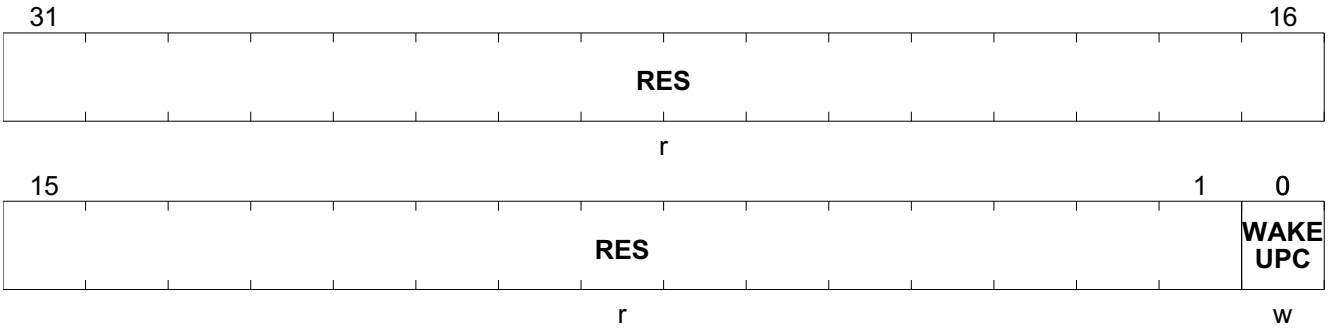
Field	Bits	Type	Description
RES	31:1	r	**Reserved** Returns 0 if read; should be written with 0.
WAKEUP	0	r	**Interrupt Flag for Wake-Up** This bit is set by hardware and can only be cleared by software. 0_B **No Int**, Interrupt event has not occurred. 1_B **Int**, Interrupt event has occurred.

Table 194 RESET of SCU_IRCON5

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Interrupt Request 5 Clear Register

SCU_IRCON5CLR	Offset	Reset Value
Interrupt Request 5 Clear Register	$19C_H$	see Table 195



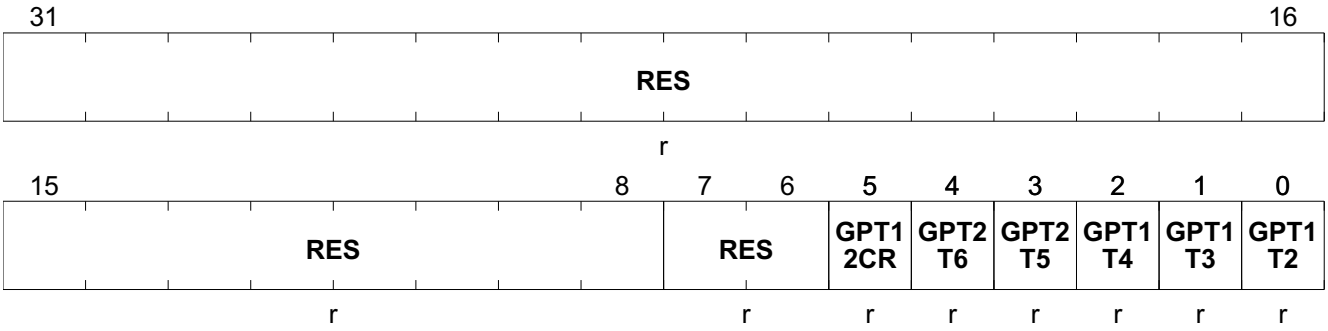
Field	Bits	Type	Description
RES	31:1	r	**Reserved** Returns 0 if read; should be written with 0.
WAKEUPC	0	w	**Clear Flag for Wake-Up Interrupt** 0_B **No Clear**, Interrupt event is not cleared 1_B **Clear**, Interrupt event is cleared

Table 195 RESET of SCU_IRCON5CLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Timer and Counter Control/Status Register

SCU_GPT12IRC	Offset	Reset Value
Timer and Counter Control/Status Register	160_H	see Table 196



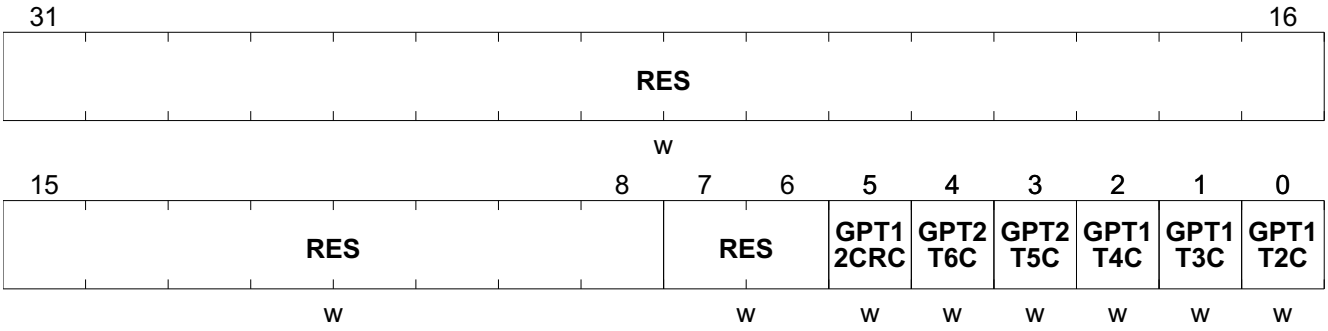
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
RES	7:6	r	**Reserved** Returns 0 if read; should be written with 0.
GPT12CR	5	r	**GPT Module 1 Capture Reload Interrupt Status** Capture Reload Event of GPT1 Module Interrupt Status 0_B **No Int**, No Capture Reload Interrupt has occurred. 1_B **Int**, Capture Reload Interrupt has occurred.
GPT2T6	4	r	**GPT Module 2Timer6 Interrupt Status** Timer 6 of GPT Module Interrupt Status 0_B **No Int**, No Timer 6 Interrupt has occurred. 1_B **Int**, Timer 6 Interrupt has occurred.
GPT2T5	3	r	**GPT Module 2 Timer5 Interrupt Status** Timer 5 of GPT2 Module Interrupt Status 0_B **No Int**, No Timer 5 Interrupt has occurred. 1_B **Int**, Timer 5 Interrupt has occurred.
GPT1T4	2	r	**GPT Module 1 Timer4 Interrupt Status** Timer 4 of GPT1 Module Interrupt Status 0_B **No Int**, No Timer 4 Interrupt has occurred. 1_B **Int**, Timer 4 Interrupt has occurred.
GPT1T3	1	r	**GPT Module 1 Timer3 Interrupt Status** Timer 3 of GPT1 Module Interrupt Status 0_B **No Int**, No Timer 3 Interrupt has occurred. 1_B **Int**, Timer 3 Interrupt has occurred.
GPT1T2	0	r	**GPT Module 1 Timer 2 Interrupt Status** Timer 2 of GPT1 Module Interrupt Status 0_B **No Int**, No Timer 2 Interrupt has occurred. 1_B **Int**, Timer 2 Interrupt has occurred.

Table 196 RESET of SCU_GPT12IRC

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Timer and Counter Control/Status Register

SCU_GPT12ICLR	Offset	Reset Value
Timer and Counter Control/Status Clear Register	180_H	see Table 197



Field	Bits	Type	Description
RES	31:8	w	**Reserved** Returns 0 if read; should be written with 0.
RES	7:6	w	**Reserved** Returns 0 if read; should be written with 0.
GPT12CRC	5	w	**GPT Module 1 Capture Reload Interrupt Status** Capture Reload Event of GPT1 Module Interrupt Status 0_B **No Clear**, Interrupt event is not cleared 1_B **Clear**, Interrupt event is cleared
GPT2T6C	4	w	**GPT Module 2 Timer6 Interrupt Status** Timer 6 of GPT Module Interrupt Status 0_B **No Clear**, Interrupt event is not cleared 1_B **Clear**, Interrupt event is cleared
GPT2T5C	3	w	**GPT Module 2 Timer5 Interrupt Status** Timer 5 of GPT2 Module Interrupt Status 0_B **No Clear**, Interrupt event is not cleared 1_B **Clear**, Interrupt event is cleared
GPT1T4C	2	w	**GPT Module 1 Timer4 Interrupt Status** Timer 4 of GPT1 Module Interrupt Status 0_B **No Clear**, Interrupt event is not cleared 1_B **Clear**, Interrupt event is cleared
GPT1T3C	1	w	**GPT Module 1 Timer3 Interrupt Status** Timer 3 of GPT1 Module Interrupt Status 0_B **No Clear**, Interrupt event is not cleared 1_B **Clear**, Interrupt event is cleared
GPT1T2C	0	w	**GPT Module 1 Timer 2 Interrupt Status** Timer 2 of GPT1 Module Interrupt Status 0_B **No Clear**, Interrupt event is not cleared 1_B **Clear**, Interrupt event is cleared

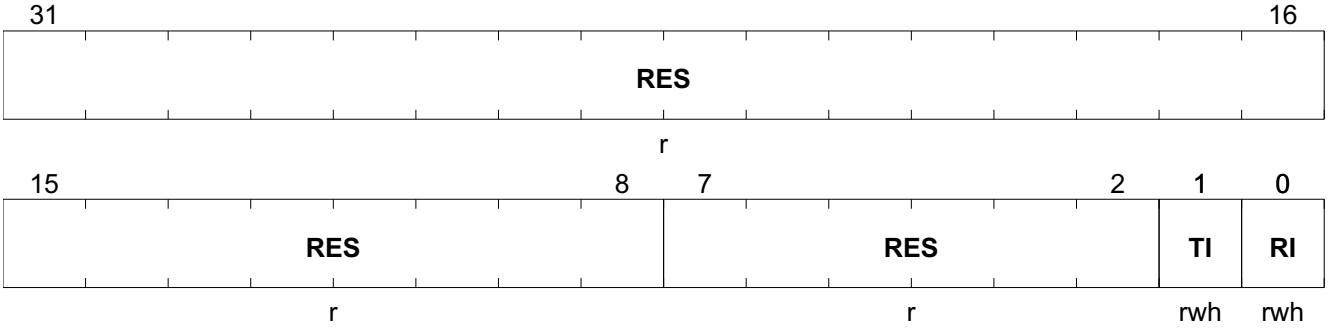
Table 197 RESET of SCU_GPT12ICLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

UART1 Control/Status Register

Refer to Register **UART_SCON** in **Chapter 19**.

SCU_SCON1	Offset	Reset Value
UART1 Control/Status Register	xxx_H	see Table 198



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
RES	7:2	r	**Reserved** Returns 0 if read; should be written with 0.
TI	1	rwh	**Serial Interface Transmitter Interrupt Flag** Set by hardware at the end of a serial data transmission. Must be cleared by software.
RI	0	rwh	**Serial Interface Receiver Interrupt Flag** Set by hardware if a serial data byte has been received. Must be cleared by software.

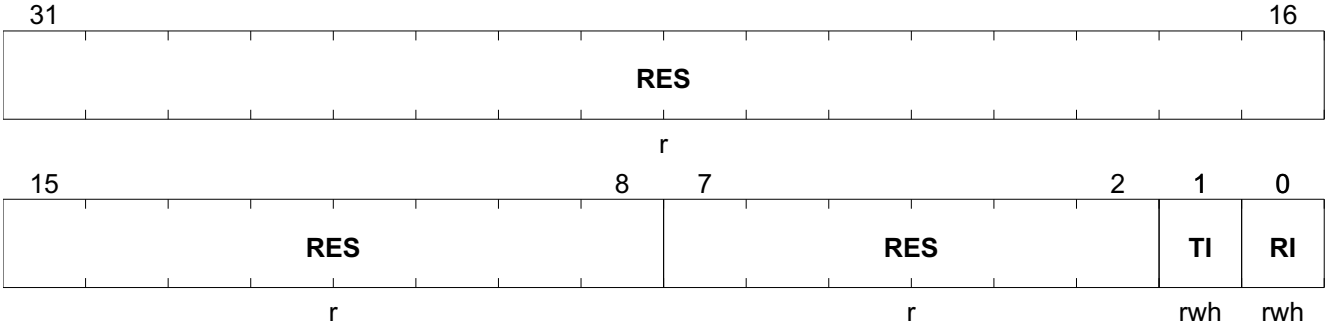
Table 198 RESET of SCU_SCON1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

UART2 Control/Status Register

Refer to Register **UART_SCON** in **Chapter 19**.

SCU_SCON2	Offset	Reset Value
UART2 Control/Status Register	xxx_H	see Table 199



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
RES	7:2	r	**Reserved** Returns 0 if read; should be written with 0.
TI	1	rwh	**Serial Interface Transmitter Interrupt Flag** Set by hardware at the end of a serial data transmission. Must be cleared by software.
RI	0	rwh	**Serial Interface Receiver Interrupt Flag** Set by hardware if a serial data byte has been received. Must be cleared by software.

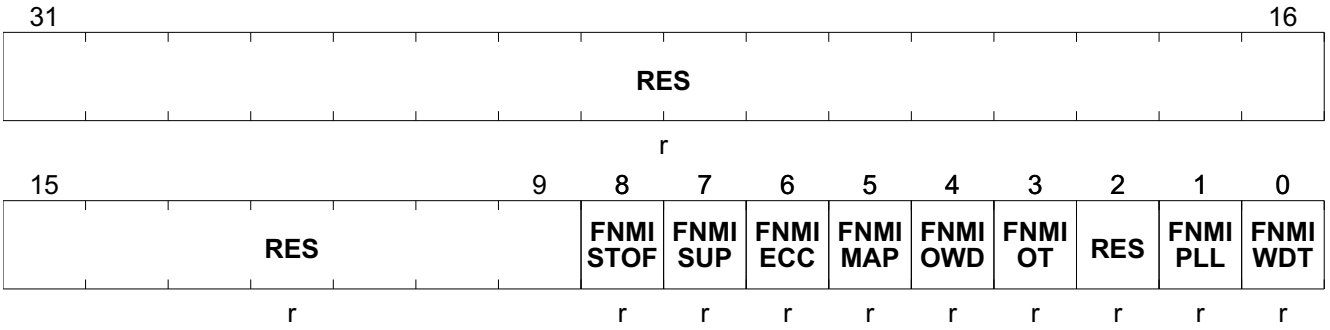
Table 199 RESET of SCU_SCON2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

NMI Status Register

Each NMI event and status flag is retained across these resets: 1) WDT reset, 2) soft reset. These include all the flags of NMISR register: FNMIWDT, FNMIPLL, FNMIOT, FNMIOWD, FNMIMAP, and indirectly, FNMIECC and FNMISUP. In the case of NMIs with shared source i.e. watchdog, ECC or supply prewarning NMI, the respective indicator or event flags not located in NMISR are also retained. Refer to Chapter 1.6.5 for identifying the NMI event.

SCU_NMISR	Offset	Reset Value
NMI Status Register	018_H	see Table 200



Field	Bits	Type	Description
RES	31:9	r	**Reserved** Returns 0 if read; should be written with 0.
FNMISTOF	8	r	**Stack Overflow NMI Flag** This flag is cleared automatically by hardware when the corresponding event flags are cleared. 0_B **no Int**, No supply prewarning NMI has occurred. 1_B **Int**, Supply prewarning has occurred.
FNMISUP	7	r	**Supply Prewarning NMI Flag** This flag is cleared automatically by hardware when the corresponding event flags are cleared. 0_B **no Int**, No supply prewarning NMI has occurred. 1_B **Int**, Supply prewarning has occurred.
FNMIECC	6	r	**ECC Error NMI Flag** This flag is cleared automatically by hardware when the corresponding enabled event flags are cleared. 0_B **no Int**, No uncorrectable ECC error has occurred on NVM, XRAM. 1_B **Int**, Uncorrectable ECC error has occurred on NVM, RAM.
FNMIMAP	5	r	**NVM Map Error NMI Flag** This bit is set by hardware and can only be cleared by software. 0_B **no Int**, No NVM Map Error NMI has occurred. 1_B **Int**, NVM Map Error has occurred.

Field	Bits	Type	Description
FNMIOWD	4	r	**Oscillator Watchdog NMI Flag** This bit is set by hardware and can only be cleared by software. 0_B **no Int**, No oscillator watchdog NMI has occurred. 1_B **Int**, Oscillator watchdog event has occurred.
FNMIOT	3	r	**Overtemperature NMI Flag** This bit is set by hardware and can only be cleared by software. As this is a shared NMI source, this flag should be cleared after checking and clearing the corresponding event flags. 0_B **no Int**, No OT NMI has occurred. 1_B **Int**, OT NMI event has occurred.
RES	2	r	**Reserved** Returns 0 if read; should be written with 0.
FNMIPLL	1	r	**PLL NMI Flag** This bit is set by hardware and can only be cleared by software. 0_B **no Int**, No PLL NMI has occurred. 1_B **Int**, PLL loss-of-lock to the external crystal has occurred.
FNMIWDT	0	r	**Watchdog Timer NMI Flag** This bit is set by hardware and can only be cleared by software. As this is a shared NMI source, this flag should be cleared after checking and clearing the corresponding event flags. 0_B **no Int**, No watchdog NMI has occurred. 1_B **Int**, WDT prewarning has occurred.

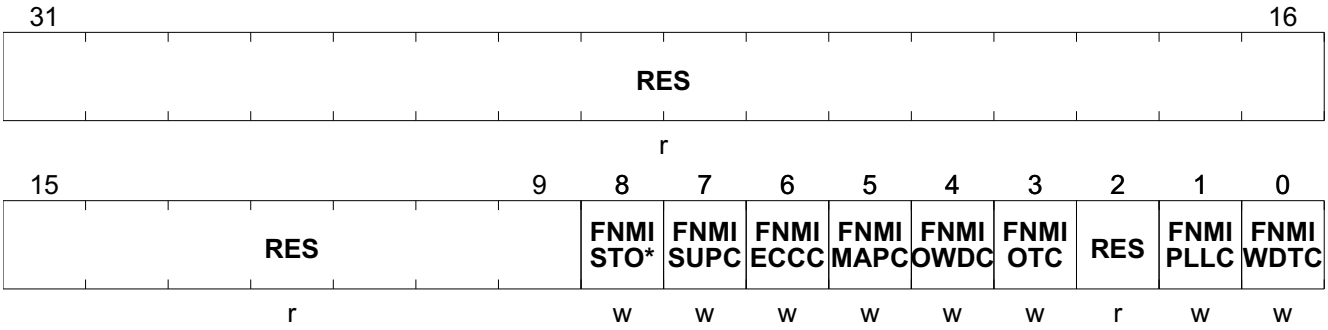
Table 200 RESET of SCU_NMISR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		

NMI Status Register

Each NMI event and status flag is retained across these resets: 1) WDT reset, 2) soft reset. These include all the flags of NMISR register: FNMIWDT, FNMIPLL, FNMIOT, FNMIOWD, FNMIMAP, and indirectly, FNMIECC and FNMISUP. In the case of NMIs with shared source i.e. watchdog, ECC or supply prewarning NMI, the respective indicator or event flags not located in NMISR are also retained. Refer to Chapter 1.6.5 for identifying the NMI event.

SCU_NMISRCLR	Offset	Reset Value
NMI Status Clear Register	000_H	see Table 201



Field	Bits	Type	Description
RES	31:9	r	**Reserved** Returns 0 if read; should be written with 0.
FNMISTOFC	8	w	**Stack Overflow NMI Flag** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
FNMISUPC	7	w	**Supply Prewarning NMI Flag** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
FNMIECCC	6	w	**ECC Error NMI Flag** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
FNMIMAPC	5	w	**NVM Map Error NMI Flag** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
FNMIOWDC	4	w	**Oscillator Watchdog NMI Flag** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
FNMIOTC	3	w	**Overtemperature NMI Flag** As this is a shared NMI source, this flag should be cleared after checking and clearing the corresponding event flags. 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
RES	2	r	**Reserved** Returns 0 if read; should be written with 0.

Field	Bits	Type	Description
FNMIPLLC	1	w	**PLL NMI Flag** 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared
FNMIWDTC	0	w	**Watchdog Timer NMI Flag** As this is a shared NMI source, this flag should be cleared after checking and clearing the corresponding event flags. 0_B **Not Cleared**, Interrupt event is not cleared. 1_B **Cleared**, Interrupt event is cleared

Table 201 RESET of SCU_NMISRCLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

12.9 Interrupt Priority Registers

Each interrupt node can be individually programmed to one of the 4 priority levels available. The user can set them in the corresponding **NVIC_IPRx** Register (see Core Chapter).

13 Math Divider Module

13.1 Features

The MATH Coprocessor includes the following features:

- Divide function with operand pre-processing and result post-processing
- AHB-Interface supports Byte/half word/ word Register access
- Supports fast execution kernel clock faster than interface clock

13.2 Introduction

The MATH Coprocessor (MATH) module supports the CPU in math-intensive computations with a Divider Unit (DIV) for signed and unsigned 32-bit division operations.

13.3 Block Diagram

Figure 76 shows a block diagram of the MATH Coprocessor.

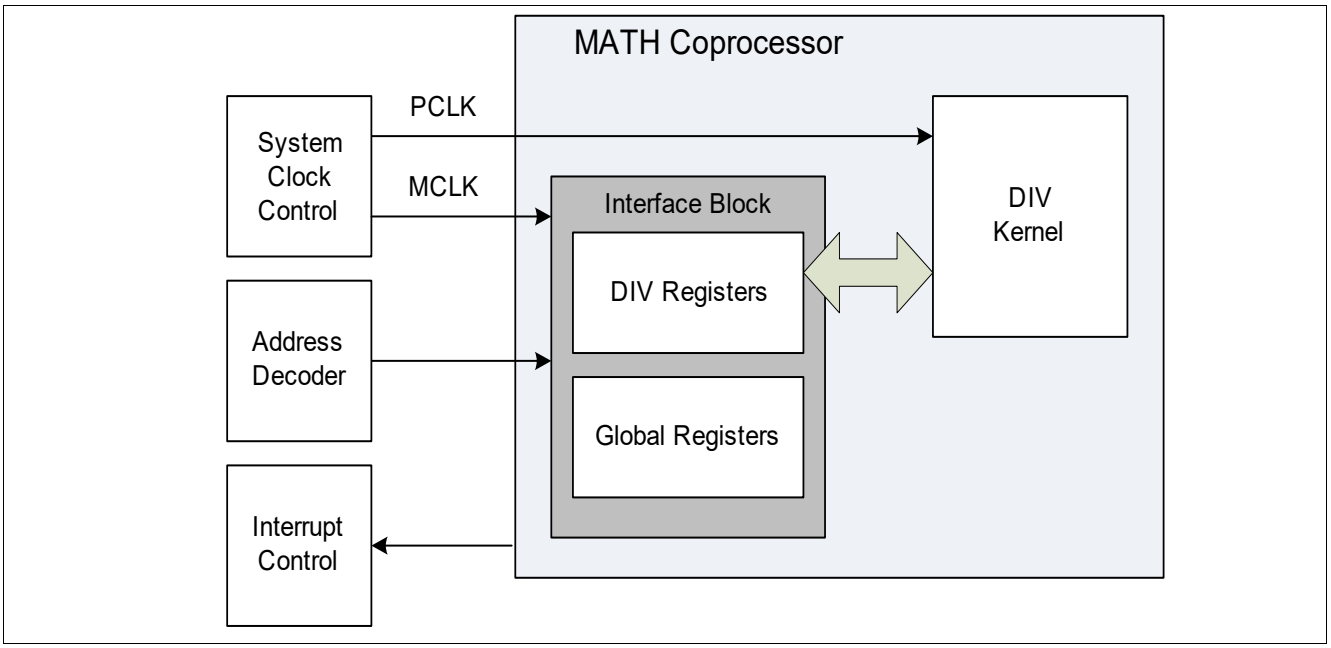


Figure 76 MATH Coprocessor Block Diagram

13.4 Divider Unit (DIV)

13.4.1 Features

The DIV supports the following features:

- Signed/unsigned 32-bit division operation in 35 kernel clock cycles
- Three division modes:
  - 32-bit divide by 32-bit
  - 32-bit divide by 16-bit
  - 16-bit divide by 16-bit
- Operands pre-processing with configurable number of:
  - Left shifts for dividend
  - Right shifts for divisor
- Result post-processing with configurable number of shifts and shift direction

Note: The execution time of 35 kernel clock cycles for a division operation does not include the time to access the operand and result registers, which can take up a large part of the time for a division function in the application software.

13.4.2 Division Operation

The DIV supports the truncated division operation, which is also the ISO C99 standard and the popular choice among modern processors. The division and modulus functions? of the truncated division are related in the following way:

If q = D div d

and r = D mod d

then D = q * d + r

and | r | < | d |

where “D” is the dividend, “d” is the divisor, “q” is the quotient and “r” is the remainder.

The truncated division rounds the quotient towards zero and the sign of its remainder is always the same as that of its dividend, i.e., sign (r) = sign (D).

To execute a divider operation with the DIV, it is first required to configure the division as follows:

- Signed or unsigned through the DIVCON.USIGN bit
- Division mode through the DIVCON.DIVMODE bits
- Start mode through the DIVCON.STMODE bit

The dividend and divisor values are then written into the DVD and DVS registers. The division is started with the write to DVS register or by setting the start bit DIVCON.ST, depending on the start mode. If the ST bit is used, the bit is automatically cleared in the next kernel clock cycle. The start of the division operation sets the busy flag, DIVST.BSY.

The division operation always takes 35 kernel clock cycles, and upon completion, the quotient and remainder values will be available at the QUOT and RMD registers, and the BSY flag will be cleared. The end of calculation event sets the Divider event flag EVFR.DIVEOC and can trigger an interrupt request to NVIC if enabled through the EVIER.DIVEOCIEN bit. The flag is cleared only by a software write to the EVFCR.DIVEOCC bit.?

Figure 77 shows the timing diagram for a division operation.

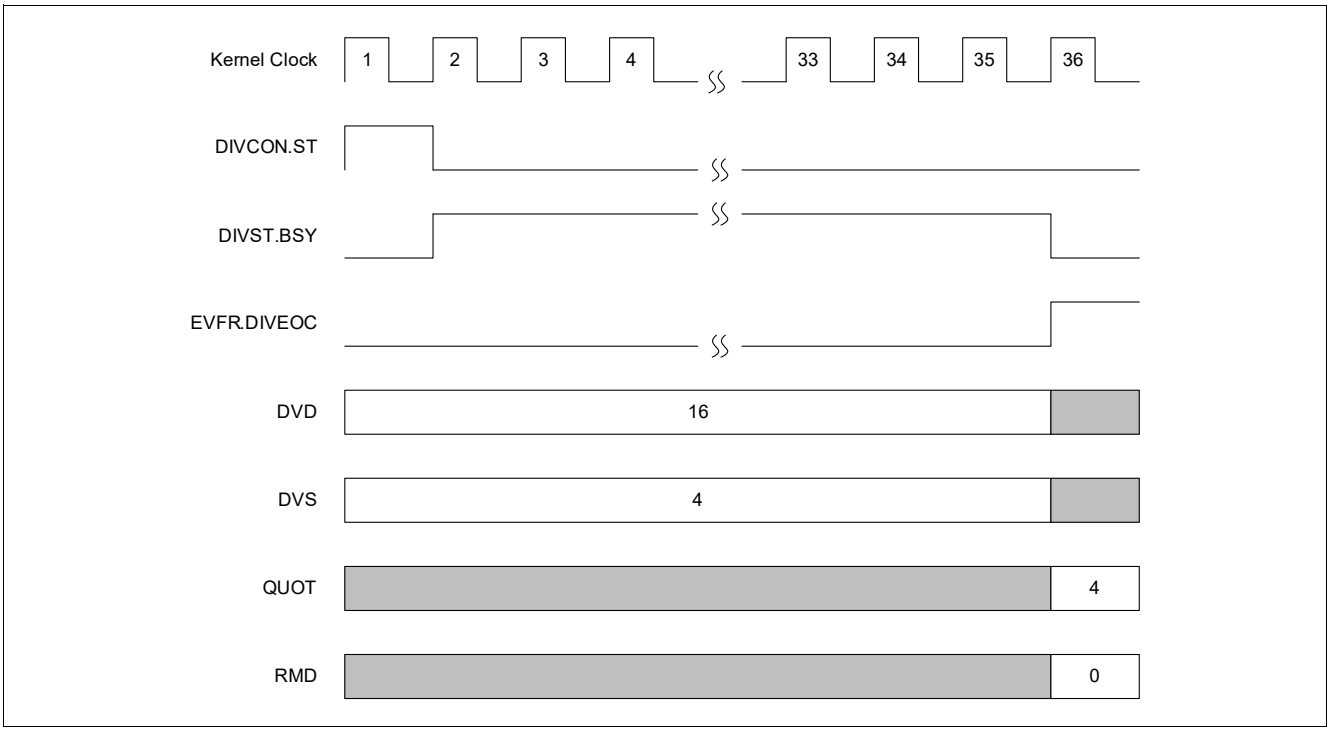


Figure 77 Timing Diagram for a Division Operation

Note: Reading the QUOT and RMD registers while BSY=1 will cause the DIV to insert wait states onto the bus until the active calculation is completed (BSY=0). This ensures that any read access on the result registers QUOT or RMD returns a valid result. However, the interrupt latency will be increased as the bus may be locked up for a number of kernel clock cycles.

13.4.2.1 Start Mode Selection

The condition to start a division operation is selectable through the DIVCON.STMODE bit:

- When STMODE = 0, a division operation is started with a write to the DVS register. In this case, no further software set of the ST bit is necessary.
- When STMODE = 1, a division is started by setting the ST bit.

For both start modes, it must be ensured that the DIV is not performing any active calculation and the DIVST.BSY flag is 0 before starting the operation, else the start request is discarded though DVS will still be updated with the write value. Write access to DIVCON register is ignored while BSY = 1. It is recommended for the application to poll for this condition before starting the divider operation with a write to ST bit or DVS register.

13.4.2.2 Error Handling

The DIV supports two types of error detection:

- Divide by zero error
- Overflow error

In both cases, the error will be indicated by the EVFR.DIVERR flag. An interrupt request to NVIC can be generated if it is enabled through EVIER.DIVERRIEN.

The DIV supports the detection of a divide by zero error. The error will be indicated by the EVFR.DIVERR flag. An interrupt request to NVIC can be generated if it is enabled through EVIER.DIVERRIEN.

The division operation will still proceed as normal and complete in 35 kernel clock cycles. The error flag becomes set at the same clock cycle as DIVEOC.

Divide by Zero Error

A divide by zero error occurs when a division operation is started with the divisor value in DVS register equal to 0.

An example of a divide by zero error is shown in **Figure 78**.

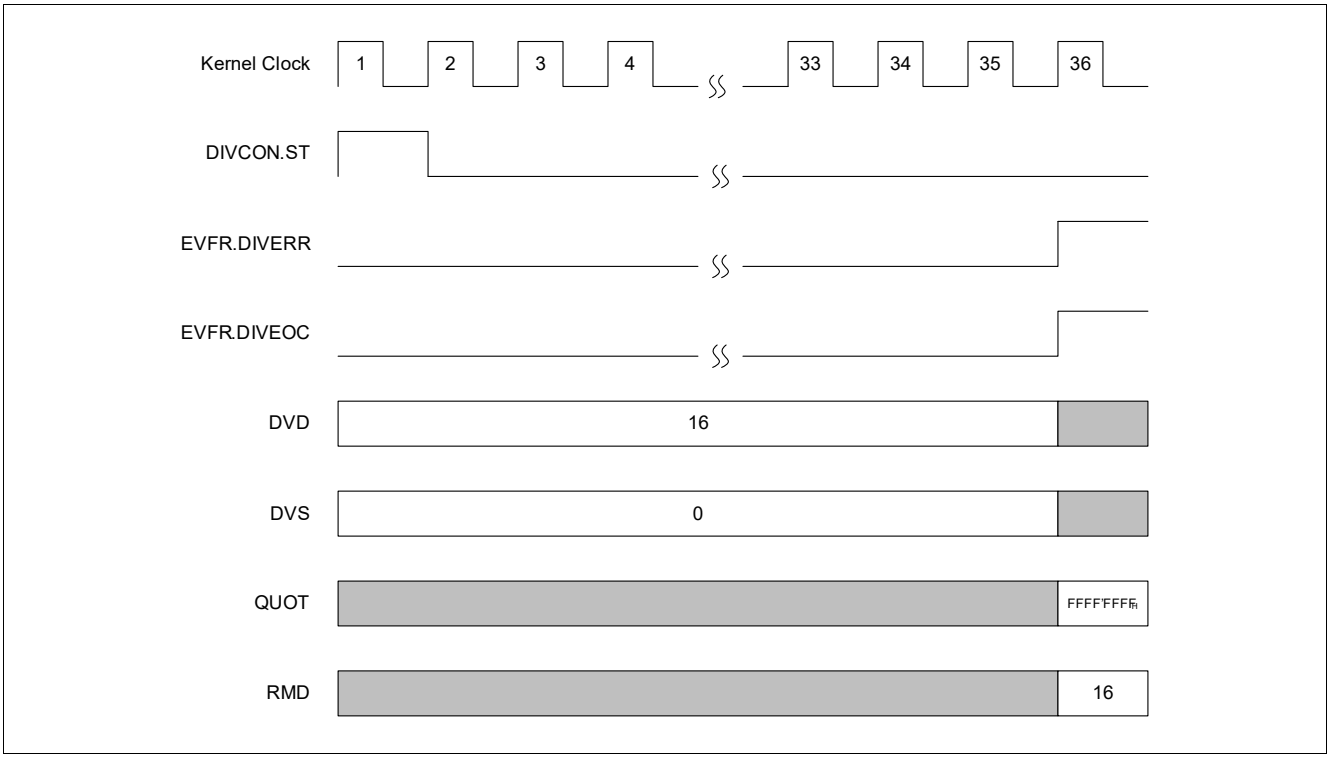


Figure 78 Timing Diagram with Divide by Zero Error

If DIVCON.USIGN is 0 (signed operation) and DVD is a negative value, the QUOT register will contain the maximum negative representation. E.g. 0x80000000 for 32 bit operation. In the all other cases the QOUT register contains the maximum positive number. E.g.: 0xFFFFFFFF for 32 bit unsigned, 0x7FFFFFFF for 32 bit signed operation.

For 16 bit operations this means that the uper 16 bits of the QUOT register are also filled with 0x0000 for unsigned operations.

For signed operations the sign bit will be extended to the upper 16 bits of the QUOT register. Please refer to **Table 202**

Table 202 QUOT/RMD Result Register content in divide by zero or overflow condition

Operating Mode	DVD positive	DVD negative	DVD positive	DVD negative
	QUOT (Hex)		RMD (Hex)	
32 bit signed	$7FFF'FFFF_H$	$8000'0000_H$	$0000'0000_H$	
32 bit unsigned	$FFFF'FFFF_H$	n.a	$0000'0000_H$	
16 bit signed$_B$	$0000'7FFF_H$	$FFFF'8000_H$	$0000'0000_H$	
16 bit unsigned$_B$	$0000'FFFF_H$	n.a	$0000'0000_H$	

Table 202 QUOT/RMD Result Register content in divide by zero or overflow condition (cont'd)

Operating Mode	DVD positive	DVD negative	DVD positive	DVD negative
	QUOT (Hex)		RMD (Hex)	
32/16 bit signed$_B$	7FFF'FFFF$_H$	8000'0000$_H$	0000'0000$_H$	
32/16 bit unsigned$_B$	FFFF'FFFF$_H$	n.a.	0000'0000$_H$	

The flag is cleared only by a software write to the EVFCR.DIVERRC bit.

*Note: If result post-processing (see **Section 13.4.3**) is enabled and an overflow condition occurs, no result post-processing is executed*

Overflow Error

An overflow error occurs when one of the three conditions listed in **Table 203** is detected.

Table 203 Overflow Error Conditions

DIVCON.DIVMODE	Dividend (Hex)	Dividend (Dec)	Divisor (Hex)	Divisor (Dec)
00$_B$	8000'0000$_H$	-2^{31}	FFFF'FFFF$_H$	-1
01$_B$	8000'0000$_H$	-2^{31}	FFFF$_H$	-1
10$_B$	8000$_H$	-32768	FFFF$_H$	-1

Also a post processing left shift operation can cause overflow errors. Following table shows the overflow error conditions depending on the operating mode.

Table 204 Overflow Error Conditions during left shift post processing

Operating Mode	Overflow Error condition
32 bit signed	'1' shifted above bit 30
32 bit unsigned	'1' shifted above bit 31
16 bit signed	'1' shifted above bit 14
16 bit unsigned	'1' shifted above bit 15
32/16 bit signed	'1' shifted above bit 30
32/16 bit unsigned	'1' shifted above bit 31

The overflow signalization is implemented on the result of a division/post processing only. This means overflows possibly caused by an pre processing are not signalized.

In case of an overflow condition the content of the result register QUOT/RMD are according to **Table 202**

13.4.3 Operand/Result Pre-/Post-Processing

The DIV supports operand pre-processing and result post-processing by allowing the following:

- Left shift of the dividend value before the start of division (**MATH_DVD**.VAL)
- Right shift of the divisor value before the start of division (**MATH_DVS**.VAL)
- Left or right shift of the quotient value after the end of division (**MATH_QUOT**.VAL)

The number of shifts is determined by the respective 5-bit shift count bit fields in DIVCON register. Additionally for the quotient, the shift direction is defined by the bit DIVCON.QSDIR.

All shifts are arithmetic shifts. This means if shift left, zeros will be inserted at LSB, while if shift right, zeros (in unsigned mode) or the signed bit (in signed mode) will be inserted at MSB.

A left shift in signed mode does not influence the sign bit. Depending on the operating mode 16/32 bit calculation the position of the sign bit is changing and with this the bit numbers influenced by a left shift operation.

Table 205 result bits influenced by left shift depending on operation mode

Operation Mode	Sign bit	bits influenced by left shift
32 bit signed	31	30:0
32/16 bit signed	31	30:0
16 bit signed	15	14:0
32 bit unsigned	-	31:0
32/16 bit unsigned	-	31:0
16 bit unsigned	-	15:0

If selected to be enabled by writing a non-zero value to the shift count bit fields, the shift operations will be triggered at the start and end of the division operation and do not consume any additional kernel clock cycles. The DIV event flag will be generated at the end of the 35-clock execution cycle.

During operand pre-processing, the shifts are performed only on the output of the DVD and DVS registers and not on the registers themselves. Therefore, the contents of these registers remain unchanged.

13.5 Global Functions

13.5.1 Result Chaining

The MATH Coprocessor supports result chaining between the result and the oparand registers of the DIV Unit.

For the DIV, this means that each of the operand registers, DVD and DVS, can be updated with the value from any one of the result registers (QUOT and RMD).

The selection is done with the operand register result chaining bit fields (xRC) in GLBCON register.

The update is done once the preceding DIV operation is completed, together with the update to the actual result register. **Figure 79** shows an example of chaining the DIV quotient result to the DVD.

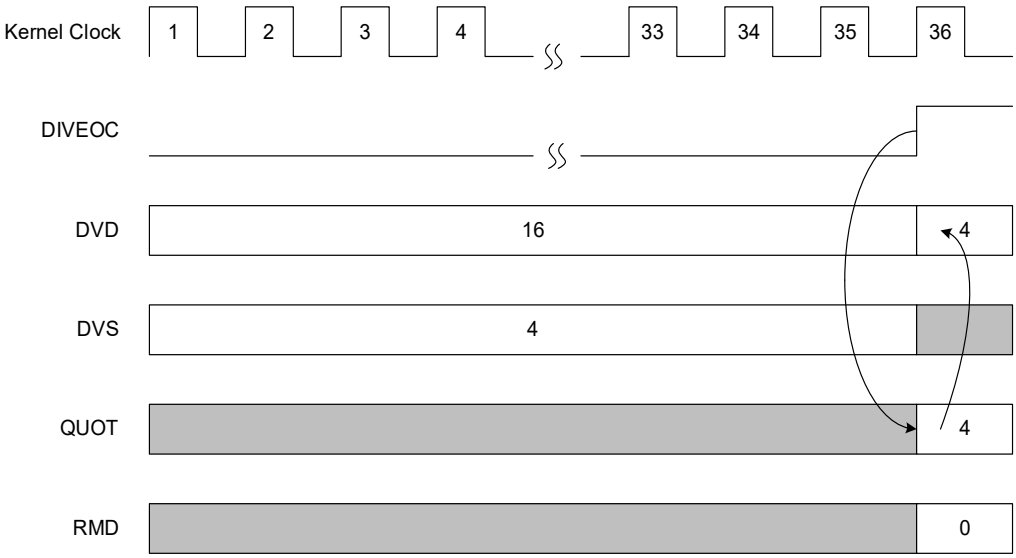


Figure 79 Chained Division Operation

13.5.1.1 Result Chaining when Start Mode = 0

When the respective start mode is 0, a DIV operation can be started by having the application software write to the DVS operand registers (see **Section 13.4.2.1**). An update of these registers due to result chaining is equivalent to a hardware write and therefore, has the same effect.

The possible deadlock situation with START Mode = 0 and DVSRC unequal to NONE (result chaining to DVS Register) will be blocked by hardware. In this case no result chainig is active

13.5.1.2 Handling Busy Flags when Result Chaining is Enabled

While the busy flag is set:

- Reading the DIV result registers will cause wait states to be inserted onto the bus.
- While the DIV is active, starting a new DIV calculation or writing to their respective control register **MATH_DIVCON** will have no effect.
- However, if the DIV is still inactive or has just returned from an active state, starting a new DIV calculation or writing to their respective control registers will take effect.
- Writing to **MATH_DVD**, **MATH_DVS** is accepted but does not influence the result registers (**MATH_QUOT**, **MATH_RMD**) of the running calculation. Written values will be effective with a start of a new calculation.

13.6 Service Request Generation

If enabled by the respective interrupt enable bits in EVIER register, the DIV error and end of calculation events will trigger the interrupt service request to NVIC. The event is indicated by the event flag in EVFR register. The event flag can be cleared only by writing a 1 to the event flag clear bit in EVFCR register.

Writing a 1 to the event flag set bit in EVFSR register has the same effect of an end of calculation event.

13.7 Debug Behaviour

The MATH_DIV can be configured to enter a suspend mode when the program execution of the CPU is halted by the debugger (indicated by the assertion of the suspend signal).

Two suspend modes are supported and can be selected through the control bit field GLBCON.SUSCFG:

- Hard Suspend Mode
  - The kernel clock is immediately switched off, thereby stopping all calculations
- Soft Suspend Mode
  - Any active calculation is allowed to continue and only after it is completed, will the kernel clock be switched off

After the kernel clock is switched off, all registers become read-only. Writing to registers in this state has no effect. Suspend mode is exited and normal operations resumed when the suspend signal becomes deasserted.

The suspend mode is non-intrusive concerning the register bits. This means register bits are not modified by hardware when entering or leaving the suspend mode.

A write access to registers during suspend mode is not signalized as an AHB error. In this case hresp = “00”

13.8 Enable/ Disable Behaviour

The MATH_DIV supports also a power saving mode which can be entered by using **MATH_GLBCON**.MATH_EN. In this mode the internal module clock will be gated off. Access to the Register interface is still possible but several boundary conditions should be respected:

- Read/Write of **MATH_GLBCON**, **MATH_DVD**, **MATH_DVS**,**MATH_DIVCON**
  - Read/Write is possible but no calculation can be triggered.
- Read/Write of **MATH_EVIER**, **MATH_EVFR**, **MATH_EVSFR**, **MATH_EVFCR**
  - Read/Write is possible. Interupts can be set/cleared/enabled/disabled.
  - Setting the an interupt bit will trigger the corresponding Interupt node also during MATH_EN = 0
  - Since the divider kernel is gated off no interupt setting by hardware is possible.
- Disabling the MATH_DIV module during **MATH_DIVST**.BSY = ‘1’ will finish the running the calculation and gates off the kernel clock after **MATH_DIVST**.BSY = ‘0’ (same behaviour as soft suspend mode).

13.9 Power, Reset and Clock

The MATH Coprocessor is located in the core power domain. The module, including all registers, will be reset to its default state by a system reset.

Math Divider Module

The MATH Coprocessor requires two input clock signals, one for the kernel clock and one for the interface clock.

The GLBCON.MATH_EN will globaly enable the DIV_UNIT. In case the Math module is not enable (GLBCON.MATH_EN = 0) the kernel clock will be gated off. The interface clock is untouched by this setting.

13.10 Register Description

Table 206 Register Address Space

Module	Base Address	End Address	Note
MATH	4801 3000$_H$	4801 3FFF$_H$	Math

Table 207 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Math Module Registers, Global Registers			
MATH_GLBCON	Global Control Register	04$_H$	0000 0000$_H$
MATH_MATH_ID	Module Identification Register	08$_H$	00F2 C0XX$_H$
MATH_EVIER	Event Interupt Enable Register	0C$_H$	0000 0000$_H$
MATH_EVFR	Event Flag Register	10$_H$	0000 0000$_H$
MATH_EVSFR	Event Flag Set Register	14$_H$	0000 0000$_H$
MATH_EVFCR	Event Flag Clear Register	18$_H$	0000 0000$_H$
Math Module Registers, Divider Registers			
MATH_DVD	Dividend Register	20$_H$	0000 0000$_H$
MATH_DVS	Divisor Register	24$_H$	0000 0000$_H$
MATH_QUOT	Quotient Register	28$_H$	0000 0000$_H$
MATH_RMD	Remainder Register	2C$_H$	0000 0000$_H$
MATH_DIVST	Divider Status Register	30$_H$	0000 0000$_H$
MATH_DIVCON	Divider Control Register	34$_H$	0000 0000$_H$

The registers are addressed bytewise.

AHB errors will be generated in case of:

- Read/Write access inside the module address range to none used Register addresses
- Write access to read only Register where all register fields are read only

No AHB Error will be generated in case of:

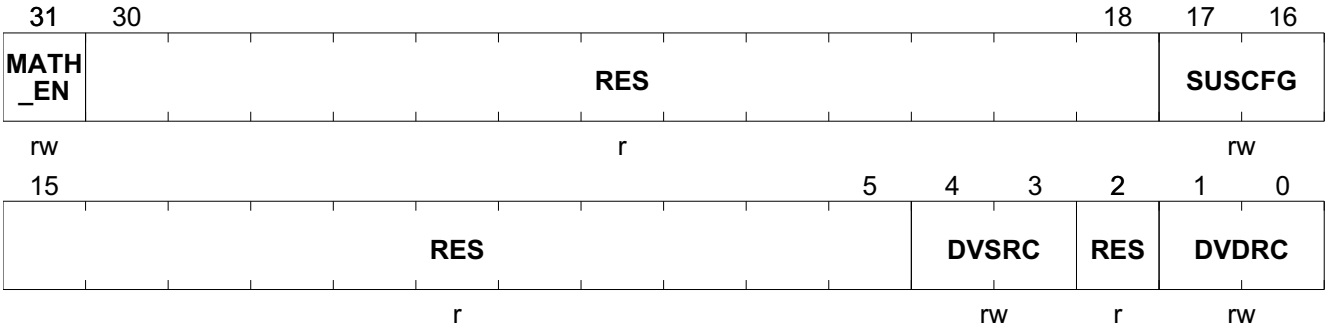
- Read access to write only Registers

13.10.1 Math Module Registers

Math Divider Module

Global Control Register

MATH_GLBCON	Offset	Reset Value
Global Control Register	04_H	see Table 208



Field	Bits	Type	Description
MATH_EN	31	rw	**Enable Math Module** *Note: This bit is RESET_TYPE_3* 1_B **Enable**, Math module is enabled 0_B **Disable**, Math module is disabled
RES	30:18	r	**Reserved** Returns 0 if read; should be written with 0.
SUSCFG	17:16	rw	**Suspend Mode Configuration** This bit determines if a suspend mode is entered by the MATH Coprocessor when the CPU is halted. *Note: This field is RESET_TYPE_4* 00_B **no suspend**, Suspend mode is never entered. 01_B **hard suspend**, Hard suspend mode will be entered when CPU is halted. 10_B **soft suspend**, Soft suspend mode will be entered when CPU is halted. 11_B **reserved**, Reserved (Suspend mode is never entered)
RES	15:5	r	**Reserved** Returns 0 if read; should be written with 0.

Field	Bits	Type	Description
DVSRC	4:3	rw	**Divisor Register Result Chaining** The DVS register in DIV will be updated with the selected result register value when the result chaining trigger event occurs. *Note: This field is RESET_TYPE_3* 00_B **disabled**, No result chaining is selected 01_B **QUOT**, QUOT register is the selected source 10_B **RMD**, RMD register is the selected source 11_B **reserved**, Reserved (no result chaining is selected)
RES	2	r	**Reserved** Returns 0 if read; should be written with 0.
DVDRC	1:0	rw	**Dividend Register Result Chaining** The DVD register in DIV will be updated with the selected result register value when the result chaining trigger event occurs. *Note: This field is RESET_TYPE_3* 00_B **disabled**, No result chaining is selected 01_B **QUOT**, QUOT register is the selected source 10_B **RMD**, RMD register is the selected source 11_B **reserved**, Reserved (no result chaining is selected)

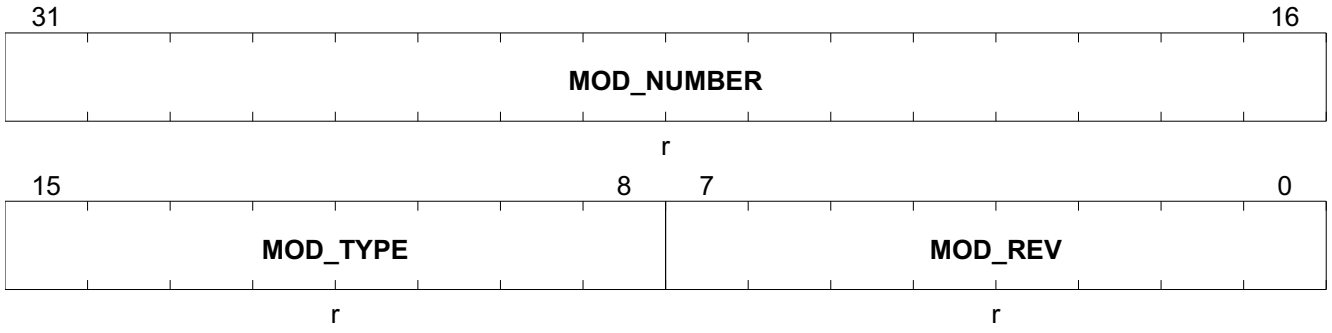
Table 208 Reset of MATH_GLBCON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		

Math Divider Module

Module Identification Register

MATH_MATH_ID	Offset	Reset Value
Module Identification Register	08_H	see Table 209



Field	Bits	Type	Description
MOD_NUMBER	31:16	r	**Module Number Value** This bit field defines the module identification number.
MOD_TYPE	15:8	r	**Module Type** This bit field is $C0_H$. It defines the module as a 32-bit module.
MOD_REV	7:0	r	**Module Revision Number** MOD_REV defines the revision number. The value of a module revision starts with 01_H (first revision).

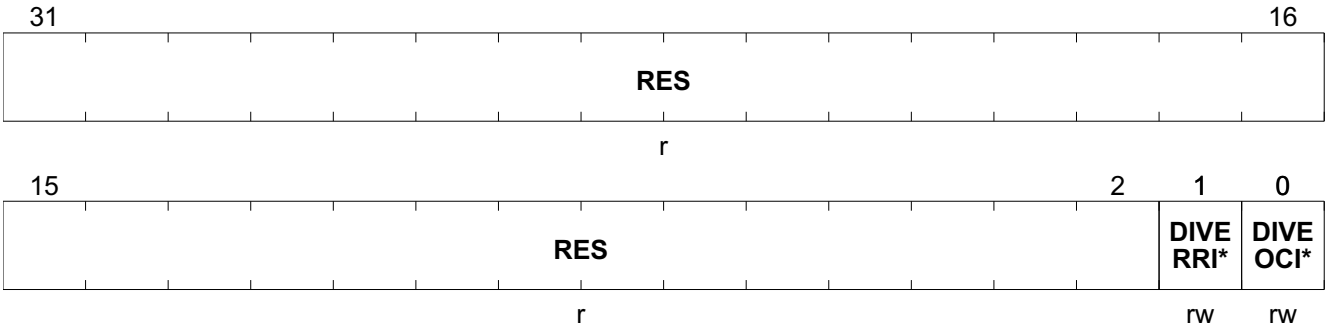
Table 209 Reset of MATH_MATH_ID

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$00F2C0XX_H$	RESET_TYPE_3		

Math Divider Module

Event Interupt EnableRegister

MATH_EVIER	Offset	Reset Value
Event Interupt Enable Register	$0C_H$	see Table 210



Field	Bits	Type	Description
RES	31:2	r	**Reserved** Returns 0 if read; should be written with 0.
DIVERRIEN	1	rw	**Divider Error Interrupt Enable** 0_B **Disable**, Divider error interrupt generation is disabled 1_B **Enable**, Divider error interrupt generation is enabled
DIVEOCIEN	0	rw	**Divider End of Calculation Interrupt Enable** 0_B **Disable**, Divider end of calculation interrupt generation is disabled. 1_B **Enable**, Divider end of calculation interrupt generation is enabled.

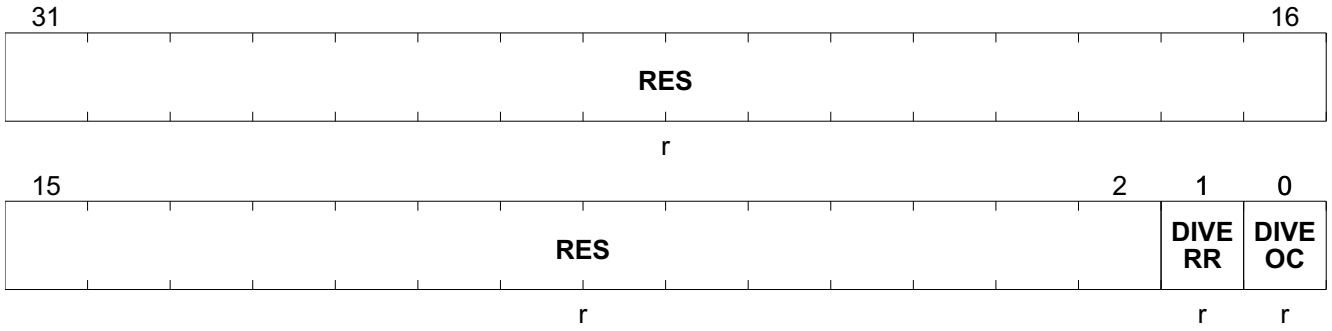
Table 210 Reset of MATH_EVIER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Math Divider Module

Event Flag Register

MATH_EVFR	Offset	Reset Value
Event Flag Register	10_H	see Table 211



Field	Bits	Type	Description
RES	31:2	r	**Reserved** Returns 0 if read; should be written with 0.
DIVERR	1	r	**Divider Error Event Flag** 0_B **no Error**, Divider error event has not been detected 1_B **Error**, Divider error event has been detected
DIVEOC	0	r	**Divider End of Calculation Event Flag** 0_B **no EOC**, Divider end of calculation event has not been detected. 1_B **EOC**, Divider end of calculation event has been detected.

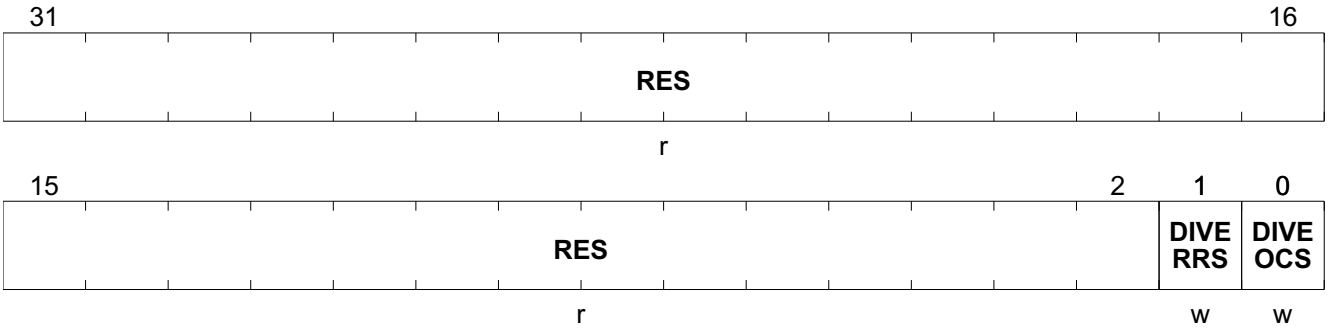
Table 211 Reset of MATH_EVFR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Math Divider Module

Event Flag Set Register

MATH_EVSFR	Offset	Reset Value
Event Flag Set Register	14_H	see Table 212



Field	Bits	Type	Description
RES	31:2	r	**Reserved** Returns 0 if read; should be written with 0.
DIVERRS	1	w	**Divider Error Event Flag Set** 0_B **no effect**, No effect. 1_B **Set**, Sets the Divider error event flag in EVFR register. Interrupt will be generated if enabled in EVIER register.
DIVEOCS	0	w	**Divider End of Calculation Event Flag Set** 0_B **no effect**, No effect. 1_B **Set**, Sets the Divider end of calculation event flag in EVFR register. Interrupt will be generated if enabled in EVIER register.

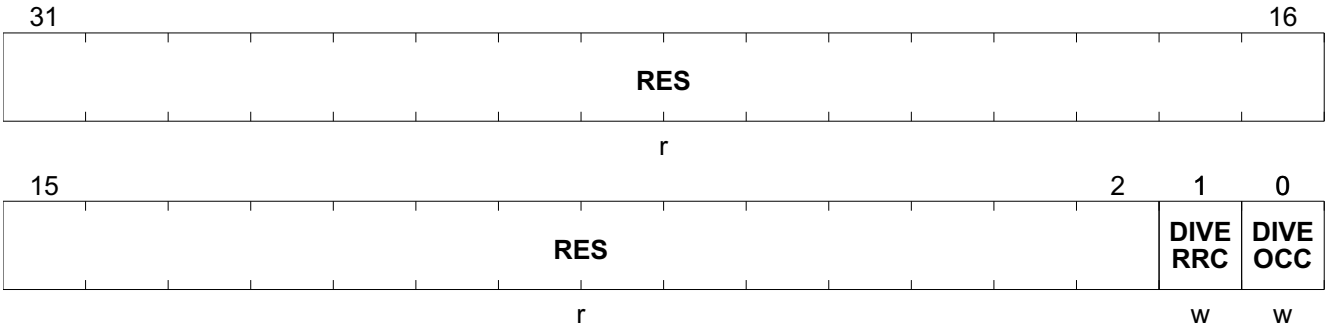
Table 212 Reset of MATH_EVSFR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Math Divider Module

Event Flag Clear Register

MATH_EVFCR	Offset	Reset Value
Event Flag Clear Register	18_H	see Table 213



Field	Bits	Type	Description
RES	31:2	r	**Reserved** Returns 0 if read; should be written with 0.
DIVERRC	1	w	**Divider Error Event Flag Clear** 0_B **no effect**, No effect. 1_B **clear**, Clears the Divider error event flag in EVFR register.
DIVEOCC	0	w	**Divider End of Calculation Event Flag Clear** 0_B **no effect**, No effect. 1_B **clear**, Clears the Divider end of calculation event flag in EVFR register.

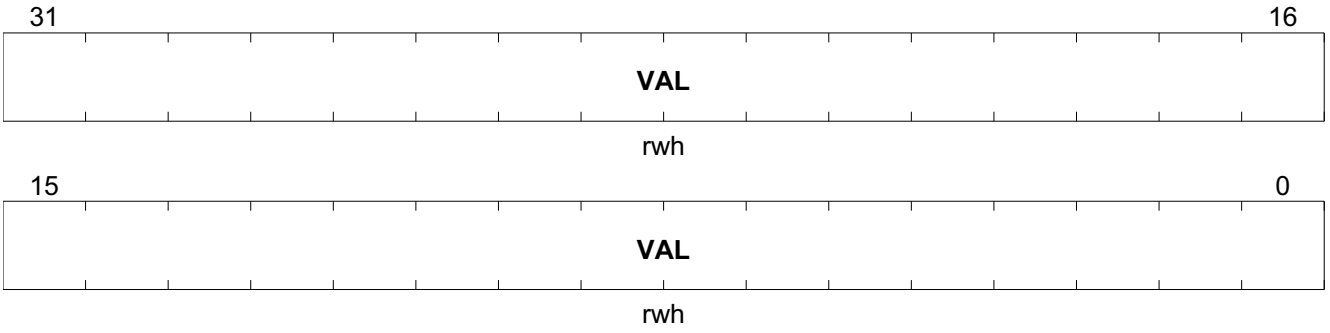
Table 213 Reset of MATH_EVFCR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Math Divider Module

Dividend Register

MATH_DVD	Offset	Reset Value
Dividend Register	20_H	see Table 214



Field	Bits	Type	Description
VAL	31:0	rwh	Dividend Value

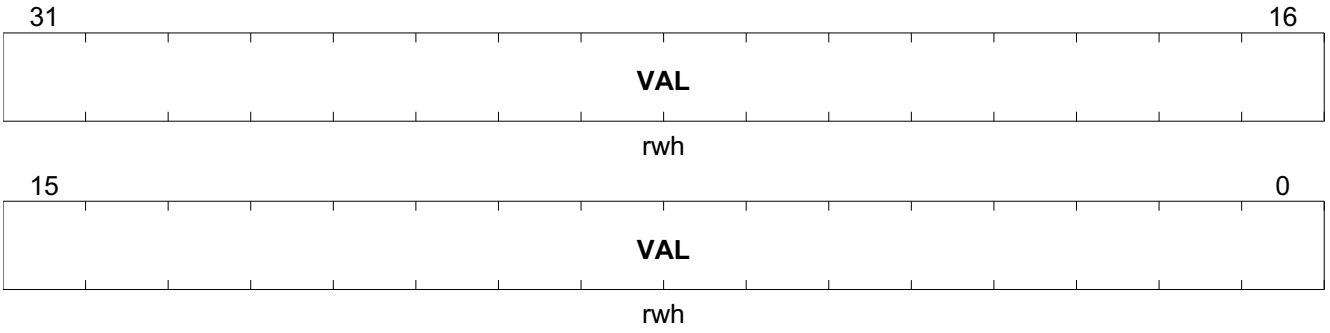
Table 214 Reset of MATH_DVD

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Math Divider Module

Divisor Register

MATH_DVS	Offset	Reset Value
Divisor Register	24_H	see Table 215



Field	Bits	Type	Description
VAL	31:0	rwh	**Divisor Value**

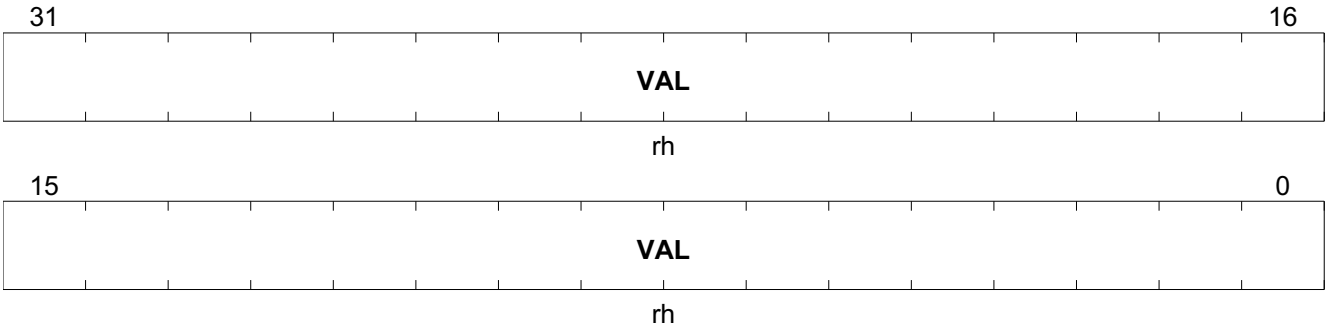
Table 215 Reset of MATH_DVS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Math Divider Module

Quotient Register

MATH_QUOT	Offset	Reset Value
Quotient Register	28_H	see Table 216



Field	Bits	Type	Description
VAL	31:0	rh	Quotient Value

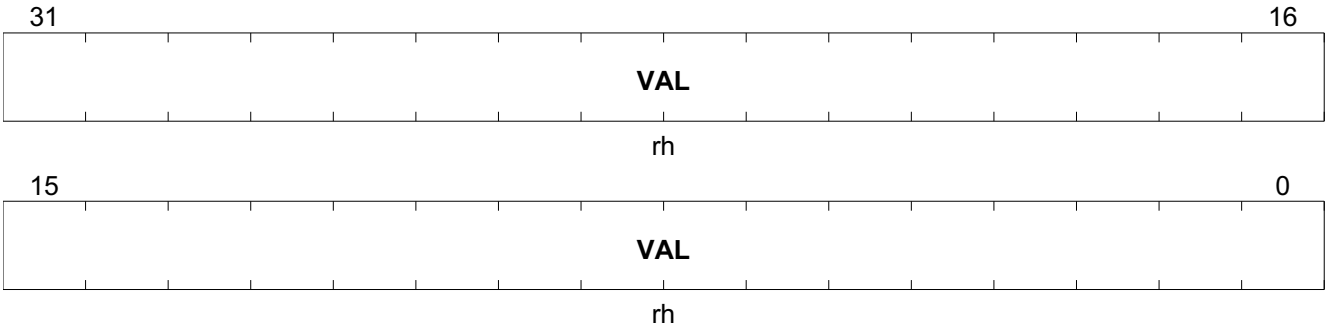
Table 216 Reset of MATH_QUOT

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Math Divider Module

Remainder Register

MATH_RMD	Offset	Reset Value
Remainder Register	$2C_H$	see Table 217



Field	Bits	Type	Description
VAL	31:0	rh	**Remainder Value**

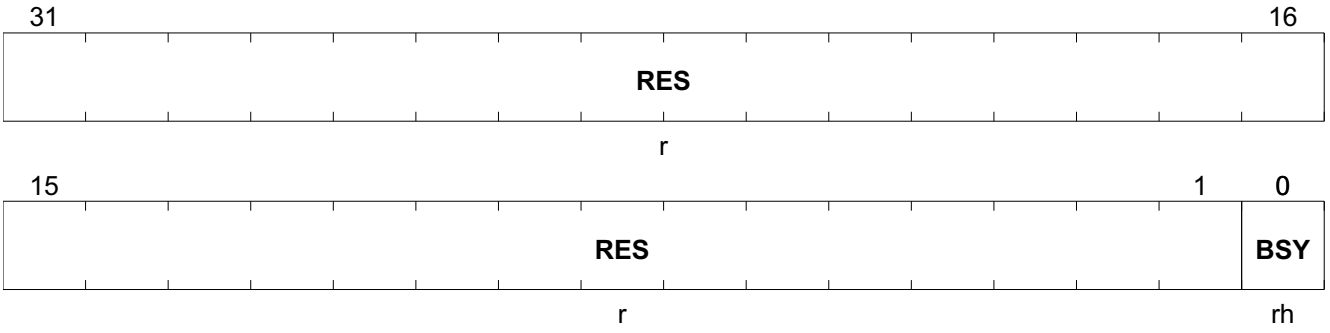
Table 217 Reset of MATH_RMD

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Math Divider Module

Divider Status Register

MATH_DIVST	Offset	Reset Value
Divider Status Register	30_H	see Table 218



Field	Bits	Type	Description
RES	31:1	r	**Reserved** Returns 0 if read; should be written with 0.
BSY	0	rh	**Busy Indication** 0_B **finish**, Divider is not running any division operation. 1_B **busy**, Divider is still running a division operation.

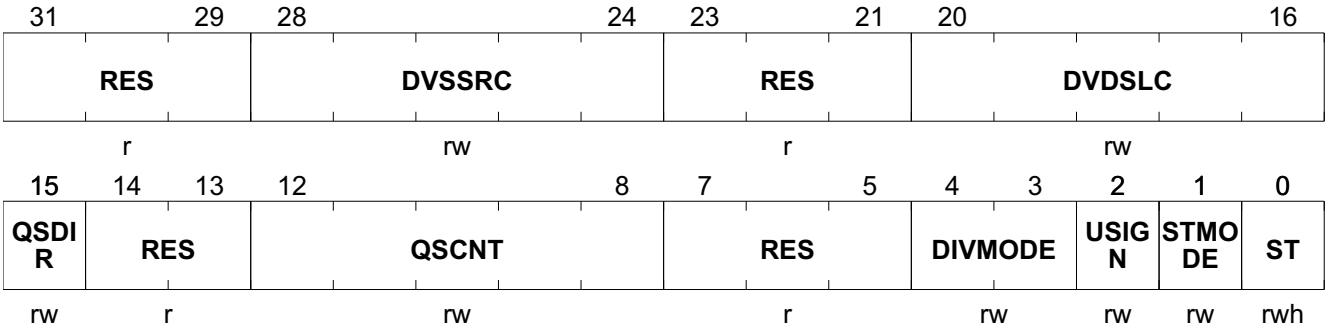
Table 218 Reset of MATH_DIVST

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Math Divider Module

Divider Control Register

MATH_DIVCON	Offset	Reset Value
Divider Control Register	34_H	see Table 219



Field	Bits	Type	Description
RES	31:29	r	**Reserved** Returns 0 if read; should be written with 0.
DVSSRC	28:24	rw	**Divisor Shift Right Count** If DVSSRC is not equal to 0, it indicates the number of bits the divisor will be shifted right by, prior to the division. If DVSSRC=0, no shift operation will take place.
RES	23:21	r	**Reserved** Returns 0 if read; should be written with 0.
DVDSLC	20:16	rw	**Dividend Shift Left Count** If DVDSLC is not equal to 0, it indicates the number of bits the dividend will be shifted left by, prior to the division. If DVDSLC=0, no shift operation will take place.
QSDIR	15	rw	**Quotient Shift Direction** This bit is used to select the shift direction for the quotient after a division: 0_B **Left shift**, Left shift 1_B **Right Shift**, Right shift
RES	14:13	r	**Reserved** Returns 0 if read; should be written with 0.
QSCNT	12:8	rw	**Quotient Shift Count** If QSCNT is not equal to 0, it indicates the number of bits the quotient will be shifted by, after the division. If QSCNT=0, no shift operation will take place.
RES	7:5	r	**Reserved** Returns 0 if read; should be written with 0.
DIVMODE	4:3	rw	**Division Mode** 00_B **32-32**, 32-bit divide by 32-bit 01_B **32-16**, 32-bit divide by 16-bit 10_B **16-16**, 16-bit divide by 16-bit 11_B **reserved**, Reserved (32-bit divide by 32-bit)

Field	Bits	Type	Description
USIGN	2	rw	**Unsigned Division Enable** 0_B **signed**, Signed division is selected 1_B **unsigned**, Unsigned division is selected
STMODE	1	rw	**Start Mode** Selects the start mode for the division operation: 0_B **Auto**, Calculation is automatically started with a write to DVS register 1_B **Manual**, Calculation is started by setting the ST bit to 1 *Note: The start request for a new division operation will be ignored if BSY = 1.*
ST	0	rwh	**Start Bit** 0_B **no effect**, No effect 1_B **Start**, Start the division operation when STMODE=1_B **The bit is automatically cleared by hardware after one kernel clock cycle.**

Table 219 Reset of MATH_DIVCON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

14 Watchdog Timer (WDT1)

14.1 Features

There are two watchdog timers in the system. The Watchdog Timer (WDT) within the microcontroller and the Watchdog Timer (WDT1), which is described in this section.

In Active Mode, the WDT1 acts as a windowed watchdog timer, which provides a highly reliable and safe way to recover from software or hardware failures.

The WDT1 is always enabled in Active Mode. In Sleep Mode, Stop Mode and Debug Mode the WDT1 is disabled.

Functional Features

- Watchdog Timer is operating with a from the system clock (f_{SYS}) independent clock source (f_{LP_CLK})
- Windowed Watchdog Timer with programmable timing (16, 32, 48, …, 1008ms period) in Active Mode
- Long open window (200 ms) after power-up, reset, wake-up
- Short open window (30 ms) to facilitate Flash programming
- System safety shutdown to Sleep Mode after 5 missed WDT1 services
- Watchdog is disabled in Debug Mode
- Watchdog cannot be deactivated in Normal Mode
- Watchdog reset is stored in reset status register **PMU_RESET_STS**

14.2 Introduction

The behavior of the Watchdog Timer in Active Mode is depicted in **Figure 80**.

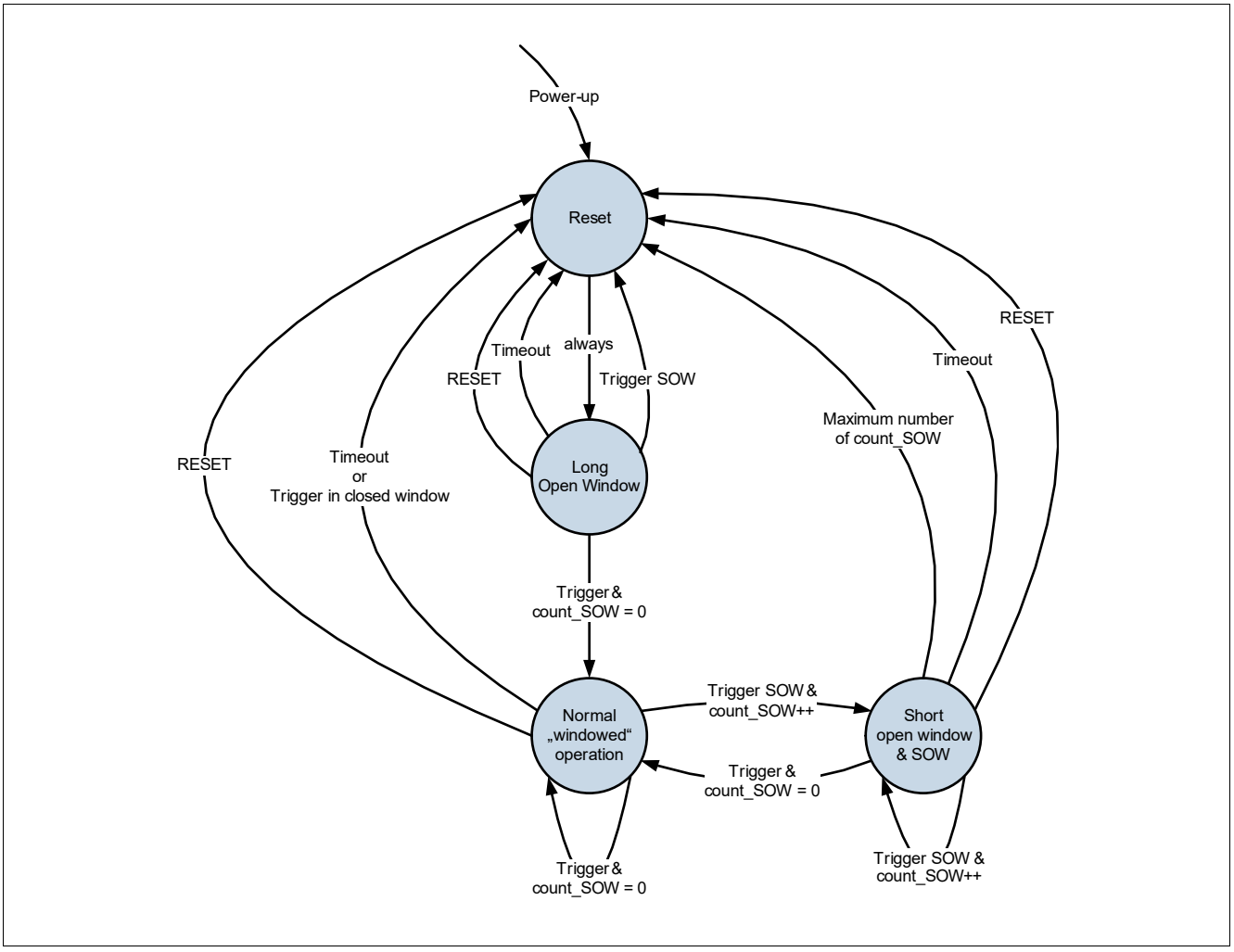


Figure 80 Watchdog Timer Behavior

14.3 Functional Description

14.3.1 Modes of Operation

The mode transition from the low power modes (WDT1 off) to active (WDT1 on) automatically initializes WDT1 to start in long open window mode.

14.3.2 Normal Operation

The software has to trigger the watchdog by writing to the WDT1_TRIG register. By triggering the watchdog also the length of the next watchdog period is selected inherently. The next period starts immediately with the trigger.

After Reset the WDT1 is starting with a long open window. The WDT1 has to be triggered within this long open window otherwise a reset will be generated at the end of the long open window. If the watchdog is not served properly consecutively 5 times, the system will enter sleep mode. After an initial successful trigger the WDT1 operates in a window watchdog mode. Configuring of a short open window inside the long open window is not allowed and will also cause a WDT1 reset.

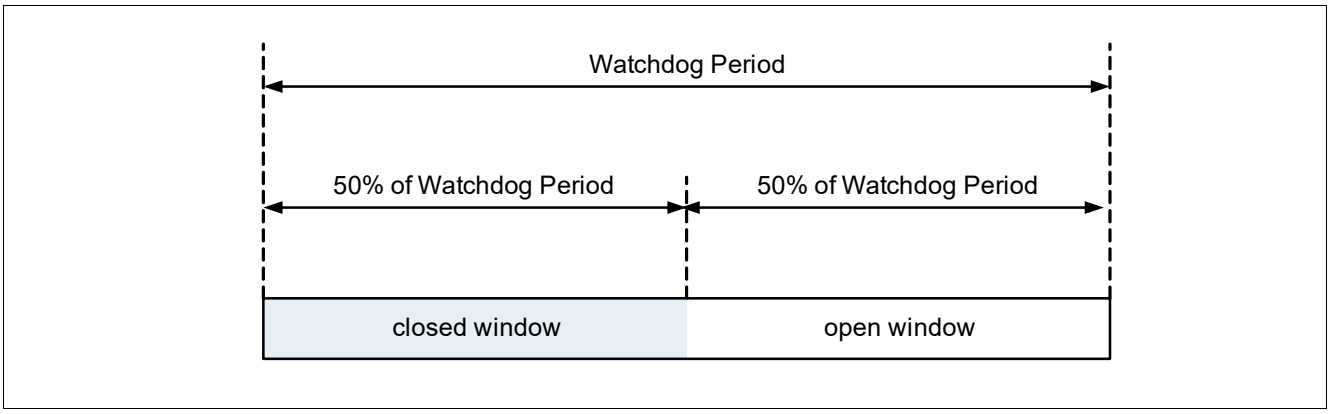


Figure 81 Windowed Watchdog

The first half of the watchdog period is the closed window and the second half is the open window. A trigger of the watchdog has to be done in the open window only. Any trigger in the closed window or failing to trigger the watchdog within the watchdog period will cause a reset. The reset will be indicated by the bit **PMU_ExtWDT** in the reset status register **PMU_RESET_STS** located inside PMU.

Effective open window (safe trigger point)

Due to the variations in the clock source of the WDT1 the effective usable open window, and therefore a safe trigger point, is shorter than 50% of the watchdog period as shown in **Figure 82**.

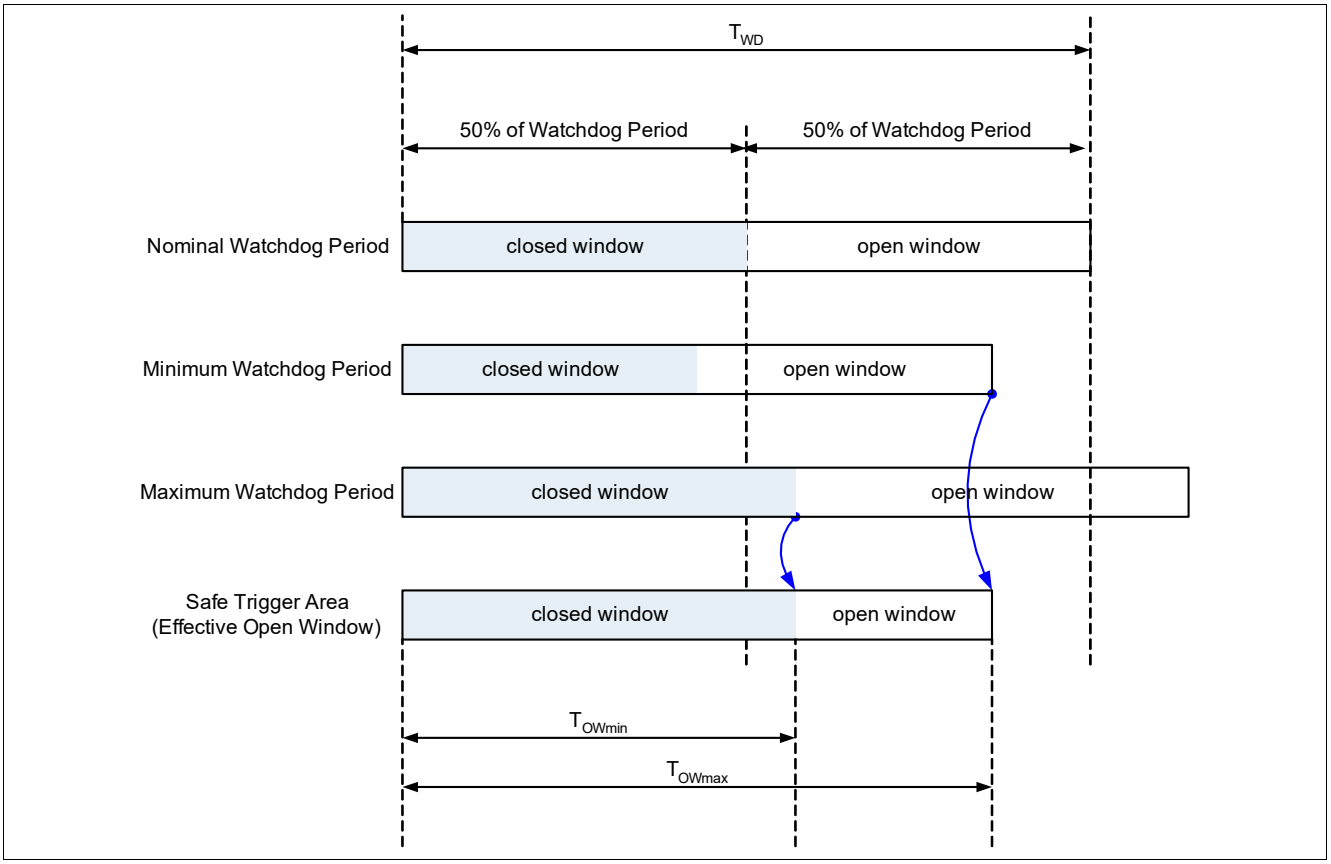


Figure 82 Effective Open Window

E.g. for a variation of 20% and a nominal watchdog period of T_{WD} the start of the effective open window T_{OWmin} is shifted back by 10%, and the end of the effective open window T_{OWmax} is shifted forward by 20%.

Short Open Window (SOW)

Under certain programming conditions, e.g. NVM programming, it might be desired to interrupt the normal windowed watchdog operation. For this purpose a special trigger of a short open window (see **Figure 83**) allows to discard the current window period (also within the closed window) and immediately starts a short open window. The short open window has a fixed length of TSOW independent of the settings of the WDP_SEL bits.

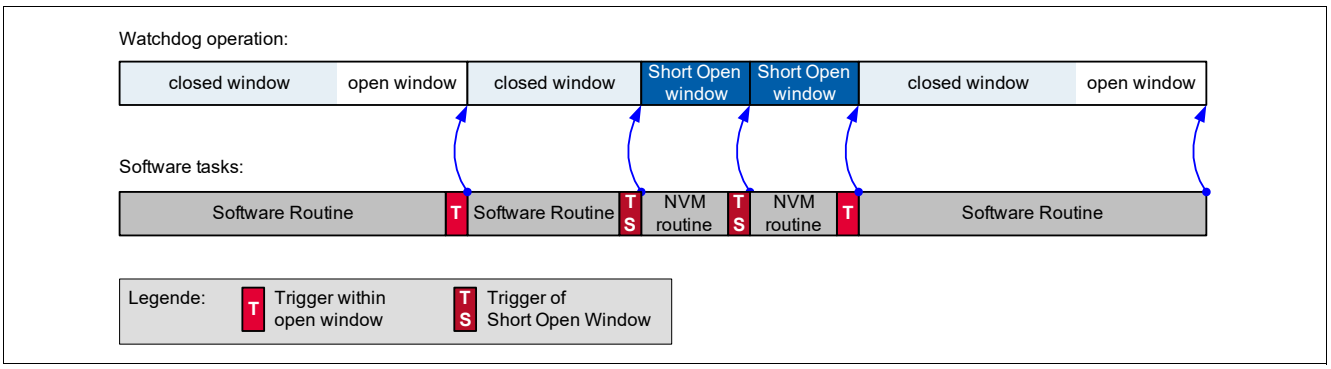


Figure 83 Short Open Window

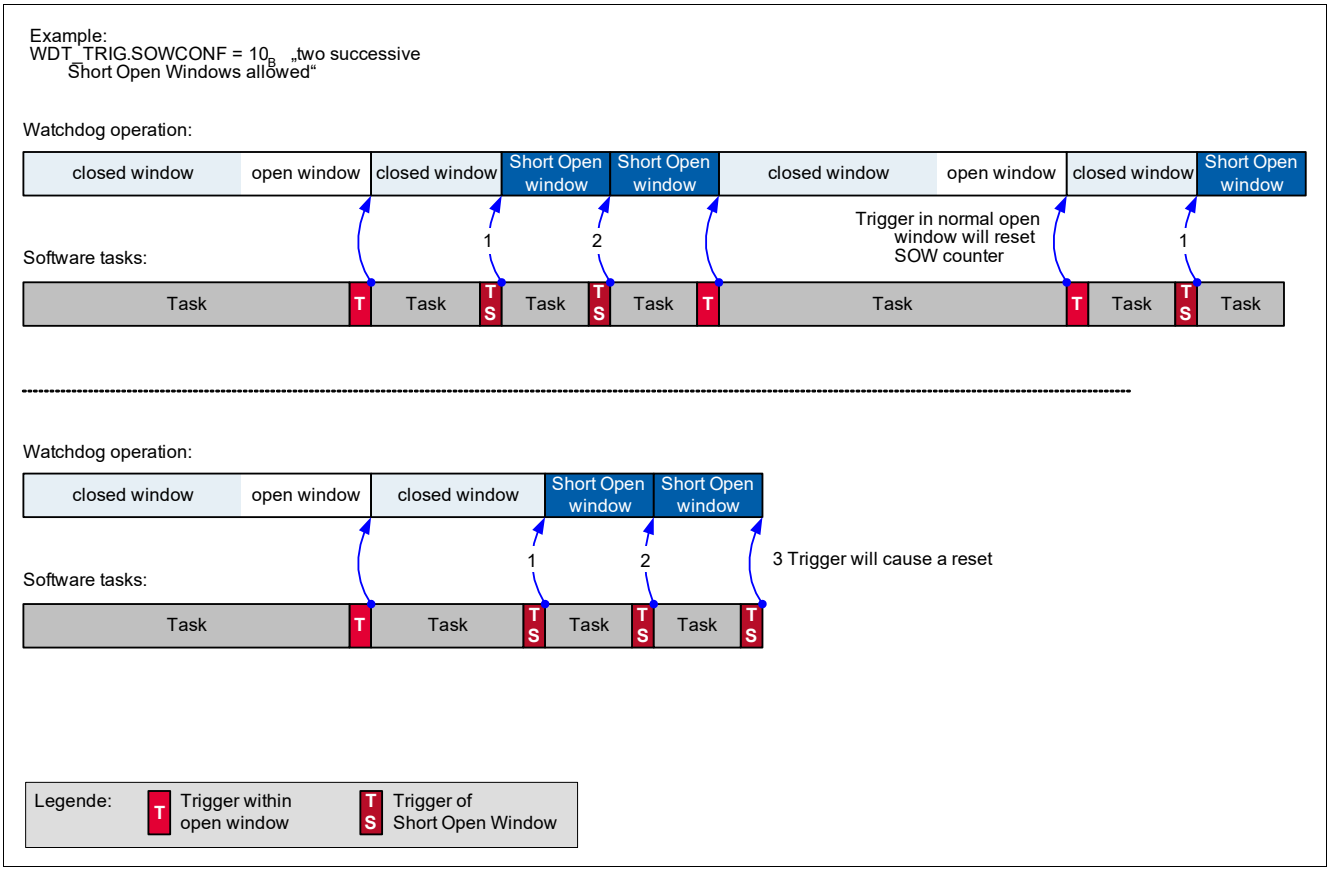


Figure 84 SOW Counter

The mechanism of inserting Short Open Windows has to be enabled/configured with the bits SOWCONF. The configuration allows to insert a maximum of three consecutive Short Open Windows. Each Trigger of the Short Open Window will increase a SOW counter, if the SOW counter exceeds the maximum configured value a reset will be generated. The SOW counter value is reset to 0 by a normal Trigger.

14.3.2.1 Watchdog Register Overview

Table 220 Register Address Space

Module	Base Address	End Address	Note
SCUPM	50006000_H	$50006FFF_H$	SCU_PM

Table 221 Register Overview

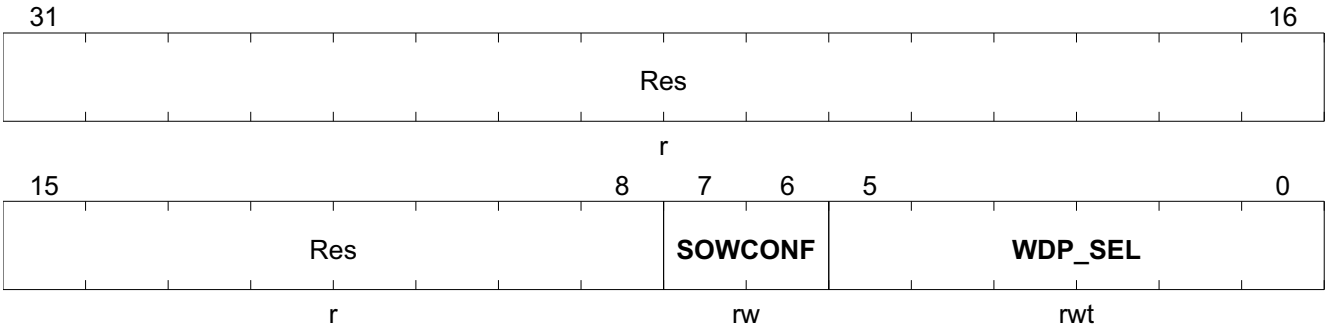
Register Short Name	Register Long Name	Offset Address	Reset Value
Watchdog Register Overview,			
SCUPM_WDT1_TRIG	WDT1 Watchdog Control	34_H	see **Table 222**

The registers are addressed bytewise.

14.3.2.1.1 Watchdog Register

WDT1 Watchdog Control

SCUPM_WDT1_TRIG	Offset	Reset Value
WDT1 Watchdog Control	34_H	see Table 222



Field	Bits	Type	Description
Res	31:8	r	**Reserved** Always read as 0
SOWCONF	7:6	rw	**Short Open Window Configuration** 0_H **DIS**, Short Open Windows disabled[1)] 1_H **SOW1**, one successive Short Open Window allowed 2_H **SOW2**, two successive Short Open Windows allowed 3_H **SOW3**, three successive Short Open Windows allowed
WDP_SEL	5:0	rwt	**Watchdog Period Selection and trigger** Selects the time for the next Watchdog period and allows to trigger the short open window. 00_H **SOW_TRIG**, trigger short open window 01_H **WP_1**, Watchdog period 16 ms 02_H **WP_2**, Watchdog period 32 ms 03_H **WP_3**, Watchdog period 48 ms $\cdots_H$ **…**, $3F_H$ **WP_63**, Watchdog period 1008 ms

1) Writing 00_H to the WDT_TRIG register will cause a reset

Table 222 RESET of SCUPM_WDT1_TRIG

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000_H	RESET_TYPE_3		

15 GPIO Ports and Peripheral I/O

This chapter describes the GPIO Ports of the TLE985xQX. It contains the following sections:

- Introduction to the GPIO Ports (see **Section 15.2**)
- GPIO Port functional descriptions (see **Section 15.3**) This section also describes the mapping of the alternate pin functions
- Registers description of the GPIO module registers (see **Section 15.4**)

The TLE985xQX has 15 port pins organized into three parallel ports: Port 0 (P0), Port 1 (P1) and Port 2 (P2). Each port pin has a pair of internal pull-up and pull-down devices that can be individually enabled or disabled. P0 and P1 are bidirectional and can be used as general purpose input/output (GPIO) or to perform alternate input/output functions for the on-chip peripherals. When configured as an output, the open drain mode can be selected. On Port 2 (P2) analog inputs are shared with general purpose input.

15.1 Features

- 10 GPIOs and 5 analog inputs.
- Strong pull-up at Reset-pin and Hall-inputs (except P2.x)

Bidirectional Port Features (P0, P1)

- Configurable pin direction
- Configurable pull-up/pull-down devices
- Configurable open drain mode
- Configurable drive strength
- Transfer of data through digital inputs and outputs (general purpose I/O)
- Possible readback of pin status when GPIO is configured as output (short detection)
- Alternate input/output for on-chip peripherals

Analog Port Features (P2)

- Configurable pull-up/pull-down devices
- Transfer of data through digital inputs
- Alternate inputs for on-chip peripherals

15.2 Introduction

15.2.1 Port 0 and Port 1

Figure 85 shows the block diagram of an TLE985xQX bidirectional port pin. Each port pin is equipped with a number of control and data bits, thus enabling very flexible usage of the pin. By defining the contents of the control register, each individual pin can be configured as an input or an output. The user can also configure each pin as an open drain pin with or without internal pull-up/pull-down device.

Each bidirectional port pin can be configured for input or output operation. Switching between input and output mode is accomplished through the register Px_DIR (x = 0 or 1), which enables or disables the output and input drivers. A port pin can only be configured as either input or output mode at any one time.

In input mode (default after reset), the output driver is switched off (high-impedance). The actual voltage level present at the port pin is translated into a logic 0 or 1 via a Schmitt-Trigger device and can be read via the register Px_DATA.

In output mode, the output driver is activated and drives the value supplied through the multiplexer to the port pin. In the output driver, each port line can be switched to open drain mode or normal mode (push-pull mode) via the register Px_OD.

The output multiplexer in front of the output driver enables the port output function to be used for different purposes. If the pin is used for general purpose output, the multiplexer is switched by software to the data register Px_DATA. Software can set or clear the bit in Px_DATA and therefore directly influence the state of the port pin. If an on-chip peripheral uses the pin for output signals, alternate output lines (AltDataOut) can be switched via the multiplexer to the output driver circuitry. Selection of the alternate function is defined in registers Px_ALTSEL0 and Px_ALTSEL1. When a port pin is used as an alternate function, its direction must be set accordingly in the register Px_DIR.

Each pin can also be programmed to activate an internal weak pull-up or pull-down device. Register Px_PUDSEL selects whether a pull-up or the pull-down device is activated while register Px_PUDEN enables or disables the pull device.

The port structure used in this device offers the possibility to select the output driver strength and the slew rate. These selections are independent from the output port functionality, such as open-drain, push/pull or input only. The driver strength for each pin can be adapted to the application requirements by registers Px_POCONy (y = 0, 1 or 2) in SCU.

The temperature compensation signals TC[1:0] of all output drivers are connected to all outputs and are controlled by register TCCR in SCU.

*Note: For the definition of Px_POCONy and TCCR registers, refer to **Chapter 7.7.3** of SCU chapter.*

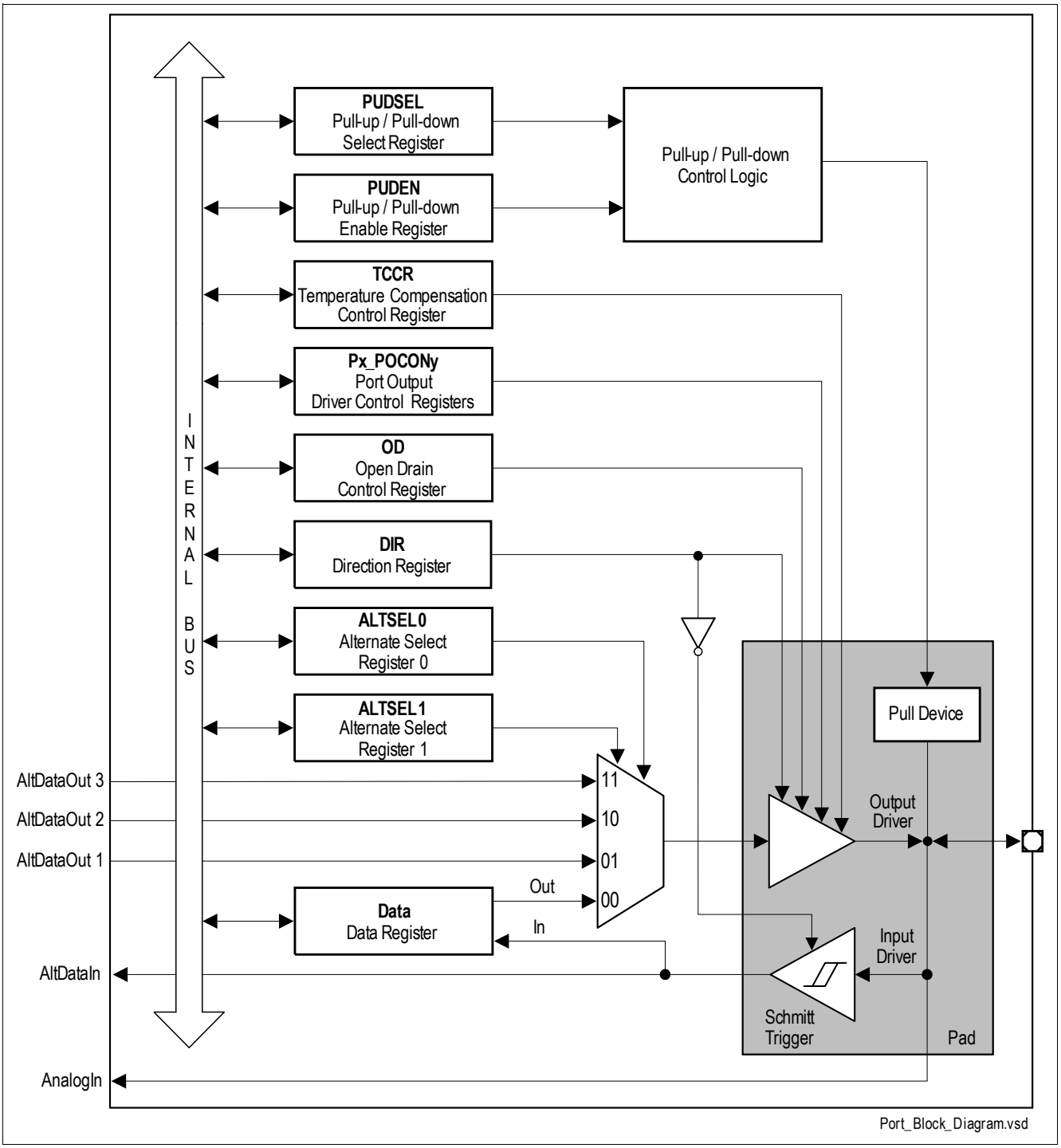


Figure 85 General Structure of Bidirectional Port

15.2.2 Port 2

Figure 86 shows the structure of an input-only port pin. Each P2 pin can only function in input mode. Register P2_DIR is provided to enable or disable the input driver. When the input driver is enabled, the actual voltage level present at the port pin is translated into a logic 0 or 1 via a Schmitt-Trigger device and can be read via the register P2_DATA. Each pin can also be programmed to activate an internal weak pull-up or pull-down device. Register P2_PUDSEL selects whether a pull-up or the pull-down device is activated while register P2_PUDEN enables or disables the pull device. The analog input (AnalogIn) bypasses the digital circuitry and Schmitt-Trigger device for direct feed-through to the ADC input channel.

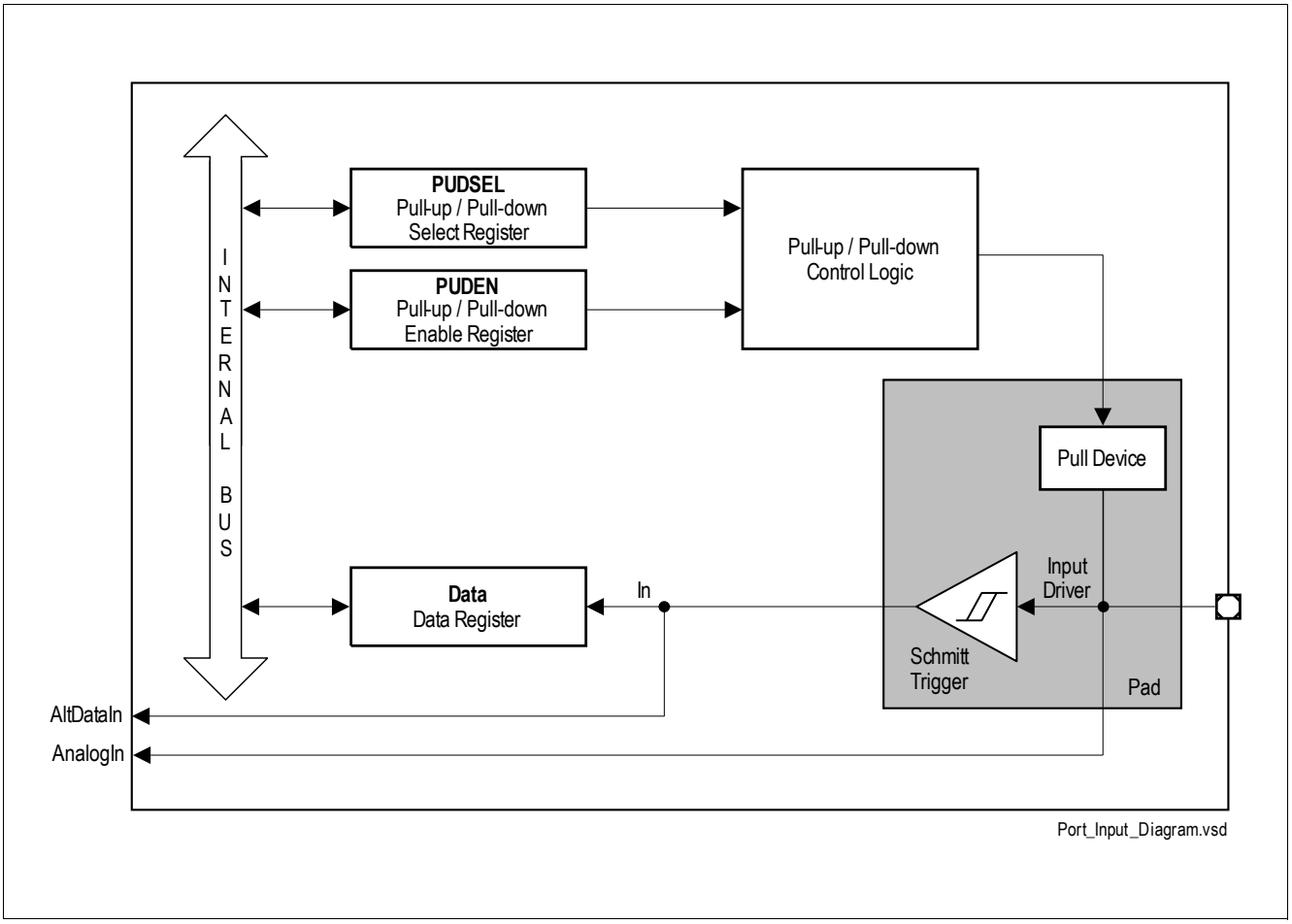


Figure 86 General Structure of Input Port

15.3 Functional Description

The main function of the GPIOs is to provide multiple digital I/Os (Port0.x and Port1.x) and multiple analog and digital Inputs (Port2.x). Each port function can be configured by control and data registers. The registers are defined in **Figure 87** and a detailed register description is provided in **Chapter 15.4**.

An overview of the available alternate functions of the GPIOs is provided in **Chapter 15.3.2**.

15.3.1 Register Controlled Functions

All GPIO functions are controlled by registers. The available register functions are described in the following Chapters.

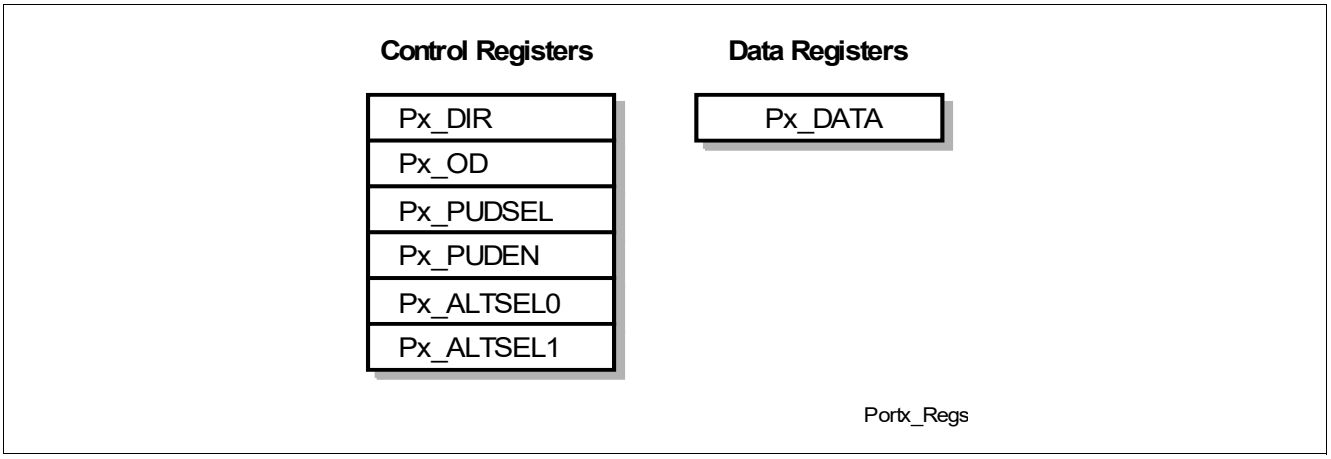


Figure 87 Port Registers

Note: Not all the registers are implemented for each port.

15.3.1.1 Data Registers - PxDATA

If a port pin is used as general purpose output, output data is written into register Px_DATA (Px_PPy_DAT bits) of port x. When the port pin is used as general purpose input, the value at a port pin can be read through the register Px_DATA (Px_PPy_DAT bits). The data register Px_DATA always contains a latched value of the assigned port pin.

Bit Px_DATA.n can only be written if the corresponding pin is set to output, i.e. Px_DIR.n = 1. The contents of Px_DATA.n are output on the assigned pin if the pin is assigned as GPIO pin and the direction is switched/set to output. A read operation of Px_DATA (Px_PPy_DAT bits) returns the register value and not the state of the Px_DATA pins, Px_DATA (Px_PPy_STS bits) can be used to read back the Port Data if the Schmitt-Trigger is enabled in register Px_DIR (Px_PPy_INEN bits).

15.3.1.2 Direction Control- PxDIR

The direction of bidirectional port pins is controlled by the respective direction register Px_DIR. For input-only port pins, register Px_DIR is used to enable or disable the input drivers.

15.3.1.3 Open Drain Control - PxOD

Each pin in output mode can be switched to Open Drain Mode. If driven with 1, no driver will be activated and the pin output state depends on the internal pull-up/pull-down device setting; if driven with 0, the driver's pull-down transistor will be activated.

The open drain mode is controlled by the register Px_OD.

15.3.1.4 Pull-Up/Pull-Down Device - PxPUDSEL PxPUDEN

Internal pull-up/pull-down devices can be optionally applied to a port pin. This offers the possibility to configure the following input characteristics:

- tristate
- high-impedance with a weak pull-up device
- high-impedance with a weak pull-down device

and the following output characteristics:

- push/pull (optional pull-up/pull-down)

- open drain with internal pull-up
- open drain with external pull-up

The pull-up/pull-down device can be fixed or controlled via the registers Px_PUDSEL and Px_PUDEN. Register Px_PUDSEL selects the type of pull-up/pull-down device, while register Px_PUDEN enables or disables it. The pull-up/pull-down device can be selected pinwise.

Note: The selected pull-up/pull-down device is enabled by setting the respective bit in the Px_PUDEN register.

15.3.1.5 Alternate Functions Control - PxALTSEL0/1

Alternate Functions Input Control

The number of alternate functions that uses a pin for input is not limited. Each port control logic of an I/O pin provides several input paths:

- Digital input value via register
- Direct digital input value

The alternate input functions are shown in **Chapter 15.3.2** and are enabled in the Module Registers (i.e. GPT12 or Timers) using a GPIO Input

Alternate Functions Output Control

Alternate output functions are selected via an output multiplexer which can select up to four output lines. This multiplexer can be controlled by the following registers:

- Register Px_ALTSEL0
- Register Px_ALTSEL1

Selection of alternate functions is defined in registers Px_ALTSEL0 and Px_ALTSEL1 and the possible pin assignment is documented in the following **Chapter 15.3.2**.

15.3.2 Alternate Functions

The following chapters describe the Portx.y mapping to their alternate functions.

15.3.2.1 Port 0 Functions

Port 0 alternate function mapping according **Table 223**

Table 223 Port 0 Input/Output Functions

Port Pin	Input/Output	Select	Connected Signal(s)	From/to Module
P0.0	Input	GPI	P0_DATA.P0	
		INP1	T12HR_0	CCU6
		INP2	T4INA	GPT12
		INP3	T2_0	Timer 2
		INP4	SWD_CLK	SWD
		INP5	EXINT2_3	SCU
	Output	GPO	P0_DATA.P0	
		ALT1	T3OUT_0	GPT12
		ALT2	EXF21_0	Timer 21
		ALT3	UART2_RXDO	UART2
P0.1	Input	GPI	P0_DATA.P1	
		INP1	T13HR_0	CCU6
		INP2	UART1_RXD	UART1
		INP3	T2EX_1	Timer 2
		INP4	T21_0	Timer 21
		INP5	EXINT0_3	SCU
		INP6	T4INC	GPT12
		INP7	CAPINA	GPT12
		INP8	SSC12_S_SCK_0	SSC1/2
		INP9	CC62_0	CCU6
	Output	GPO	P0_DATA.P1	
		ALT1	T6OUT_0	GPT12
		ALT2	CC62_0	CCU6
		ALT3	SSC12_M_SCK	SSC1/2

Table 223 Port 0 Input/Output Functions (cont'd)

Port Pin	Input/Output	Select	Connected Signal(s)	From/to Module
P0.2	Input	GPI	P0_DATA.P2	
		INP1	T2EUDA	GPT12
		INP2	CTRAP_0	CCU6
		INP3	SSC12_M_MRST_0	SSC1/2
		INP4	T21EX_0	Timer 21
		INP5	EXINT1_3	SCU
	Output	GPO	P0_DATA.P2	
		ALT1	SSC12_S_MRST	SSC1/2
		ALT2	UART1_TXD	UART1
		ALT3	EXF2_0	Timer 2
P0.3	Input	GPI	P0_DATA.P3	
		INP1	SSC1_S_SCK	SSC1
		INP2	T4EUDA	GPT12
		INP3	CAPINB	GPT12
		INP4	EXINT1_2	SCU
		INP5	T3EUDD	GPT12
		INP6	CCPOS0_1	CCU6
	Output	GPO	P0_DATA.P3	
		ALT1	SSC1_M_SCK	SSC1
		ALT2	T6OFL	GPT12
		ALT3	T6OUT_1	GPT12
P0.4	Input	GPI	P0_DATA.P4	
		INP1	SSC1_S_MTSR	SSC1
		INP2	CC60_0	CCU6
		INP3	T21_2	Timer 21
		INP4	EXINT2_2	SCU
		INP5	T3EUDA	GPT12
		INP6	CCPOS1_1	CCU6
	Output	GPO	P0_DATA.P4	
		ALT1	SSC1_M_MTSR	SSC1
		ALT2	CC60_0	CCU6
		ALT3	CLKOUT_0	SCU

Table 223 Port 0 Input/Output Functions (cont'd)

Port Pin	Input/Output	Select	Connected Signal(s)	From/to Module
P0.5	Input	GPI	P0_DATA.P5	
		INP1	SSC1_M_MRST	SSC1
		INP2	EXINT0_0	SCU
		INP3	T21EX_2	Timer 21
		INP4	T5INA	GPT12
		INP5	CCPOS2_1	CCU6
	Output	GPO	P0_DATA.P5	
		ALT1	SSC1_S_MRST	SSC1
		ALT2	COUT60_0	CCU6
		ALT3	LIN_RXD	LIN

15.3.2.2 Port 1 Functions

Port 1 alternate function mapping according **Table 224**

Table 224 Port 1 Input / Output Functions

Port Pin	Input/Output	Select	Connected Signal(s)	From/to Module
P1.0	Input	GPI	P1_DATA.P0	
		INP1	T3INC	GPT12
		INP2	CC61_0	CCU6
		INP3	SSC2_S_SCK	SSC2
		INP4	T4EUDB	GPT12
	Output	GPO	P1_DATA.P0	
		ALT1	SSC2_M_SCK	SSC2
		ALT2	CC61_0	CCU6
		ALT3	UART2_TXD	UART2
P1.1	Input	GPI	P1_DATA.P1	
		INP1	T6EUDA	GPT12
		INP2	T5INB	GPT12
		INP3	T3EUDC	GPT12
		INP4	SSC2_S_MTSR	SSC2
		INP5	T21EX_3	Timer 21
		INP6	UART2_RXD	UART2
	Output	GPO	P1_DATA.P1	
		ALT1	SSC2_M_MTSR	SSC2
		ALT2	COUT61_0	CCU6
		ALT3	EXF21_1	Timer 21
P1.2	Input	GPI	P1_DATA.P2	
		INP1	EXINT0_1	SCU
		INP2	T21_1	Timer 21
		INP3	T2INA	GPT12
		INP4	SSC2_M_MRST	SSC2
		INP5	CCPOS2_2	CCU6
	Output	GPO	P1_DATA.P2	
		ALT1	SSC2_S_MRST	SSC2
		ALT2	COUT63_0	CCU6
		ALT3	T3OUT_1	GPT12

Table 224 Port 1 Input / Output Functions (cont'd)

Port Pin	Input/Output	Select	Connected Signal(s)	From/to Module
P1.4	Input	GPI	P1_DATA.P4	
		INP1	EXINT2_1	SCU
		INP2	T21EX_1	Timer 21
		INP3	T2INB	GPT12
		INP4	T5EUDA	GPT12
		INP5	SSC12_S_MTSR_0	SSC1/2
		INP6	CCPOS1_2	CCU6
	Output	GPO	P1_DATA.P4	
		ALT1	CLKOUT_1	SCU
		ALT2	COUT62_0	CCU6
		ALT3	SSC12_M_MTSR	SSC1/2

15.3.2.3 Port 2 Functions

Port 2 alternate function mapping according **Table 225**

Table 225 Port 2 Input Functions

Port Pin	Input/Output	Select	Connected Signal(s)	From/to Module
P2.0	Input	GPI	P2_DATA.P0	
		INP1	EXINT1_1	SCU
		INP2	CCPOS0_2	CCU6
		INP3	T5EUDB	GPT12
		INP4	T13HR_2	CCU6
		ANALOG	AN0	ADC
		IN	XTAL (in)[1)]	XTAL
P2.1	Input	GPI	P2_DATA.P1	
		INP1	CCPOS0_0	CCU6
		INP2	EXINT1_0	SCU
		INP3	T12HR_1	CCU6
		INP4	CC61_1	CCU6
		INP5	T4EUDD	GPT12
		INP6	T2EX_3	Timer2
		INP7	LIN_TXD	LIN
		INP8	SSC12_S_SCK_1	SSC1/2
		ANALOG	AN1	ADC
P2.2	Input / Output	GPI	P2_DATA.P2	
		INP1	T6EUDB	GPT12
		INP2	T2EX_0	Timer 2
		INP3	T12HR_2	CCU6
		INP4	CTRAP_2	CCU6
		ANALOG	AN2	ADC
		OUT	XTAL (out)[1)]	XTAL

Table 225 Port 2 Input Functions (cont'd)

Port Pin	Input/Output	Select	Connected Signal(s)	From/to Module
P2.3	Input	GPI	P2_DATA.P3	
		INP1	CCPOS1_0	CCU6
		INP2	EXINT0_2	SCU
		INP3	CTRAP_1	CCU6
		INP4	T3IND	GPT12
		INP5	CC60_1	CCU6
		INP6	T2EUDB	GPT12
		INP7	T2_2	Timer2
		INP8	T2EX_2	Timer2
		INP9	SSC12_S_MTSR_1	SSC1/2
		ANALOG	AN3	ADC
P2.7	Input	GPI	P2_DATA.P7	
		INP1	CCPOS2_0	CCU6
		INP2	EXINT2_0	SCU
		INP3	T13HR_1	CCU6
		INP4	CC62_1	CCU6
		INP5	T3EUDB	GPT12
		INP6	T4EUDC	GPT12
		INP7	T2_1	Timer2
		INP8	SSC12_M_MRST_1	SSC1/2
		ANALOG	AN7	ADC

1) configurable by user

15.4 Register Description

15.4.1 Port 0 Register Description

Table 226 Register Address Space

Module	Base Address	End Address	Note
PORT	48028000_H	$48029FFF_H$	Ports
SCU	$5000\ 5000_H$	$5000\ 5FFF_H$	System Control Unit - Digital Modules

Table 227 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Port 0 Register Description,			
P0_DATA	Port 0 Data Register	00_H	$000000XX_H$
P0_DIR	Port 0 Direction Register	04_H	00000000_H
P0_OD	Port 0 Open Drain Control Register	08_H	00000000_H
P0_PUDSEL	Port 0 Pull-Up/Pull-Down Select Register	$0C_H$	$0000007B_H$
P0_PUDEN	Port 0 Pull-Up/Pull-Down Enable Register	10_H	$0000007F_H$
P0_ALTSEL0	Port 0 Alternate Select Register 0	14_H	00000000_H
P0_ALTSEL1	Port 0 Alternate Select Register 1	18_H	00000000_H
SCU_P0_POCON0	Port Output Control Register	$0E8_H$	00000000_H

The registers are addressed wordwise.

Data Register

P0_DATA	Offset	Reset Value
Port 0 Data Register	00_H	see Table 228

31	22	21	20	19	18	17	16
Res		PP5_STS	PP4_STS	PP3_STS	PP2_STS	PP1_STS	PP0_STS
r		r	r	r	r	r	r

15	6	5	4	3	2	1	0
RES		PP5	PP4	PP3	PP2	PP1	PP0
r		rwh	rwh	rwh	rwh	rwh	rwh

Field	Bits	Type	Description
Res	31:22	r	**Reserved** Always read as 0
PP5_STS	21	r	**Port 0 Pin 5 Data Value (read back of Port Data when IO is configured as output)** 0_B **0**, Port 0 pin 5 data value = 0 1_B **1**, Port 0 pin 5 data value = 1
PP4_STS	20	r	**Port 0 Pin 4 Data Value (read back of Port Data when IO is configured as output)** 0_B **0**, Port 0 pin 4 data value = 0 1_B **1**, Port 0 pin 4 data value = 1
PP3_STS	19	r	**Port 0 Pin 3 Data Value (read back of Port Data when IO is configured as output)** 0_B **0**, Port 0 pin 3 data value = 0 1_B **1**, Port 0 pin 3 data value = 1
PP2_STS	18	r	**Port 0 Pin 2 Data Value (read back of Port Data when IO is configured as output)** 0_B **0**, Port 0 pin 2 data value = 0 1_B **1**, Port 0 pin 2 data value = 1
PP1_STS	17	r	**Port 0 Pin 1 Data Value (read back of Port Data when IO is configured as output)** 0_B **0**, Port 0 pin 1 data value = 0 1_B **1**, Port 0 pin 1 data value = 1
PP0_STS	16	r	**Port 0 Pin 0 Data Value (read back of Port Data when IO is configured as output)** 0_B **0**, Port 0 pin 0 data value = 0 1_B **1**, Port 0 pin 0 data value = 1
RES	15:6	r	**Reserved** Returns 0 if read.
PP5	5	rwh	**Port 0 Pin 5 Data Value** 0_B **0**, Port 0 pin 5 data value = 0 1_B **1**, Port 0 pin 5 data value = 1
PP4	4	rwh	**Port 0 Pin 4 Data Value** 0_B **0**, Port 0 pin 4 data value = 0 1_B **1**, Port 0 pin 4 data value = 1
PP3	3	rwh	**Port 0 Pin 3 Data Value** 0_B **0**, Port 0 pin 3 data value = 0 1_B **1**, Port 0 pin 3 data value = 1
PP2	2	rwh	**Port 0 Pin 2 Data Value** 0_B **0**, Port 0 pin 2 data value = 0 1_B **1**, Port 0 pin 2 data value = 1
PP1	1	rwh	**Port 0 Pin 1 Data Value** 0_B **0**, Port 0 pin 1 data value = 0 1_B **1**, Port 0 pin 1 data value = 1

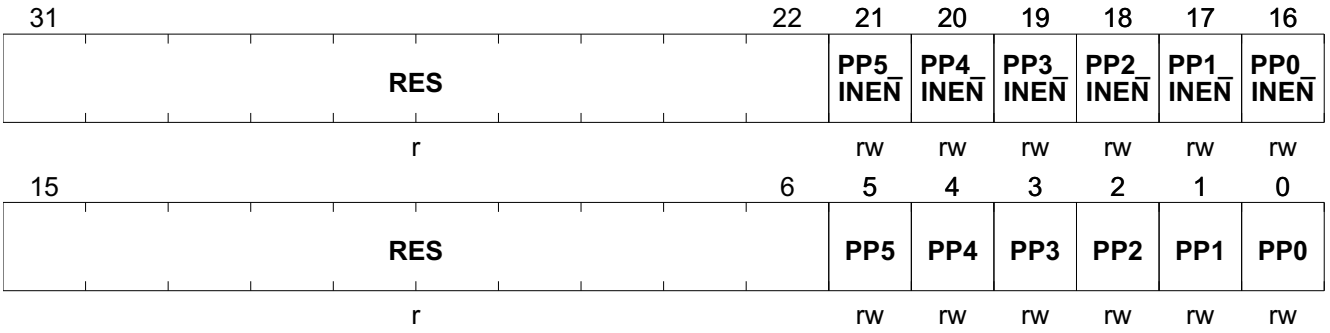
Field	Bits	Type	Description
PP0	0	rwh	**Port 0 Pin 0 Data Value** 0_B **0**, Port 0 pin 0 data value = 0 1_B **1**, Port 0 pin 0 data value = 1

Table 228 RESET of P0_DATA

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$000000XX_H$	RESET_TYPE_3		

Direction Register

P0_DIR	Offset	Reset Value
Port 0 Direction Register	04_H	see Table 229



Field	Bits	Type	Description
RES	31:22	r	**Reserved** Returns 0 if read.
PP5_INEN	21	rw	**Port 0 Pin 5 Input Schmitt Trigger enable (only valid if IO is configured as output)** 0_B **0**, Schmitt Trigger is disabled 1_B **1**, Schmitt Trigger is enabled
PP4_INEN	20	rw	**Port 0 Pin 4 Input Schmitt Trigger enable (only valid if IO is configured as output)** 0_B **0**, Schmitt Trigger is disabled 1_B **1**, Schmitt Trigger is enabled
PP3_INEN	19	rw	**Port 0 Pin 3 Input Schmitt Trigger enable (only valid if IO is configured as output)** 0_B **0**, Schmitt Trigger is disabled 1_B **1**, Schmitt Trigger is enabled
PP2_INEN	18	rw	**Port 0 Pin 2 Input Schmitt Trigger enable (only valid if IO is configured as output)** 0_B **0**, Schmitt Trigger is disabled 1_B **1**, Schmitt Trigger is enabled

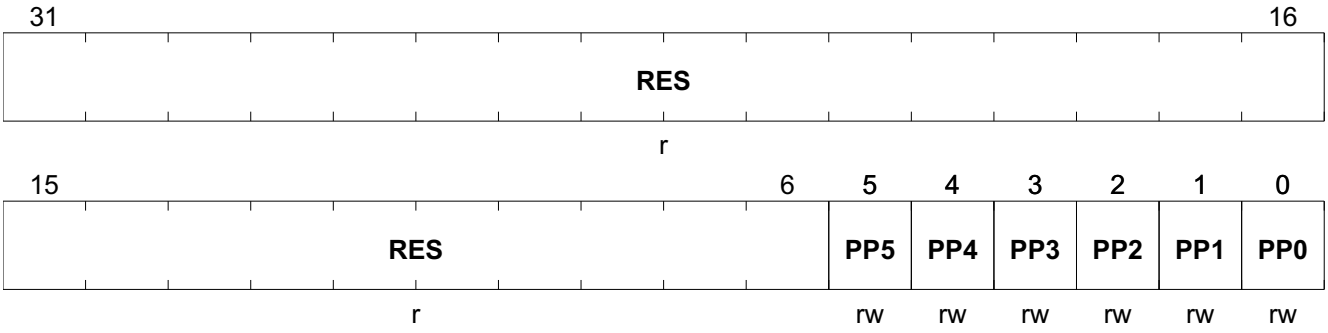
Field	Bits	Type	Description
PP1_INEN	17	rw	**Port 0 Pin 1 Input Schmitt Trigger enable (only valid if IO is configured as output)** 0_B **0**, Schmitt Trigger is disabled 1_B **1**, Schmitt Trigger is enabled
PP0_INEN	16	rw	**Port 0 Pin 0 Input Schmitt Trigger enable (only valid if IO is configured as output)** 0_B **0**, Schmitt Trigger is disabled 1_B **1**, Schmitt Trigger is enabled
RES	15:6	r	**Reserved** Returns 0 if read.
PP5	5	rw	**Port 0 Pin 5 Direction Control** 0_B **0**, Direction is set to input 1_B **1**, Direction is set to output
PP4	4	rw	**Port 0 Pin 4 Direction Control** 0_B **0**, Direction is set to input 1_B **1**, Direction is set to output
PP3	3	rw	**Port 0 Pin 3 Direction Control** 0_B **0**, Direction is set to input 1_B **1**, Direction is set to output
PP2	2	rw	**Port 0 Pin 2 Direction Control** 0_B **0**, Direction is set to input 1_B **1**, Direction is set to output
PP1	1	rw	**Port 0 Pin 1 Direction Control** 0_B **0**, Direction is set to input 1_B **1**, Direction is set to output
PP0	0	rw	**Port 0 Pin 0 Direction Control** 0_B **0**, Direction is set to input 1_B **1**, Direction is set to output

Table 229 RESET of P0_DIR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Open Drain Control Register

P0_OD	Offset	Reset Value
Port 0 Open Drain Control Register	08_H	see Table 230



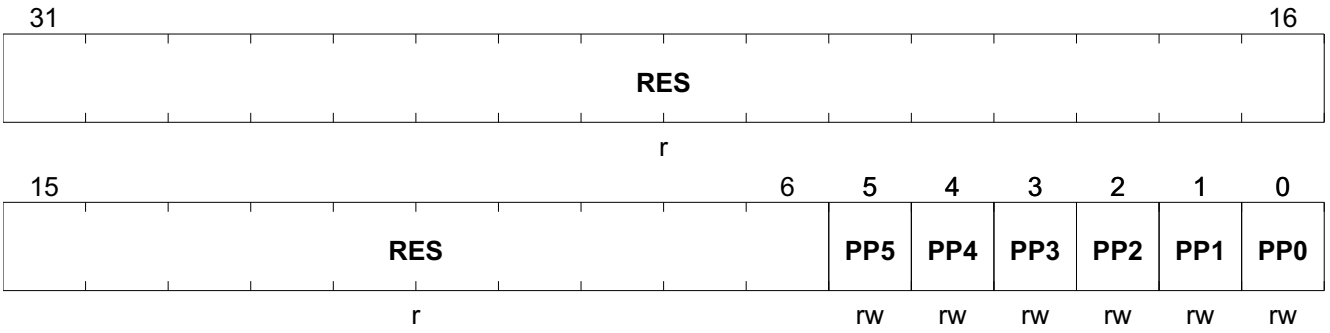
Field	Bits	Type	Description
RES	31:6	r	**Reserved** Returns 0 if read.
PP5	5	rw	**Port 0 Pin 5 Open Drain Mode** 0_B **Normal Mode**, Output is actively driven for 0 and 1 state 1_B **Open Drain Mode**, Output is actively driven only for 0 state
PP4	4	rw	**Port 0 Pin 4 Open Drain Mode** 0_B **Normal Mode**, Output is actively driven for 0 and 1 state 1_B **Open Drain Mode**, Output is actively driven only for 0 state
PP3	3	rw	**Port 0 Pin 3 Open Drain Mode** 0_B **Normal Mode**, Output is actively driven for 0 and 1 state 1_B **Open Drain Mode**, Output is actively driven only for 0 state
PP2	2	rw	**Port 0 Pin 2 Open Drain Mode** 0_B **Normal Mode**, Output is actively driven for 0 and 1 state 1_B **Open Drain Mode**, Output is actively driven only for 0 state
PP1	1	rw	**Port 0 Pin 1 Open Drain Mode** 0_B **Normal Mode**, Output is actively driven for 0 and 1 state 1_B **Open Drain Mode**, Output is actively driven only for 0 state
PP0	0	rw	**Port 0 Pin 0 Open Drain Mode** 0_B **Normal Mode**, Output is actively driven for 0 and 1 state 1_B **Open Drain Mode**, Output is actively driven only for 0 state

Table 230 RESET of P0_OD

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Port 0 Pull-Up/Pull-Down Select Register

P0_PUDSEL	Offset	Reset Value
Port 0 Pull-Up/Pull-Down Select Register	$0C_H$	see Table 231



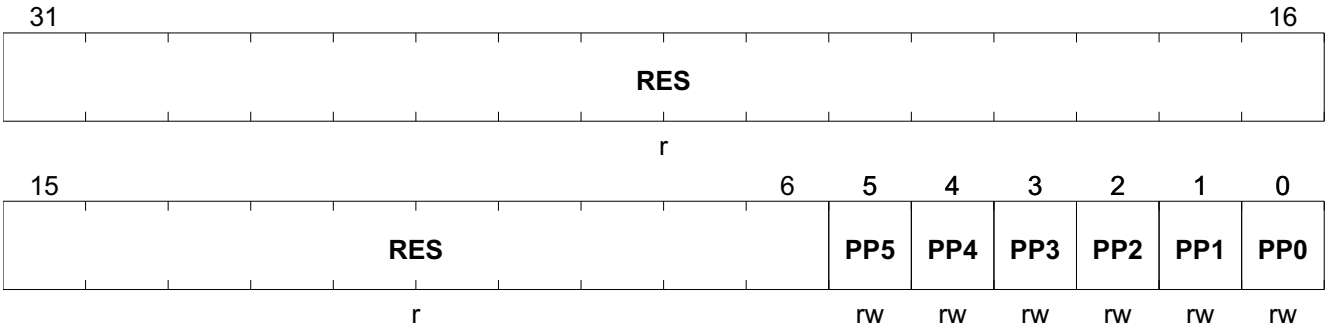
Field	Bits	Type	Description
RES	31:6	r	**Reserved** Returns 0 if read.
PP5	5	rw	**Pull-Up/Pull-Down Select Port 0 Bit 5** 0_B **Pull-down**, Pull-down device is selected 1_B **Pull-up**, Pull-up device is selected
PP4	4	rw	**Pull-Up/Pull-Down Select Port 0 Bit 4** 0_B **Pull-down**, Pull-down device is selected 1_B **Pull-up**, Pull-up device is selected
PP3	3	rw	**Pull-Up/Pull-Down Select Port 0 Bit 3** 0_B **Pull-down**, Pull-down device is selected 1_B **Pull-up**, Pull-up device is selected
PP2	2	rw	**Pull-Up/Pull-Down Select Port 0 Bit 2** 0_B **Pull-down**, Pull-down device is selected 1_B **Pull-up**, Pull-up device is selected
PP1	1	rw	**Pull-Up/Pull-Down Select Port 0 Bit 1** 0_B **Pull-down**, Pull-down device is selected 1_B **Pull-up**, Pull-up device is selected
PP0	0	rw	**Pull-Up/Pull-Down Select Port 0 Bit 0** 0_B **Pull-down**, Pull-down device is selected 1_B **Pull-up**, Pull-up device is selected

Table 231 RESET of P0_PUDSEL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000007B_H$	RESET_TYPE_3		

Port 0 Pull-Up/Pull-Down Enable Register

P0_PUDEN	Offset	Reset Value
Port 0 Pull-Up/Pull-Down Enable Register	10_H	see Table 232



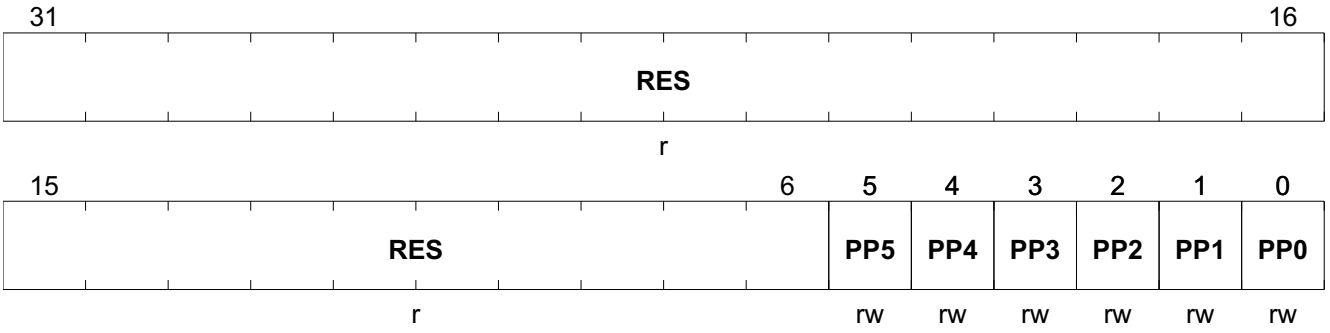
Field	Bits	Type	Description
RES	31:6	r	**Reserved** Returns 0 if read.
PP5	5	rw	**Pull-Up/Pull-Down Enable at Port 0 Bit 5** 0_B **Disabled**, Pull-up or Pull-down device is disabled 1_B **Enabled**, Pull-up or Pull-down device is enabled
PP4	4	rw	**Pull-Up/Pull-Down Enable at Port 0 Bit 4** 0_B **Disabled**, Pull-up or Pull-down device is disabled 1_B **Enabled**, Pull-up or Pull-down device is enabled
PP3	3	rw	**Pull-Up/Pull-Down Enable at Port 0 Bit 3** 0_B **Disabled**, Pull-up or Pull-down device is disabled 1_B **Enabled**, Pull-up or Pull-down device is enabled
PP2	2	rw	**Pull-Up/Pull-Down Enable at Port 0 Bit 2** 0_B **Disabled**, Pull-up or Pull-down device is disabled 1_B **Enabled**, Pull-up or Pull-down device is enabled
PP1	1	rw	**Pull-Up/Pull-Down Enable at Port 0 Bit 1** 0_B **Disabled**, Pull-up or Pull-down device is disabled 1_B **Enabled**, Pull-up or Pull-down device is enabled
PP0	0	rw	**Pull-Up/Pull-Down Enable at Port 0 Bit 0** 0_B **Disabled**, Pull-up or Pull-down device is disabled 1_B **Enabled**, Pull-up or Pull-down device is enabled

Table 232 RESET of P0_PUDEN

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000007F_H$	RESET_TYPE_3		

Alternate Output Select Register

P0_ALTSEL0	Offset	Reset Value
Port 0 Alternate Select Register 0	14_H	see Table 233



Field	Bits	Type	Description
RES	31:6	r	**Reserved** Returns 0 if read.
PP5	5	rw	**See Table 236**
PP4	4	rw	**See Table 236**
PP3	3	rw	**See Table 236**
PP2	2	rw	**See Table 236**
PP1	1	rw	**See Table 236**
PP0	0	rw	**See Table 236**

Table 233 RESET of P0_ALTSEL0

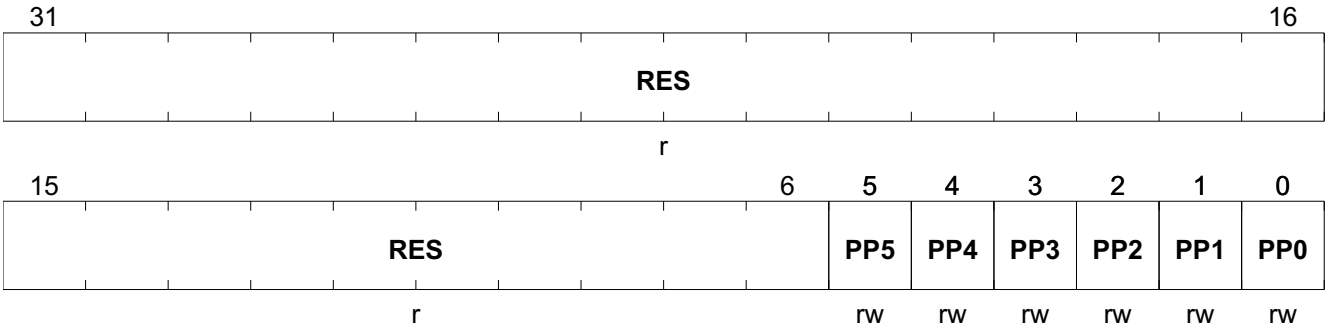
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Table 234 Function of Bits P0_ALTSEL0.PPn and P0_ALTSEL1.PPn

P0_ALTSEL0.PPn	P0_ALTSEL1.PPn	Function
0	0	Normal GPIO
1	0	Alternate Select 1
0	1	Alternate Select 2
1	1	Alternate Select 3

Alternate Output Select Register

P0_ALTSEL1	Offset	Reset Value
Port 0 Alternate Select Register 1	18_H	see Table 235



Field	Bits	Type	Description
RES	31:6	r	**Reserved** Returns 0 if read.
PP5	5	rw	**See Table 236**
PP4	4	rw	**See Table 236**
PP3	3	rw	**See Table 236**
PP2	2	rw	**See Table 236**
PP1	1	rw	**See Table 236**
PP0	0	rw	**See Table 236**

Table 235 RESET of P0_ALTSEL1

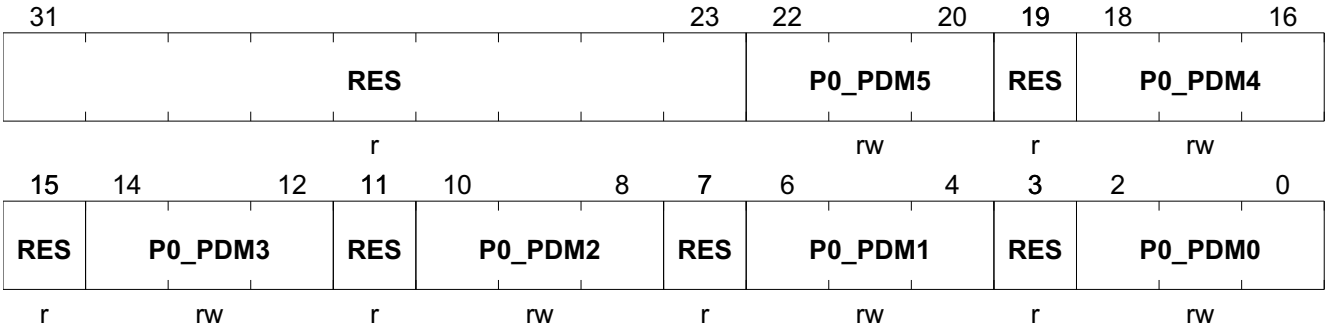
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Table 236 Function of Bits P0_ALTSEL0.PPn and P0_ALTSEL1.PPn

P0_ALTSEL0.PPn	P0_ALTSEL1.PPn	Function
0	0	Normal GPIO
1	0	Alternate Select 1
0	1	Alternate Select 2
1	1	Alternate Select 3

Port Output Control Register

SCU_P0_POCON0	Offset	Reset Value
Port Output Control Register	$0E8_H$	see Table 237



Field	Bits	Type	Description
RES	31:23	r	**Reserved** Returns 0 if read; should be written with 0.
P0_PDM5	22:20	rw	**P0.5 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong driver and sharp edge mode**, 001_B **Strong driver and medium edge mode**, 010_B **Strong driver and soft edge mode**, 011_B **Weak Driver**, 100_B **Medium Driver**, 101_B **Medium Driver**, 110_B **Medium Driver**, 111_B **Weak Driver**,
RES	19	r	**Reserved** Returns 0 if read; should be written with 0.
P0_PDM4	18:16	rw	**P0.4 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong driver and sharp edge mode**, 001_B **Strong driver and medium edge mode**, 010_B **Strong driver and soft edge mode**, 011_B **Weak Driver**, 100_B **Medium Driver**, 101_B **Medium Driver**, 110_B **Medium Driver**, 111_B **Weak Driver**,
RES	15	r	**Reserved** Returns 0 if read; should be written with 0.
P0_PDM3	14:12	rw	**P0.3 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong driver and sharp edge mode**, 001_B **Strong driver and medium edge mode**, 010_B **Strong driver and soft edge mode**, 011_B **Weak Driver**, 100_B **Medium Driver**, 101_B **Medium Driver**, 110_B **Medium Driver**, 111_B **Weak Driver**,

Field	Bits	Type	Description
RES	11	r	**Reserved** Returns 0 if read; should be written with 0.
P0_PDM2	10:8	rw	**P0.2 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong driver and sharp edge mode**, 001_B **Strong driver and medium edge mode**, 010_B **Strong driver and soft edge mode**, 011_B **Weak Driver**, 100_B **Medium Driver**, 101_B **Medium Driver**, 110_B **Medium Driver**, 111_B **Weak Driver**,
RES	7	r	**Reserved** Returns 0 if read; should be written with 0.
P0_PDM1	6:4	rw	**P0.1 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong driver and sharp edge mode**, 001_B **Strong driver and medium edge mode**, 010_B **Strong driver and soft edge mode**, 011_B **Weak Driver**, 100_B **Medium Driver**, 101_B **Medium Driver**, 110_B **Medium Driver**, 111_B **Weak Driver**,
RES	3	r	**Reserved** Returns 0 if read; should be written with 0.
P0_PDM0	2:0	rw	**P0.0 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong driver and sharp edge mode**, 001_B **Strong driver and medium edge mode**, 010_B **Strong driver and soft edge mode**, 011_B **Weak Driver**, 100_B **Medium Driver**, 101_B **Medium Driver**, 110_B **Medium Driver**, 111_B **Weak Driver**,

1) Defines the current the respective driver can deliver to the external circuitry.

2) Defines the switching characteristics to the respective new output driver. This also influences the peak currents through the driver when producing an edge, i.e. when changing the output level.

Table 237 RESET of SCU_P0_POCON0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

15.4.2 Port 1 Register Description

Table 238 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Port 1 Register Description,			
P1_DATA	Port 1 Data Register	20_H	$000000XX_H$
P1_DIR	Port 1 Direction Register	24_H	00000000_H
P1_OD	Port 1 Open Drain Control Register	28_H	00000000_H
P1_PUDSEL	Port 1 Pull-Up/Pull-Down Select Register	$2C_H$	00000017_H
P1_PUDEN	Port 1 Pull-Up/Pull-Down Enable Register	30_H	00000000_H
P1_ALTSEL0	Port 1 Alternate Select Register 0	34_H	00000000_H
P1_ALTSEL1	Port 1 Alternate Select Register 1	38_H	00000000_H
SCU_P1_POCON0	Port Output Control Register	$0F8_H$	00000000_H

The registers are addressed wordwise.

Data Register

P1_DATA **Offset** **Reset Value**

Port 1 Data Register **20_H** **see Table 239**

31	21	20	19	18	17	16
RES		PP4_STS	RES	PP2_STS	PP1_STS	PP0_STS
r		r	r	r	r	r

15	5	4	3	2	1	0
RES		PP4	RES	PP2	PP1	PP0
r		rwh	r	rwh	rwh	rwh

Field	Bits	Type	Description
RES	31:21	r	**Reserved** Returns 0 if read.
PP4_STS	20	r	**Port 1 Pin 4 Data Value (read back of Port Data when IO is configured as output)** 0_B **0**, Port 0 pin 4 data value = 0 1_B **1**, Port 0 pin 4 data value = 1
RES	19	r	**Reserved** Returns 0 if read.

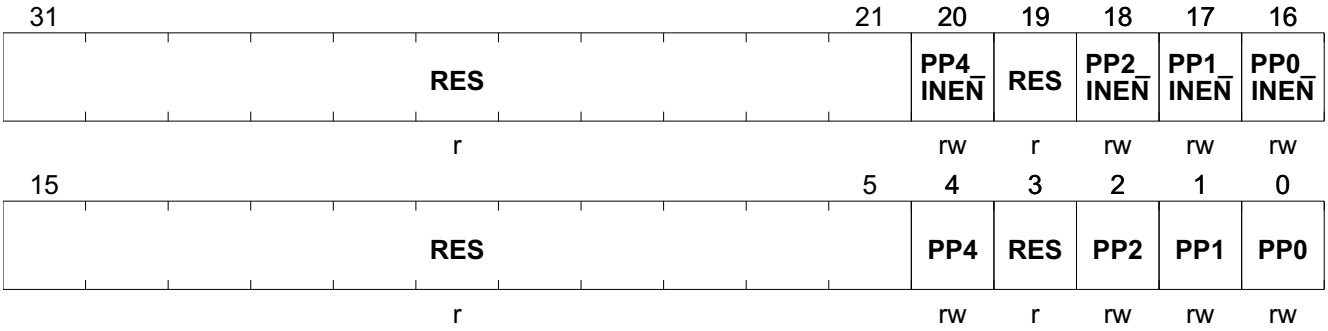
Field	Bits	Type	Description
PP2_STS	18	r	**Port 1 Pin 2 Data Value (read back of Port Data when IO is configured as output)** 0_B **0**, Port 0 pin 2 data value = 0 1_B **1**, Port 0 pin 2 data value = 1
PP1_STS	17	r	**Port 1 Pin 1 Data Value (read back of Port Data when IO is configured as output)** 0_B **0**, Port 0 pin 1 data value = 0 1_B **1**, Port 0 pin 1 data value = 1
PP0_STS	16	r	**Port 1 Pin 0 Data Value (read back of Port Data when IO is configured as output)** 0_B **0**, Port 0 pin 0 data value = 0 1_B **1**, Port 0 pin 0 data value = 1
RES	15:5	r	**Reserved** Returns 0 if read.
PP4	4	rwh	**Port 1 Pin 4 Data Value** 0_B **0**, Port 0 pin 4 data value = 0 1_B **1**, Port 0 pin 4 data value = 1
RES	3	r	**Reserved** Returns 0 if read.
PP2	2	rwh	**Port 1 Pin 2 Data Value** 0_B **0**, Port 0 pin 2 data value = 0 1_B **1**, Port 0 pin 2 data value = 1
PP1	1	rwh	**Port 1 Pin 1 Data Value** 0_B **0**, Port 0 pin 1 data value = 0 1_B **1**, Port 0 pin 1 data value = 1
PP0	0	rwh	**Port 1 Pin 0 Data Value** 0_B **0**, Port 0 pin 0 data value = 0 1_B **1**, Port 0 pin 0 data value = 1

Table 239 RESET of P1_DATA

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$000000XX_H$	RESET_TYPE_3		

Direction Register

P1_DIR	Offset	Reset Value
Port 1 Direction Register	**24_H**	**see Table 240**



Field	Bits	Type	Description
RES	31:21	r	**Reserved** Returns 0 if read.
PP4_INEN	20	rw	**Port 1 Pin 4 Input Schmitt Trigger enable (only valid if IO is configured as output)** 0_B **0**, Schmitt Trigger is disabled 1_B **1**, Schmitt Trigger is enabled
RES	19	r	**Reserved** Returns 0 if read.
PP2_INEN	18	rw	**Port 1 Pin 2 Input Schmitt Trigger enable (only valid if IO is configured as output)** 0_B **0**, Schmitt Trigger is disabled 1_B **1**, Schmitt Trigger is enabled
PP1_INEN	17	rw	**Port 1 Pin 1 Input Schmitt Trigger enable (only valid if IO is configured as output)** 0_B **0**, Schmitt Trigger is disabled 1_B **1**, Schmitt Trigger is enabled
PP0_INEN	16	rw	**Port 1 Pin 0 Input Schmitt Trigger enable (only valid if IO is configured as output)** 0_B **0**, Schmitt Trigger is disabled 1_B **1**, Schmitt Trigger is enabled
RES	15:5	r	**Reserved** Returns 0 if read.
PP4	4	rw	**Port 1 Pin 4 Direction Control** 0_B **Input**, Direction is set to input 1_B **Output**, Direction is set to output
RES	3	r	**Reserved** Returns 0 if read.
PP2	2	rw	**Port 1 Pin 2 Direction Control** 0_B **Input**, Direction is set to input 1_B **Output**, Direction is set to output
PP1	1	rw	**Port 1 Pin 1 Direction Control** 0_B **Input**, Direction is set to input 1_B **Output**, Direction is set to output

Field	Bits	Type	Description
PP0	0	rw	**Port 1 Pin 0 Direction Control** 0_B **Input**, Direction is set to input 1_B **Output**, Direction is set to output

Table 240 RESET of P1_DIR

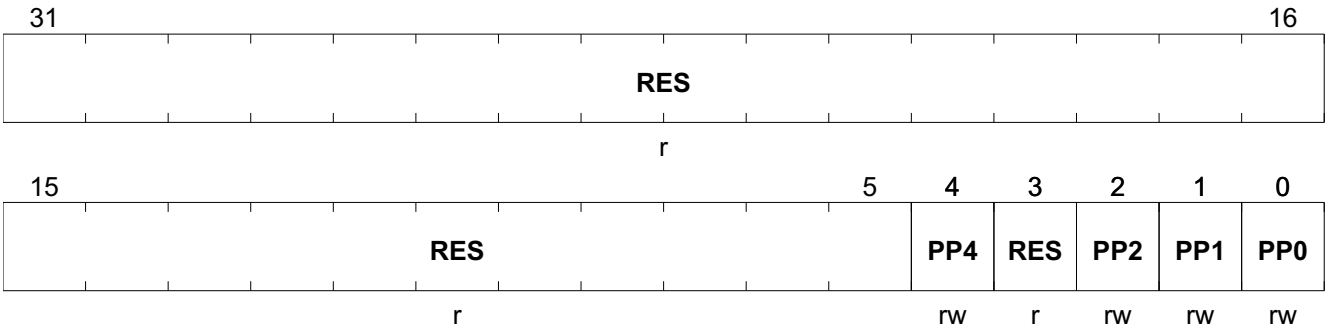
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Open Drain Control Register

P1_OD — **Port 1 Open Drain Control Register**

Offset 28_H

Reset Value see Table 241



Field	Bits	Type	Description
RES	31:5	r	**Reserved** Returns 0 if read.
PP4	4	rw	**Port 1 Pin 4 Open Drain Mode** 0_B **Normal Mode**, Output is actively driven for 0 and 1 state 1_B **Open Drain Mode**, Output is actively driven only for 0 state
RES	3	r	**Reserved** Returns 0 if read.
PP2	2	rw	**Port 1 Pin 2 Open Drain Mode** 0_B **Normal Mode**, Output is actively driven for 0 and 1 state 1_B **Open Drain Mode**, Output is actively driven only for 0 state
PP1	1	rw	**Port 1 Pin 1 Open Drain Mode** 0_B **Normal Mode**, Output is actively driven for 0 and 1 state 1_B **Open Drain Mode**, Output is actively driven only for 0 state

Field	Bits	Type	Description
PP0	0	rw	**Port 1 Pin 0 Open Drain Mode** 0_B **Normal Mode**, Output is actively driven for 0 and 1 state 1_B **Open Drain Mode**, Output is actively driven only for 0 state

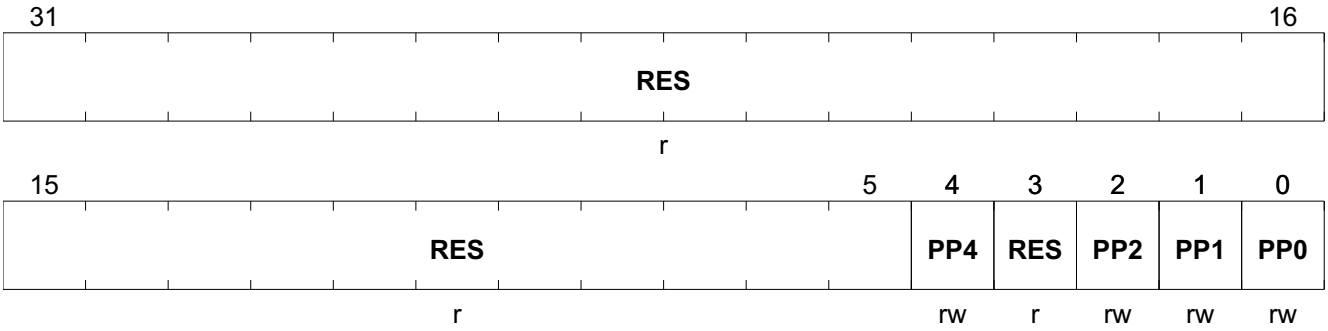
Table 241 RESET of P1_OD

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Pull-Up/Pull-Down Device Register

P1_PUDSEL — **Offset** — **Reset Value**

Port 1 Pull-Up/Pull-Down Select Register — **$2C_H$** — **see Table 242**



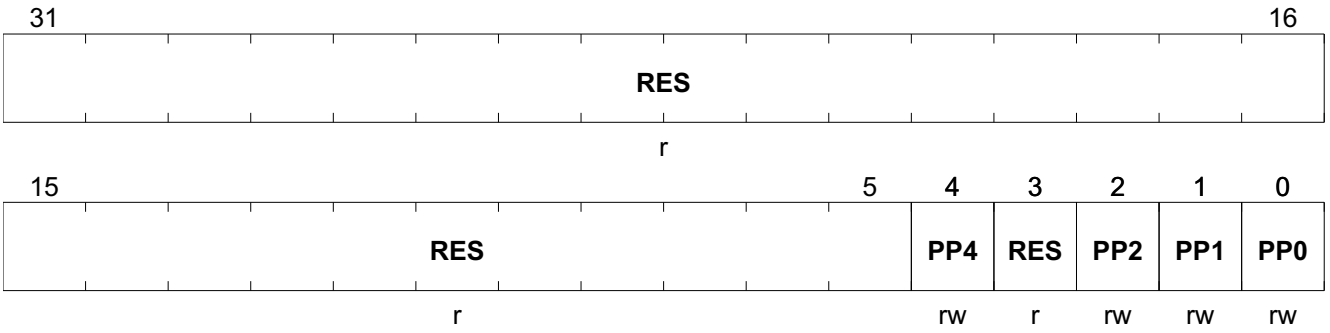
Field	Bits	Type	Description
RES	31:5	r	**Reserved** Returns 0 if read.
PP4	4	rw	**Pull-Up/Pull-Down Select Port 1 Bit 4** 0_B **Pull-down**, Pull-down device is selected 1_B **Pull-up**, Pull-up device is selected
RES	3	r	**Reserved** Returns 0 if read.
PP2	2	rw	**Pull-Up/Pull-Down Select Port 1 Bit 2** 0_B **Pull-down**, Pull-down device is selected 1_B **Pull-up**, Pull-up device is selected
PP1	1	rw	**Pull-Up/Pull-Down Select Port 1 Bit 1** 0_B **Pull-down**, Pull-down device is selected 1_B **Pull-up**, Pull-up device is selected
PP0	0	rw	**Pull-Up/Pull-Down Select Port 1 Bit 0** 0_B **Pull-down**, Pull-down device is selected 1_B **Pull-up**, Pull-up device is selected

Table 242 RESET of P1_PUDSEL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000017_H	RESET_TYPE_3		

Port 1 Pull-Up/Pull-Down Enable Register

P1_PUDEN	Offset	Reset Value
Port 1 Pull-Up/Pull-Down Enable Register	30_H	see Table 243



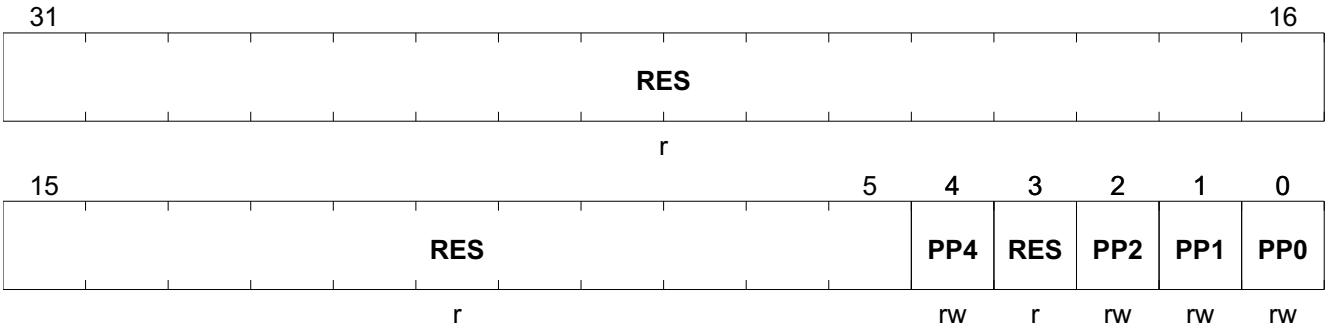
Field	Bits	Type	Description
RES	31:5	r	**Reserved** Returns 0 if read.
PP4	4	rw	**Pull-Up/Pull-Down Enable at Port 1 Bit 4** 0_B **Disabled**, Pull-up or Pull-down device is disabled 1_B **Enabled**, Pull-up or Pull-down device is enabled
RES	3	r	**Reserved** Returns 0 if read.
PP2	2	rw	**Pull-Up/Pull-Down Enable at Port 1 Bit 2** 0_B **Disabled**, Pull-up or Pull-down device is disabled 1_B **Enabled**, Pull-up or Pull-down device is enabled
PP1	1	rw	**Pull-Up/Pull-Down Enable at Port 1 Bit 1** 0_B **Disabled**, Pull-up or Pull-down device is disabled 1_B **Enabled**, Pull-up or Pull-down device is enabled
PP0	0	rw	**Pull-Up/Pull-Down Enable at Port 1 Bit 0** 0_B **Disabled**, Pull-up or Pull-down device is disabled 1_B **Enabled**, Pull-up or Pull-down device is enabled

Table 243 RESET of P1_PUDEN

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Alternate Output Select Register

P1_ALTSEL0	Offset	Reset Value
Port 1 Alternate Select Register 0	34_H	see Table 244



Field	Bits	Type	Description
RES	31:5	r	**Reserved** Returns 0 if read.
PP4	4	rw	**See Table 247**
RES	3	r	**Reserved** Returns 0 if read.
PP2	2	rw	**See Table 247**
PP1	1	rw	**See Table 247**
PP0	0	rw	**See Table 247**

Table 244 RESET of P1_ALTSEL0

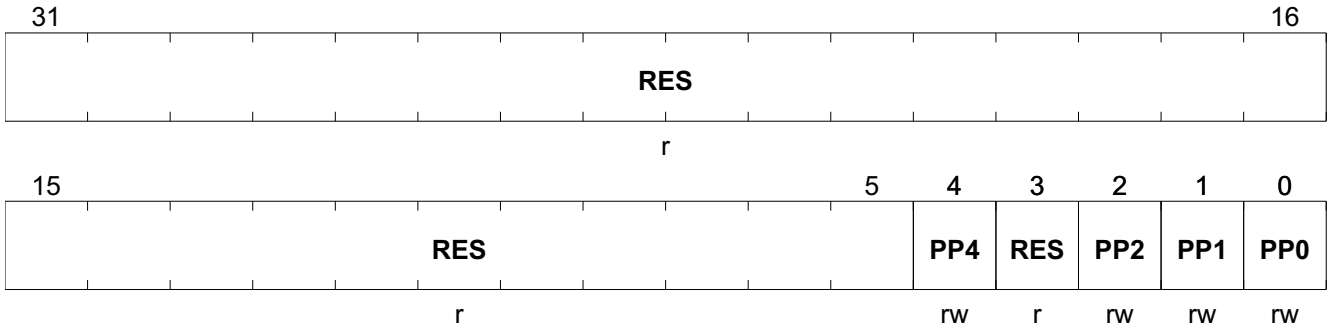
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Table 245 Function of Bits P1_ALTSEL0.PPn and P1_ALTSEL1.PPn

P1_ALTSEL0.PPn	P1_ALTSEL1.PPn	Function
0	0	Normal GPIO
1	0	Alternate Select 1
0	1	Alternate Select 2
1	1	Alternate Select 3

Alternate Output Select Register

P1_ALTSEL1	Offset	Reset Value
Port 1 Alternate Select Register 1	38_H	see Table 246



Field	Bits	Type	Description
RES	31:5	r	**Reserved** Returns 0 if read.
PP4	4	rw	**See Table 247**
RES	3	r	**Reserved** Returns 0 if read.
PP2	2	rw	**See Table 247**
PP1	1	rw	**See Table 247**
PP0	0	rw	**See Table 247**

Table 246 RESET of P1_ALTSEL1

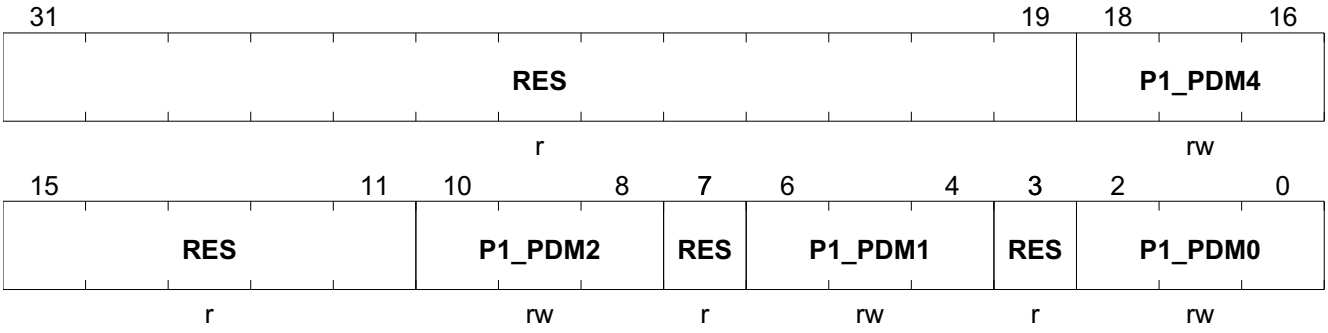
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Table 247 Function of Bits P1_ALTSEL0.PPn and P1_ALTSEL1.PPn

P1_ALTSEL0.PPn	P1_ALTSEL1.PPn	Function
0	0	Normal GPIO
1	0	Alternate Select 1
0	1	Alternate Select 2
1	1	Alternate Select 3

Port Output Control Register

SCU_P1_POCON0	Offset	Reset Value
Port Output Control Register	$0F8_H$	see Table 248



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Returns 0 if read; should be written with 0.
P1_PDM4	18:16	rw	**P1.4 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong driver and sharp edge mode**, 001_B **Strong driver and medium edge mode**, 010_B **Strong driver and soft edge mode**, 011_B **Weak Driver**, 100_B **Medium Driver**, 101_B **Medium Driver**, 110_B **Medium Driver**, 111_B **Weak Driver**,
RES	15:11	r	**Reserved** Returns 0 if read; should be written with 0.
P1_PDM2	10:8	rw	**P1.2 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong driver and sharp edge mode**, 001_B **Strong driver and medium edge mode**, 010_B **Strong driver and soft edge mode**, 011_B **Weak Driver**, 100_B **Medium Driver**, 101_B **Medium Driver**, 110_B **Medium Driver**, 111_B **Weak Driver**,
RES	7	r	**Reserved** Returns 0 if read; should be written with 0.

Field	Bits	Type	Description
P1_PDM1	6:4	rw	**P1.1 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong driver and sharp edge mode**, 001_B **Strong driver and medium edge mode**, 010_B **Strong driver and soft edge mode**, 011_B **Weak Driver**, 100_B **Medium Driver**, 101_B **Medium Driver**, 110_B **Medium Driver**, 111_B **Weak Driver**,
RES	3	r	**Reserved** Returns 0 if read; should be written with 0.
P1_PDM0	2:0	rw	**P1.0 Port Driver Mode** Code Driver Strength[1] and Edge Shape[2] 000_B **Strong driver and sharp edge mode**, 001_B **Strong driver and medium edge mode**, 010_B **Strong driver and soft edge mode**, 011_B **Weak Driver**, 100_B **Medium Driver**, 101_B **Medium Driver**, 110_B **Medium Driver**, 111_B **Weak Driver**,

1) Defines the current the respective driver can deliver to the external circuitry.

2) Defines the switching characteristics to the respective new output driver. This also influences the peak currents through the driver when producing an edge, i.e. when changing the output level.

Table 248 RESET of SCU_P1_POCON0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

15.4.3 Port 2 Register Description

Table 249 Register Overview

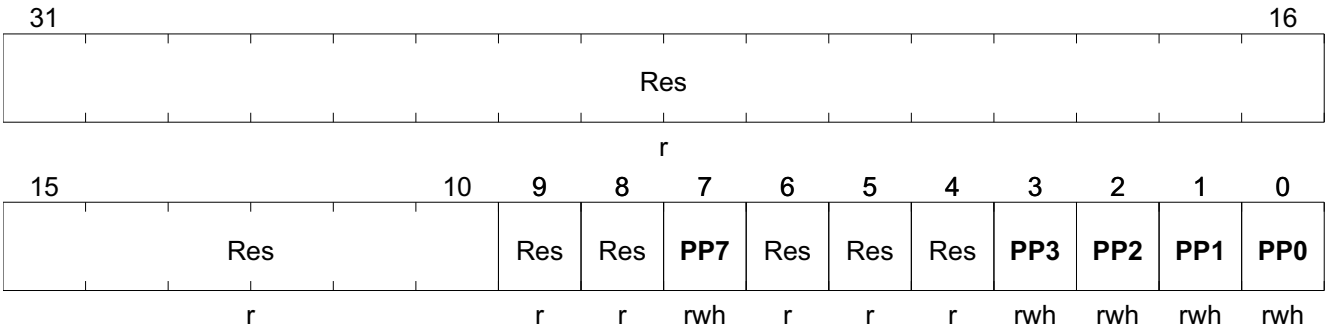
Register Short Name	Register Long Name	Offset Address	Reset Value
Port 2 Register Description,			
P2_DATA	Port 2 Data Register	40_H	$000000XX_H$
P2_DIR	Port 2 Direction Register	44_H	00000000_H
P2_PUDSEL	Port 2 Pull-Up/Pull-Down Select Register	$4C_H$	00000000_H
P2_PUDEN	Port 2 Pull-Up/Pull-Down Enable Register	50_H	00000000_H

The registers are addressed wordwise.

Data Register

P2_DATA **Offset** **Reset Value**

Port 2 Data Register **40_H** **see Table 250**



Field	Bits	Type	Description
Res	31:10	r	**Reserved** Always read as 0
Res	9	r	**Reserved** Always read as 0
Res	8	r	**Reserved** Always read as 0
PP7	7	rwh	**Port 2 Pin 7 Data Value** 0_B **0**, Port 2 pin 7 data value = 0 1_B **1**, Port 2 pin 7 data value = 1
Res	6	r	**Reserved** Always read as 0

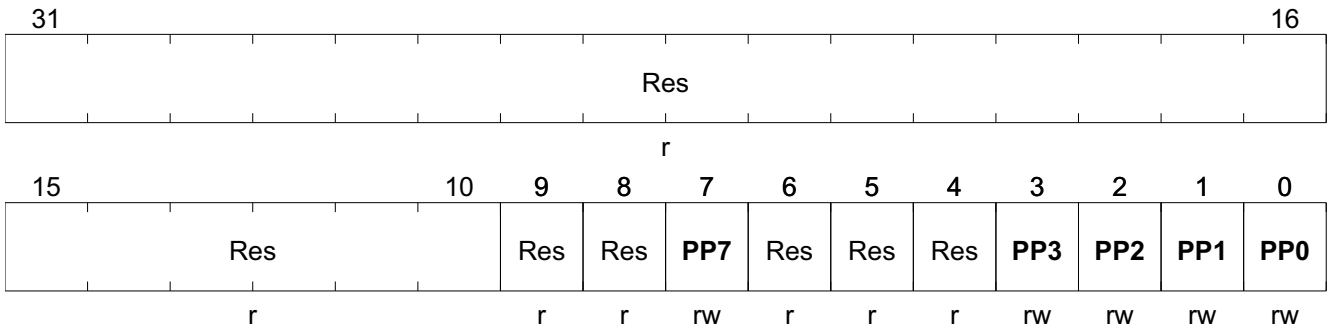
Field	Bits	Type	Description
Res	5	r	**Reserved** Always read as 0
Res	4	r	**Reserved** Always read as 0
PP3	3	rwh	**Port 2 Pin 3 Data Value** 0_B **0**, Port 2 pin 3 data value = 0 1_B **1**, Port 2 pin 3 data value = 1
PP2	2	rwh	**Port 2 Pin 2 Data Value** 0_B **0**, Port 2 pin 2 data value = 0 1_B **1**, Port 2 pin 2 data value = 1
PP1	1	rwh	**Port 2 Pin 1 Data Value** 0_B **0**, Port 2 pin 1 data value = 0 1_B **1**, Port 2 pin 1 data value = 1
PP0	0	rwh	**Port 2 Pin 0 Data Value** 0_B **0**, Port 2 pin 0 data value = 0 1_B **1**, Port 2 pin 0 data value = 1

Table 250 RESET of P2_DATA

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$000000XX_H$	RESET_TYPE_3		

Direction Register

P2_DIR	Offset	Reset Value
Port 2 Direction Register	**44_H**	**see Table 251**



Field	Bits	Type	Description
Res	31:10	r	**Reserved** Always read as 0
Res	9	r	**Reserved** Always read as 0
Res	8	r	**Reserved** Always read as 0

Field	Bits	Type	Description
PP7	7	rw	**Port 2 Pin 7 Driver Control** 0_B **Enabled**, Input driver is enabled 1_B **Disabled**, Input driver is disabled
Res	6	r	**Reserved** Always read as 0
Res	5	r	**Reserved** Always read as 0
Res	4	r	**Reserved** Always read as 0
PP3	3	rw	**Port 2 Pin 3 Driver Control** 0_B **Enabled**, Input driver is enabled 1_B **Disabled**, Input driver is disabled
PP2	2	rw	**Port 2 Pin 2 Driver Control** 0_B **Enabled**, Input driver is enabled 1_B **Disabled**, Input driver is disabled
PP1	1	rw	**Port 2 Pin 1 Driver Control** 0_B **Enabled**, Input driver is enabled 1_B **Disabled**, Input driver is disabled
PP0	0	rw	**Port 2 Pin 0 Driver Control** 0_B **Enabled**, Input driver is enabled 1_B **Disabled**, Input driver is disabled

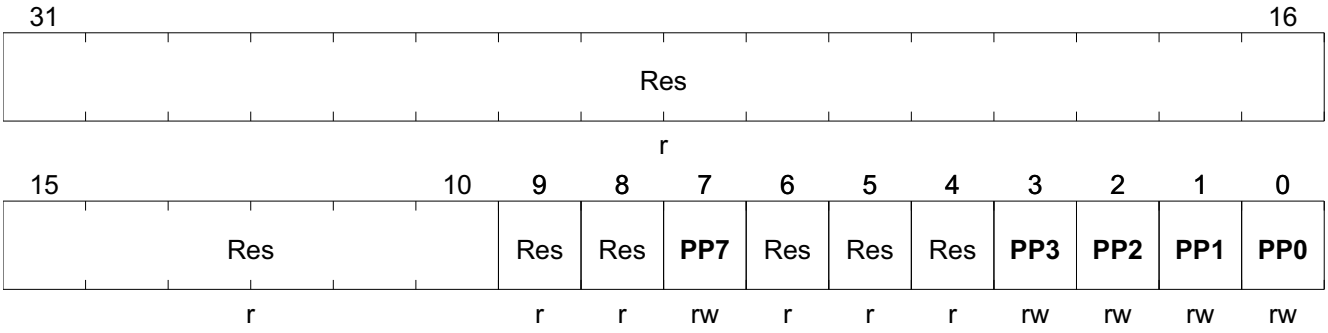
Table 251 RESET of P2_DIR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Pull-Up/Pull-Down Device Register

P2_PUDSEL — **Offset** — **Reset Value**

Port 2 Pull-Up/Pull-Down Select Register — **$4C_H$** — **see Table 252**



Field	Bits	Type	Description
Res	31:10	r	**Reserved** Always read as 0

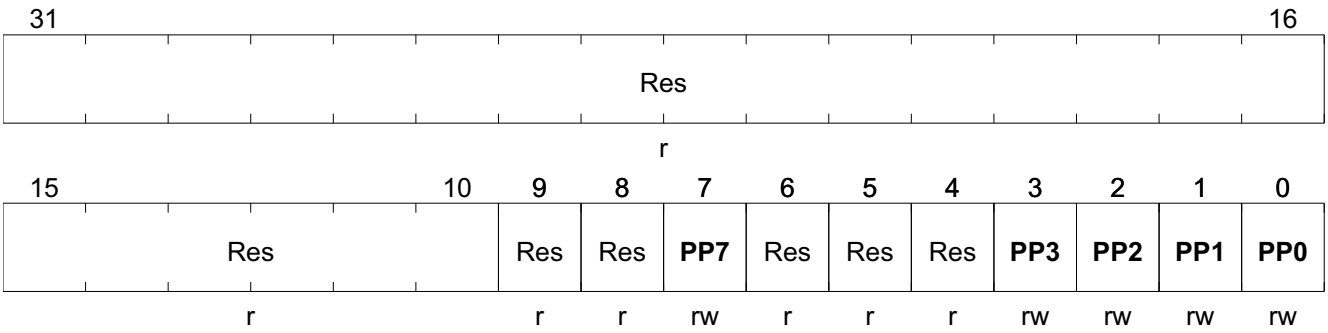
Field	Bits	Type	Description
Res	9	r	**Reserved** Always read as 0
Res	8	r	**Reserved** Always read as 0
PP7	7	rw	**Pull-Up/Pull-Down Select Port 2 Bit 7** 0_B **Pull-down**, Pull-down device is selected 1_B **Pull-up**, Pull-up device is selected
Res	6	r	**Reserved** Always read as 0
Res	5	r	**Reserved** Always read as 0
Res	4	r	**Reserved** Always read as 0
PP3	3	rw	**Pull-Up/Pull-Down Select Port 2 Bit 3** 0_B **Pull-down**, Pull-down device is selected 1_B **Pull-up**, Pull-up device is selected
PP2	2	rw	**Pull-Up/Pull-Down Select Port 2 Bit 2** 0_B **Pull-down**, Pull-down device is selected 1_B **Pull-up**, Pull-up device is selected
PP1	1	rw	**Pull-Up/Pull-Down Select Port 2 Bit 1** 0_B **Pull-down**, Pull-down device is selected 1_B **Pull-up**, Pull-up device is selected
PP0	0	rw	**Pull-Up/Pull-Down Select Port 2 Bit 0** 0_B **Pull-down**, Pull-down device is selected 1_B **Pull-up**, Pull-up device is selected

Table 252 RESET of P2_PUDSEL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Pull-Up/Pull-Down Enable Register

P2_PUDEN	Offset	Reset Value
Port 2 Pull-Up/Pull-Down Enable Register	50_H	see Table 253



Field	Bits	Type	Description
Res	31:10	r	**Reserved** Always read as 0
Res	9	r	**Reserved** Always read as 0
Res	8	r	**Reserved** Always read as 0
PP7	7	rw	**Pull-Up/Pull-Down Enable at Port 2 Bit 7** 0_B **Disabled**, Pull-up or Pull-down device is disabled 1_B **Enabled**, Pull-up or Pull-down device is enabled
Res	6	r	**Reserved** Always read as 0
Res	5	r	**Reserved** Always read as 0
Res	4	r	**Reserved** Always read as 0
PP3	3	rw	**Pull-Up/Pull-Down Enable at Port 2 Bit 3** 0_B **Disabled**, Pull-up or Pull-down device is disabled 1_B **Enabled**, Pull-up or Pull-down device is enabled
PP2	2	rw	**Pull-Up/Pull-Down Enable at Port 2 Bit 2** 0_B **Disabled**, Pull-up or Pull-down device is disabled 1_B **Enabled**, Pull-up or Pull-down device is enabled
PP1	1	rw	**Pull-Up/Pull-Down Enable at Port 2 Bit 1** 0_B **Disabled**, Pull-up or Pull-down device is disabled 1_B **Enabled**, Pull-up or Pull-down device is enabled
PP0	0	rw	**Pull-Up/Pull-Down Enable at Port 2 Bit 0** 0_B **Disabled**, Pull-up or Pull-down device is disabled 1_B **Enabled**, Pull-up or Pull-down device is enabled

Table 253 RESET of P2_PUDEN

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

16 General Purpose Timer Units (GPT12)

16.1 Features

16.1.1 Features Block GPT1

The following list summarizes the supported features:

- f_{GPT}/4 maximum resolution
- 3 independent timers/counters
- Timers/counters can be concatenated
- 4 operating modes:
  - Timer Mode
  - Gated Timer Mode
  - Counter Mode
  - Incremental Interface Mode
- Reload and Capture functionality
- Shared interrupt: Node 0

16.1.2 Features Block GPT2

The following list summarizes the supported features:

- f_{GPT}/2 maximum resolution
- 2 independent timers/counters
- Timers/counters can be concatenated
- 3 operating modes:
  - Timer Mode
  - Gated Timer Mode
  - Counter Mode
- Extended capture/reload functions via 16-bit capture/reload register CAPREL
- Shared interrupt: Node 1

16.2 Introduction

The General Purpose Timer Unit blocks GPT1 and GPT2 have very flexible multifunctional timer structures which may be used for timing, event counting, pulse width measurement, pulse generation, frequency multiplication, and other purposes.

They incorporate five 16-bit timers that are grouped into the two timer blocks GPT1 and GPT2. Each timer in each block may operate independently in a number of different modes such as Gated timer or Counter Mode, or may be concatenated with another timer of the same block.

Each block has alternate input/output functions and specific interrupts associated with it. Input signals can be selected from several sources by register PISEL.

The GPT module is clocked with clock f_{GPT}. f_{GPT} is a clock derived from f_{SYS}.

16.2.1 Block Diagram GPT1

Block GPT1 contains three timers/counters: The core timer T3 and the two auxiliary timers T2 and T4. The maximum resolution is $f_{GPT}/4$. The auxiliary timers of GPT1 may optionally be configured as reload or capture registers for the core timer. These registers are listed in **Section 16.3.8.1**.

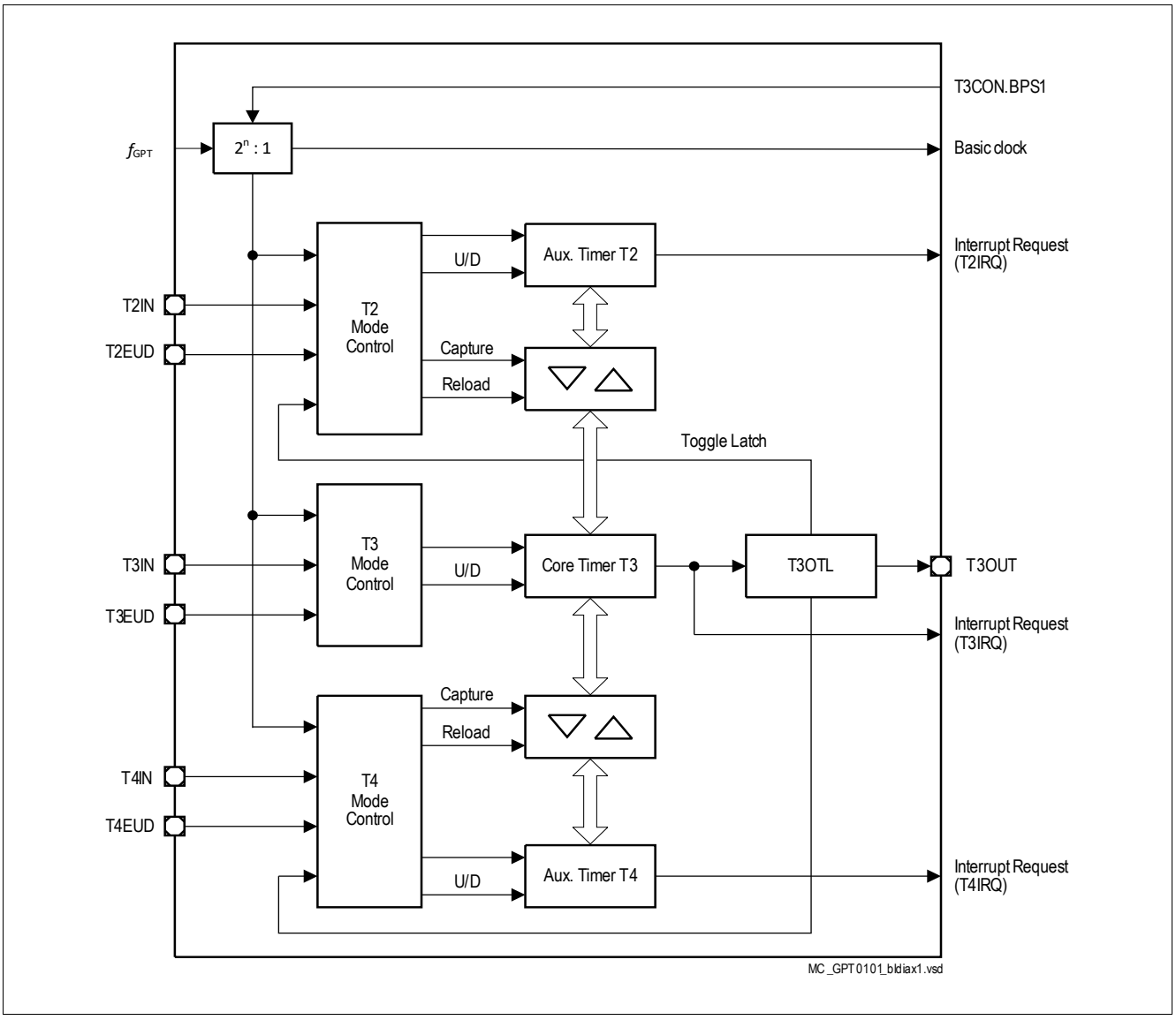


Figure 88 GPT1 Block Diagram (n = 2 … 5)

16.2.2 Block Diagram GPT2

Block GPT2 contains two timers/counters: The core timer T6 and the auxiliary timer T5. The maximum resolution is $f_{GPT}/2$. An additional Capture/Reload register (CAPREL) supports capture and reload operation with extended functionality. These registers are listed in **Section 16.4.8.1**.

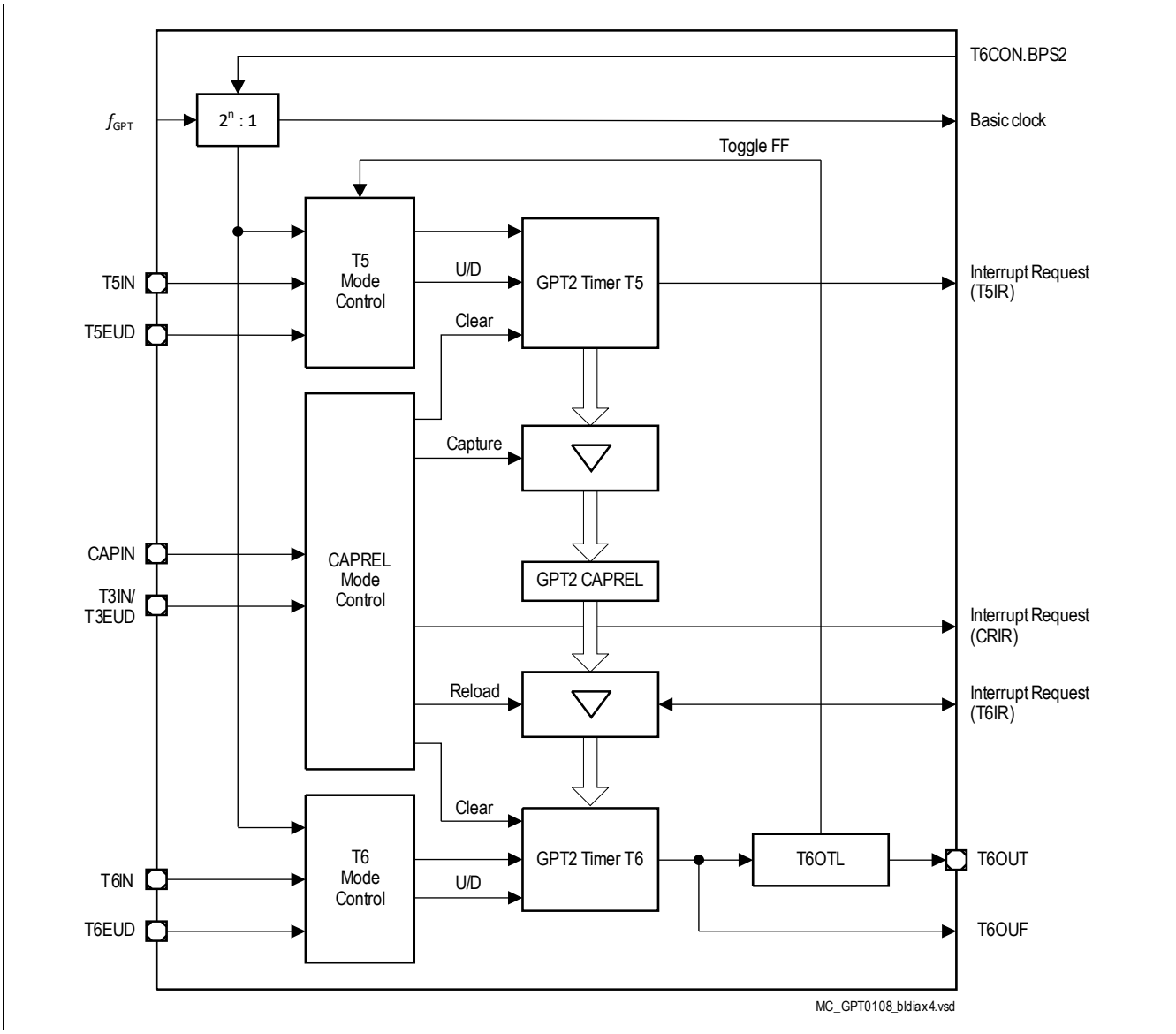


Figure 89 GPT2 Block Diagram

16.3 Timer Block GPT1

From a programmer's point of view, the GPT1 block is composed of a set of SFRs as summarized below. Those portions of port and direction registers which are used for alternate functions by the GPT1 block are shaded.

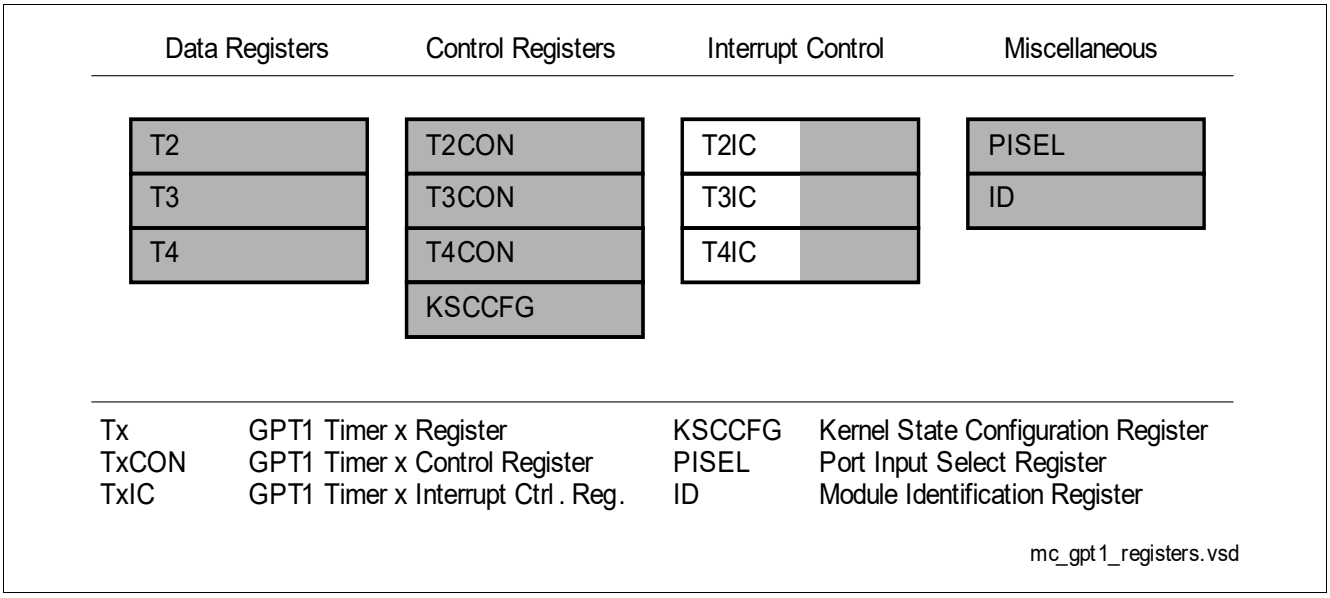


Figure 90 SFRs Associated with Timer Block GPT1

All three timers of block GPT1 (T2, T3, T4) can run in one of 4 basic modes: Timer Mode, Gated Timer Mode, Counter Mode, or Incremental Interface Mode. All timers can count up or down. Each timer of GPT1 is controlled by a separate control register TxCON.

Each timer has an input pin TxIN (alternate pin function) associated with it, which serves as the gate control in Gated Timer Mode, or as the count input in Counter Mode. The count direction (up/down) may be programmed via software or may be dynamically altered by a signal at the External Up/Down control input TxEUD (alternate pin function). An overflow/underflow of core timer T3 is indicated by the Output Toggle Latch T3OTL, whose state may be output on the associated pin T3OUT (alternate pin function). The auxiliary timers T2 and T4 may additionally be concatenated with the core timer T3 (through T3OTL) or may be used as capture or reload registers for the core timer T3.

The current contents of each timer can be read or modified by the CPU by accessing the corresponding timer count registers T2, T3, or T4, located in the non-bitaddressable SFR space (see **Section 16.3.8.1**). When any of the timer registers is written to by the CPU in the state immediately preceding a timer increment, decrement, reload, or capture operation, the CPU write operation has priority in order to guarantee correct results.

The interrupts of GPT1 are controlled through the GPTM1IEN and GPTM1IRC. These registers are not part of the GPT1 block.

The input and output lines of GPT1 are connected to pins. The control registers for the port functions are located in the respective port modules.

*Note: The timing requirements for external input signals can be found in **Section 16.3.5**, **Section 16.6.1** summarizes the module interface signals, including pins.*

16.3.1 GPT1 Core Timer T3 Control

The current contents of the core timer T3 are reflected by its count register T3. This register can also be written to by the CPU, for example, to set the initial start value.

The core timer T3 is configured and controlled via its control register T3CON.

Timer T3 Run Control

The core timer T3 can be started or stopped by software through bit T3R (Timer T3 Run Bit). This bit is relevant in all operating modes of T3. Setting bit T3R will start the timer, clearing bit T3R stops the timer.

In Gated Timer Mode, the timer will only run if T3R = 1 and the gate is active (high or low, as programmed).

Note: When bit T2RC or T4RC in timer control register T2CON or T4CON is set, bit T3R will also control (start and stop) the auxiliary timer(s) T2 and/or T4.

Count Direction Control

The count direction of the GPT1 timers (core timer and auxiliary timers) can be controlled either by software or by the external input pin TxEUD (Timer Tx External Up/Down Control Input). These options are selected by bits TxUD and TxUDE in the respective control register TxCON. When the up/down control is provided by software (bit TxUDE = 0), the count direction can be altered by setting or clearing bit TxUD. When bit TxUDE = 1, pin TxEUD is selected to be the controlling source of the count direction. However, bit TxUD can still be used to reverse the actual count direction, as shown in **Table 266**. The count direction can be changed regardless of whether or not the timer is running.

Note: When pin TxEUD is used as external count direction control input, it must be configured as input.

Timer T3 Output Toggle Latch

The overflow/underflow signal of timer T3 is connected to a block named ‘Toggle Latch’, shown in the Timer Mode diagrams. **Figure 91** illustrates the details of this block. An overflow or underflow of T3 will clock two latches: The first latch represents bit T3OTL in control register T3CON. The second latch is an internal latch toggled by T3OTL’s output. Both latch outputs are connected to the input control blocks of the auxiliary timers T2 and T4. The output level of the shadow latch will match the output level of T3OTL, but is delayed by one clock cycle. When the T3OTL value changes, this will result in a temporarily different output level from T3OTL and the shadow latch, which can trigger the selected count event in T2 and/or T4.

When software writes to T3OTL, both latches are set or cleared simultaneously. In this case, both signals to the auxiliary timers carry the same level and no edge will be detected. Bit T3OE (overflow/underflow output enable) in register T3CON enables the state of T3OTL to be monitored via an external pin T3OUT. When T3OTL is linked to an external port pin (must be configured as output), T3OUT can be used to control external HW. If T3OE = 1, pin T3OUT outputs the state of T3OTL. If T3OE = 0, pin T3OUT outputs a high level (as long as the T3OUT alternate function is selected for the port pin).

The trigger signals can serve as an input for the counter function or as a trigger source for the reload function of the auxiliary timers T2 and T4.

As can be seen from **Figure 91**, when latch T3OTL is modified by software to determine the state of the output line, also the internal shadow latch is set or cleared accordingly. Therefore, no trigger condition is detected by T2/T4 in this case.

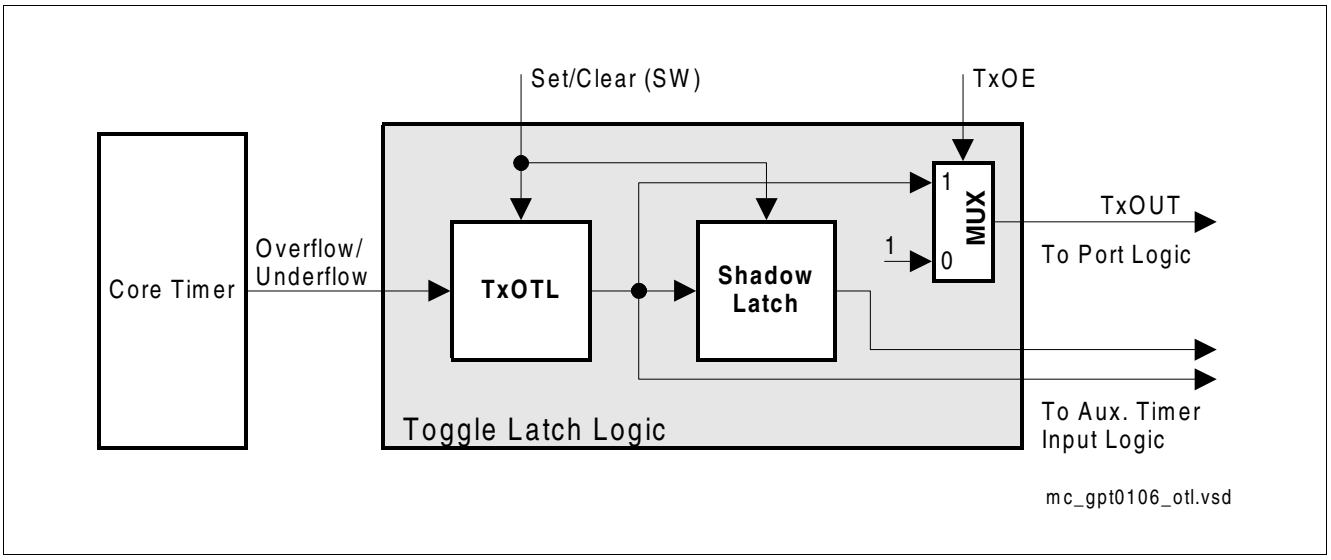


Figure 91 Block Diagram of the Toggle Latch Logic of Core Timer T3 (x = 3)

16.3.2 GPT1 Core Timer T3 Operating Modes

Timer T3 can operate in one of several modes.

Timer T3 in Timer Mode

Timer mode for the core timer T3 is selected by setting bitfield T3M in register T3CON to 000_B. In Timer Mode, T3 is clocked with the module's input clock f_{GPT} divided by two programmable prescalers controlled by bitfields BPS1 and T3I in register T3CON. Please see **Section 16.3.5** for details on the input clock options.

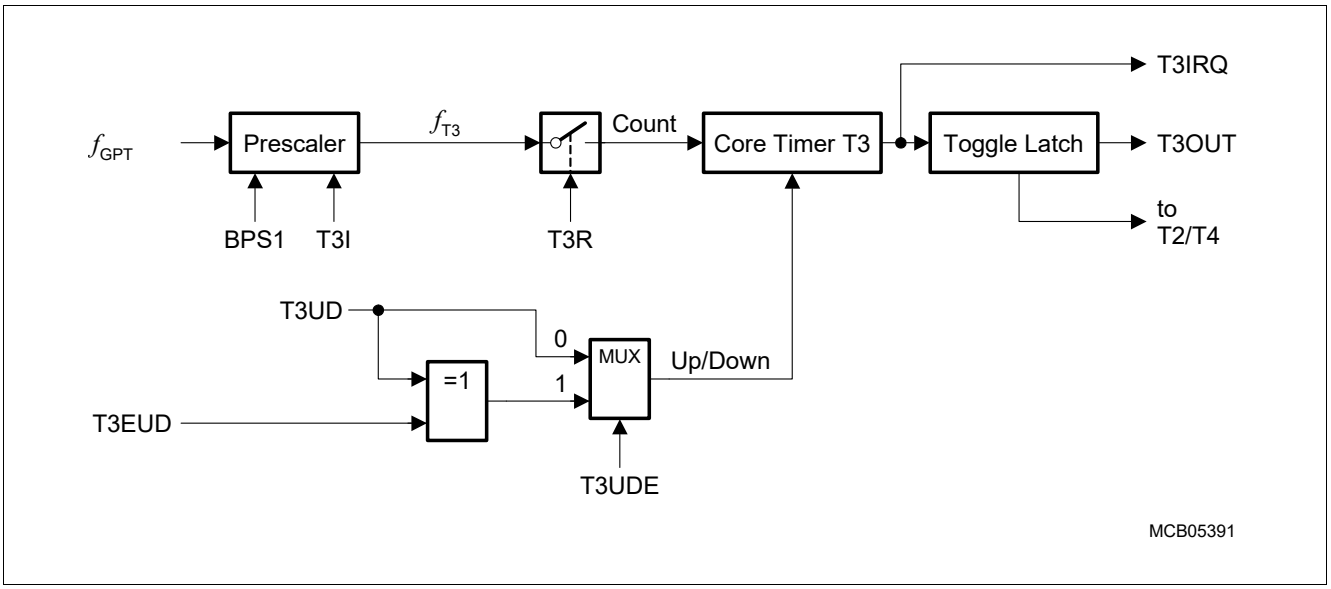


Figure 92 Block Diagram of Core Timer T3 in Timer Mode

Timer T3 in Gated Timer Mode

Gated Timer Mode for the core timer T3 is selected by setting bitfield T3M in register T3CON to 010_B or 011_B. Bit T3M.0 (T3CON.3) selects the active level of the gate input. The same options for the input frequency are available in Gated Timer Mode as in Timer Mode (see **Section 16.3.5**). However, the input clock to the timer in this mode is gated by the external input pin T3IN (Timer T3 External Input).
To enable this operation, the associated pin T3IN must be configured as input.

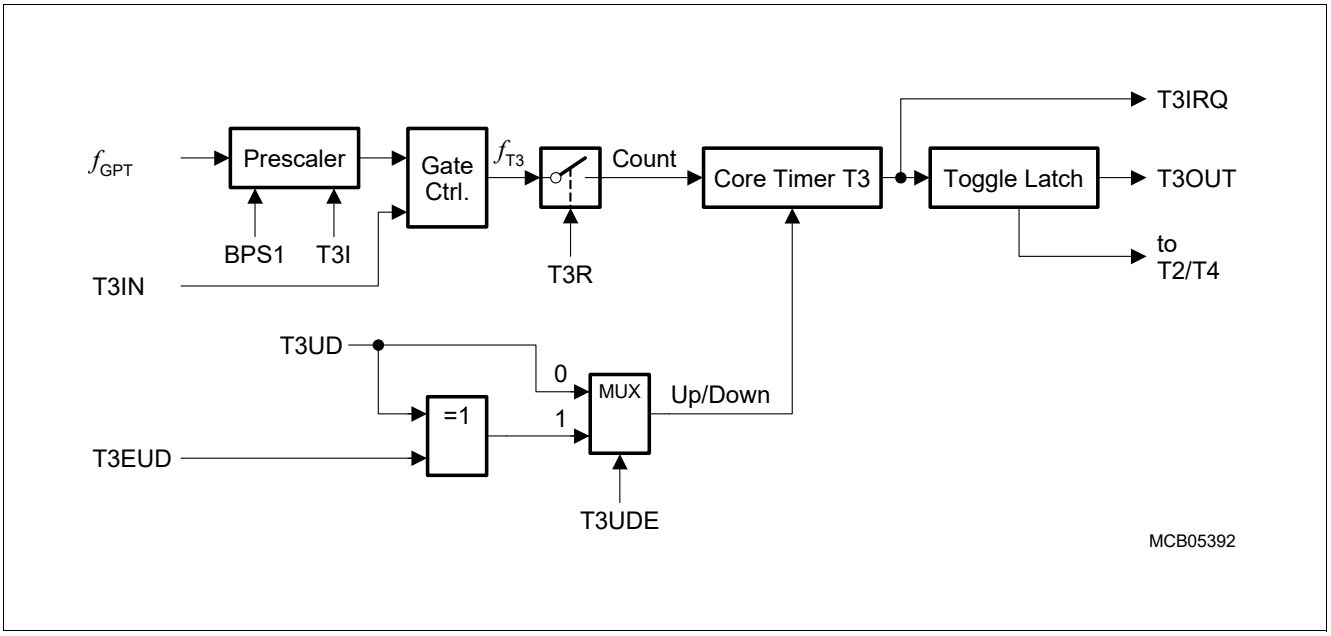


Figure 93 Block Diagram of Core Timer T3 in Gated Timer Mode

If T3M = 010_B, the timer is enabled when T3IN shows a low level. A high level at this line stops the timer. If T3M = 011_B, line T3IN must have a high level in order to enable the timer. Additionally, the timer can be turned on or off by software using bit T3R. The timer will only run if T3R is 1 and the gate is active. It will stop if either T3R is 0 or the gate is inactive.

Note: A transition of the gate signal at pin T3IN does not cause an interrupt request.

Timer T3 in Counter Mode

Counter Mode for the core timer T3 is selected by setting bitfield T3M in register T3CON to 001_B. In Counter Mode, timer T3 is clocked by a transition at the external input pin T3IN. The event causing an increment or decrement of the timer can be a positive, a negative, or both a positive and a negative transition at this line. Bitfield T3I in control register T3CON selects the triggering transition (see **Table 268**).

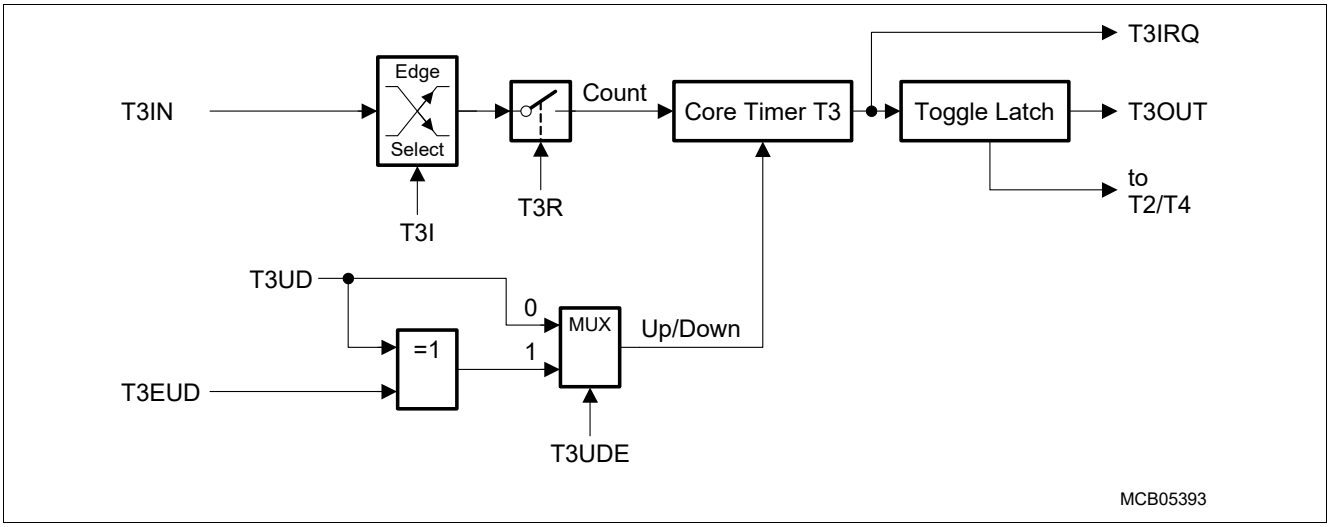


Figure 94 Block Diagram of Core Timer T3 in Counter Mode

For Counter Mode operation, pin T3IN must be configured as input. The maximum input frequency allowed in Counter Mode depends on the selected prescaler value. To ensure that a transition of the count input signal applied to T3IN is recognized correctly, its level must be held high or low for a minimum number of module clock cycles before it changes. This information can be found in **Section 16.3.5**.

Timer T3 in Incremental Interface Mode

Incremental interface mode for the core timer T3 is selected by setting bitfield T3M in register T3CON to 110_B or 111_B. In Incremental Interface Mode, the two inputs associated with core timer T3 (T3IN, T3EUD) are used to interface to an incremental encoder. T3 is clocked by each transition on one or both of the external input pins to provide 2-fold or 4-fold resolution of the encoder input.

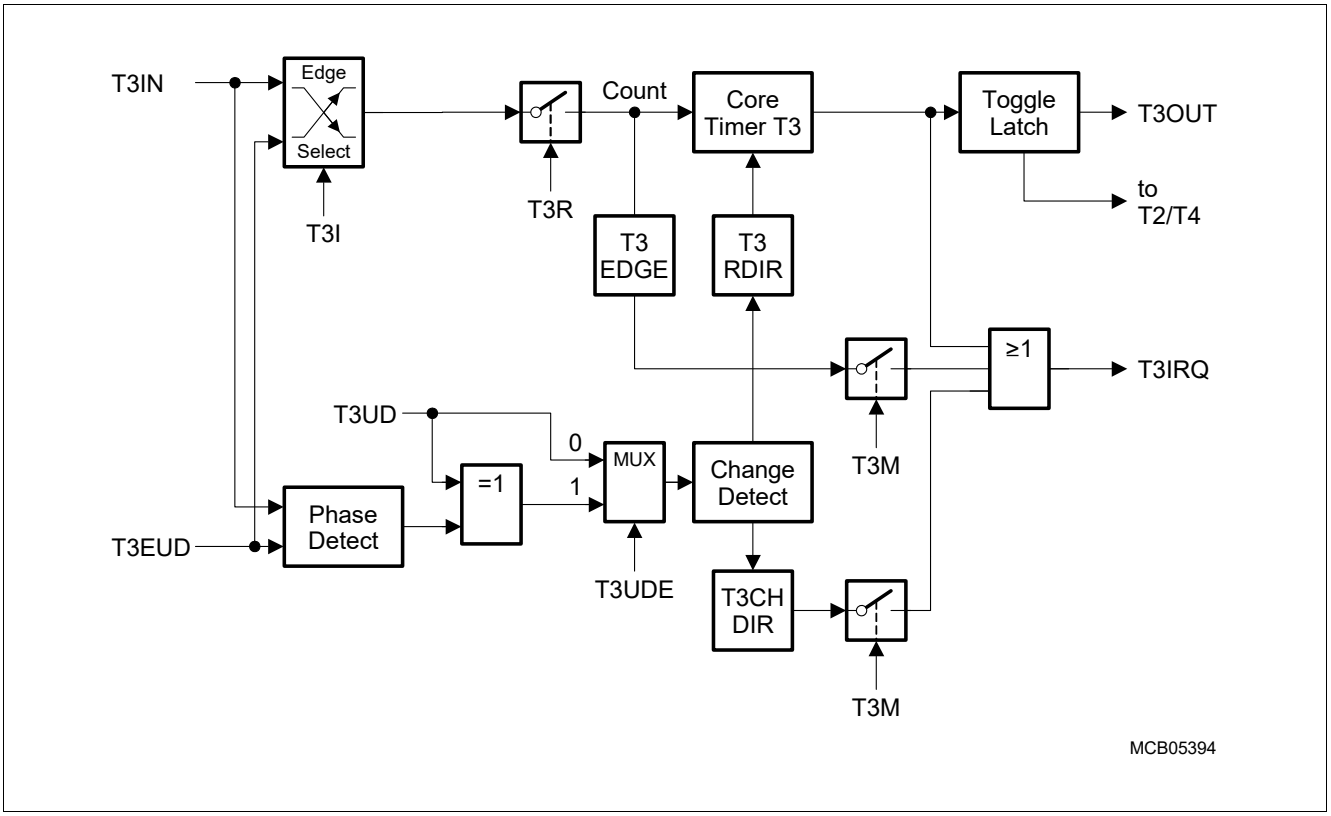


Figure 95 Block Diagram of Core Timer T3 in Incremental Interface Mode

Bitfield T3I in control register T3CON selects the triggering transitions (see **Table 270**). The sequence of the transitions of the two input signals is evaluated and generates count pulses as well as the direction signal. So T3 is modified automatically according to the speed and the direction of the incremental encoder and, therefore, its contents always represent the encoder's current position.

The interrupt request (T3IRQ) generation mode can be selected: In Rotation Detection Mode (T3M = 110_B), an interrupt request is generated each time the count direction of T3 changes. In Edge Detection Mode (T3M = 111_B), an interrupt request is generated each time a count edge for T3 is detected. Count direction, changes in the count direction, and count requests are monitored by status bits T3RDIR, T3CHDIR, and T3EDGE in register T3CON.

The incremental encoder can be connected directly to the TLE985xQX without external interface logic. In a standard system, however, comparators will be employed to convert the encoder's differential outputs (such as A, $\overline{A}$) to digital signals (such as A). This greatly increases noise immunity.

Note: The third encoder output T0, which indicates the mechanical zero position, may be connected to an external interrupt input and trigger a reset of timer T3.
If input T4IN is available, T0 can be connected there and clear T3 automatically without requiring an interrupt.

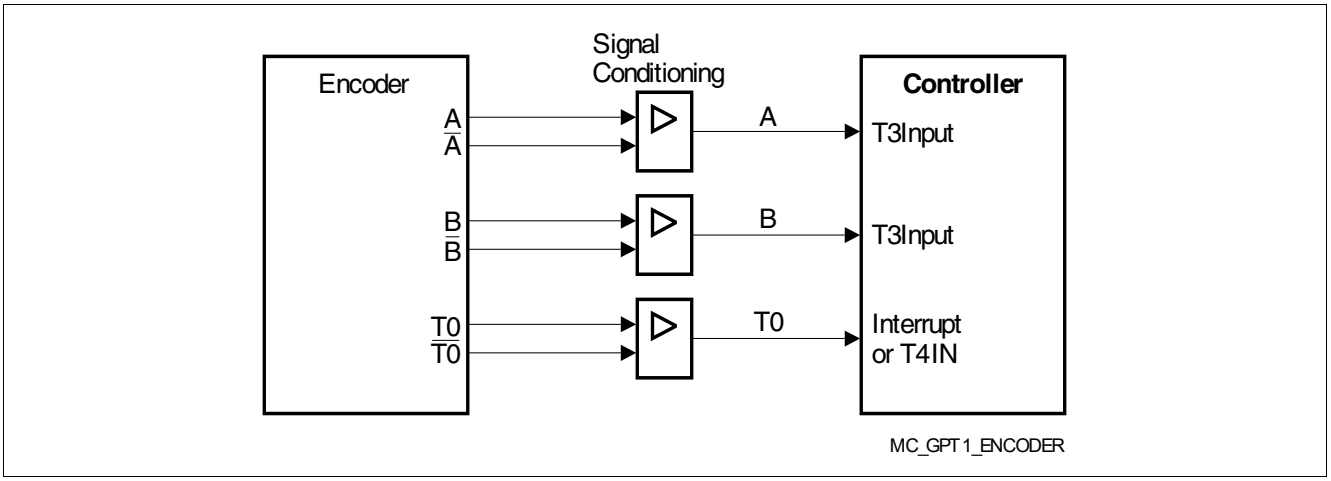


Figure 96 Connection of the Encoder to the TLE985xQX

For incremental interface operation, the following conditions must be met:

- Bitfield T3M must be 110_B or 111_B.
- Both pins T3IN and T3EUD must be configured as input.
- Pin T4IN must be configured as input, if used for T0.
- Bit T3UDE must be 1 to enable automatic external direction control.

The maximum count frequency allowed in Incremental Interface Mode depends on the selected prescaler value. To ensure that a transition of any input signal is recognized correctly, its level must be held high or low for a minimum number of module clock cycles before it changes. This information can be found in **Section 16.3.5**.

As in Incremental Interface Mode two input signals with a 90° phase shift are evaluated, their maximum input frequency can be half the maximum count frequency.

In Incremental Interface Mode, the count direction is automatically derived from the sequence in which the input signals change, which corresponds to the rotation direction of the connected sensor. **Table 254** summarizes the possible combinations.

Table 254 GPT1 Core Timer T3 (Incremental Interface Mode) Count Direction

Level on Respective other Input	T3IN Input		T3EUD Input	
	Rising ↑	**Falling ↓**	**Rising ↑**	**Falling ↓**
High	Down	Up	Up	Down
Low	Up	Down	Down	Up

Figure 97 and **Figure 98** give examples of T3's operation, visualizing count signal generation and direction control. They also show how input jitter is compensated, which might occur if the sensor rests near to one of its switching points.

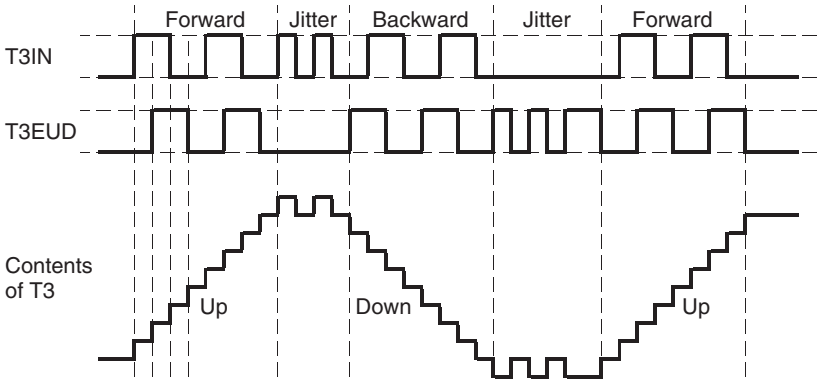


Figure 97 Evaluation of Incremental Encoder Signals, 2 Count Inputs

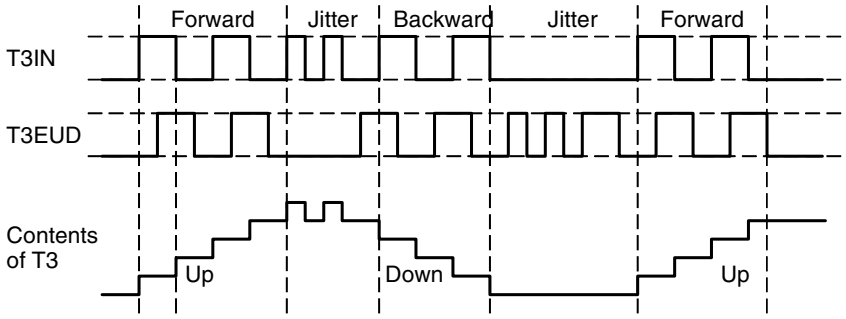


Figure 98 Evaluation of Incremental Encoder Signals, 1 Count Input

Note: Timer T3 operating in Incremental Interface Mode automatically provides information on the sensor's current position. Dynamic information (speed, acceleration, deceleration) may be obtained by measuring the incoming signal periods (see ***"Combined Capture Modes" on Page 518****).*

16.3.3 GPT1 Auxiliary Timers T2/T4 Control

Auxiliary timers T2 and T4 have exactly the same functionality. They can be configured for Timer Mode, Gated Timer Mode, Counter Mode, or Incremental Interface Mode with the same options for the timer frequencies and the count signal as the core timer T3. In addition to these 4 counting modes, the auxiliary timers can be concatenated with the core timer, or they may be used as reload or capture registers in conjunction with the core timer. The start/stop function of the auxiliary timers can be remotely controlled by the T3 run control bit. Several timers may thus be controlled synchronously.

The current contents of an auxiliary timer are reflected by its count register T2 or T4, respectively. These registers can also be written to by the CPU, for example, to set the initial start value.

The individual configurations for timers T2 and T4 are determined by their control registers T2CON and T4CON, which are organized identically. Note that functions which are present in all 3 timers of block GPT1 are controlled in the same bit positions and in the same manner in each of the specific control registers.

Note: The auxiliary timers have no output toggle latch and no alternate output function.

Timer T2/T4 Run Control

Each of the auxiliary timers T2 and T4 can be started or stopped by software in two different ways:

- Through the associated timer run bit (T2R or T4R). In this case it is required that the respective control bit TxRC = 0.
- Through the core timer's run bit (T3R). In this case the respective remote control bit must be set (TxRC = 1).

The selected run bit is relevant in all operating modes of T2/T4. Setting the bit will start the timer, clearing the bit stops the timer.

In Gated Timer Mode, the timer will only run if the selected run bit is set and the gate is active (high or low, as programmed).

Note: If remote control is selected T3R will start/stop timer T3 and the selected auxiliary timer(s) synchronously.

Count Direction Control

The count direction of the GPT1 timers (core timer and auxiliary timers) is controlled in the same way, either by software or by the external input pin TxEUD. Please refer to the description in **Table 266**.

Note: When pin TxEUD is used as external count direction control input, it must be configured as input.

16.3.4 GPT1 Auxiliary Timers T2/T4 Operating Modes

The operation of the auxiliary timers in the basic operating modes is almost identical with the core timer's operation, with very few exceptions. Additionally, some combined operating modes can be selected.

Timers T2 and T4 in Timer Mode

Timer mode for an auxiliary timer Tx is selected by setting its bitfield TxM in register TxCON to 000_B.

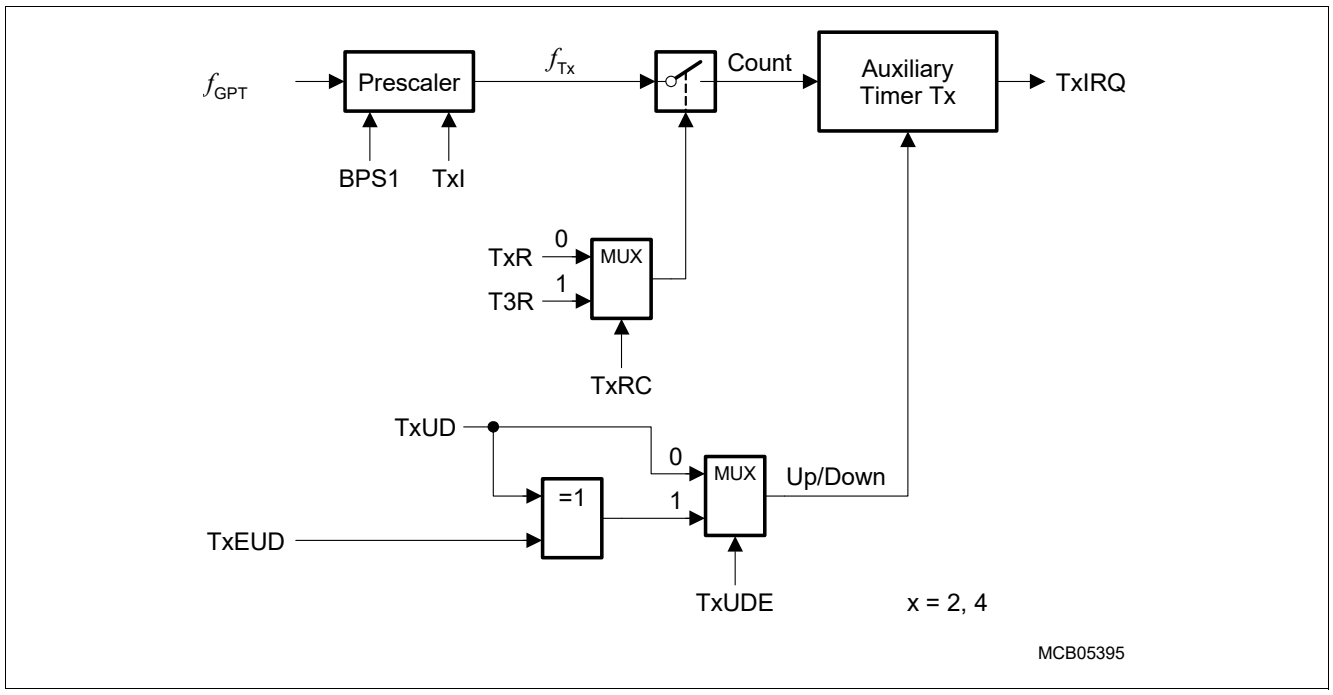


Figure 99 Block Diagram of an Auxiliary Timer in Timer Mode

Timers T2 and T4 in Gated Timer Mode

Gated Timer Mode for an auxiliary timer Tx is selected by setting bitfield TxM in register TxCON to 010_B or 011_B. Bit TxM.0 (TxCON.3) selects the active level of the gate input.

Note: A transition of the gate signal at line TxIN does not cause an interrupt request.

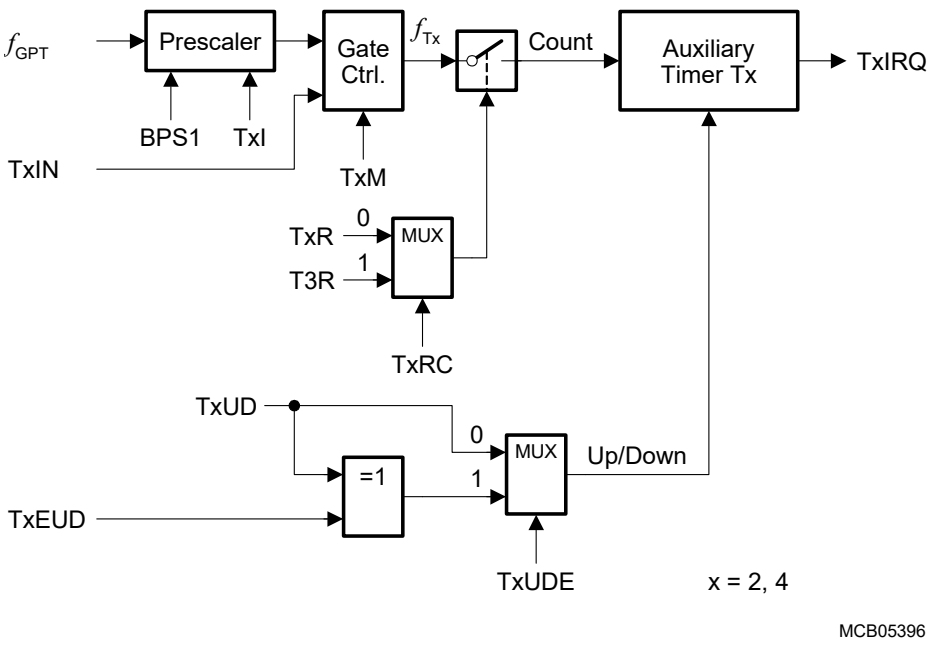


Figure 100 Block Diagram of an Auxiliary Timer in Gated Timer Mode

Note: There is no output toggle latch for T2 and T4.
Start/stop of an auxiliary timer can be controlled locally or remotely.

Timers T2 and T4 in Counter Mode

Counter Mode for an auxiliary timer Tx is selected by setting bitfield TxM in register TxCON to 001_B. In Counter Mode, an auxiliary timer can be clocked either by a transition at its external input line TxIN, or by a transition of timer T3's toggle latch T3OTL. The event causing an increment or decrement of a timer can be a positive, a negative, or both a positive and a negative transition at either the respective input pin or at the toggle latch. Bitfield TxI in control register TxCON selects the triggering transition (see **Table 269**).

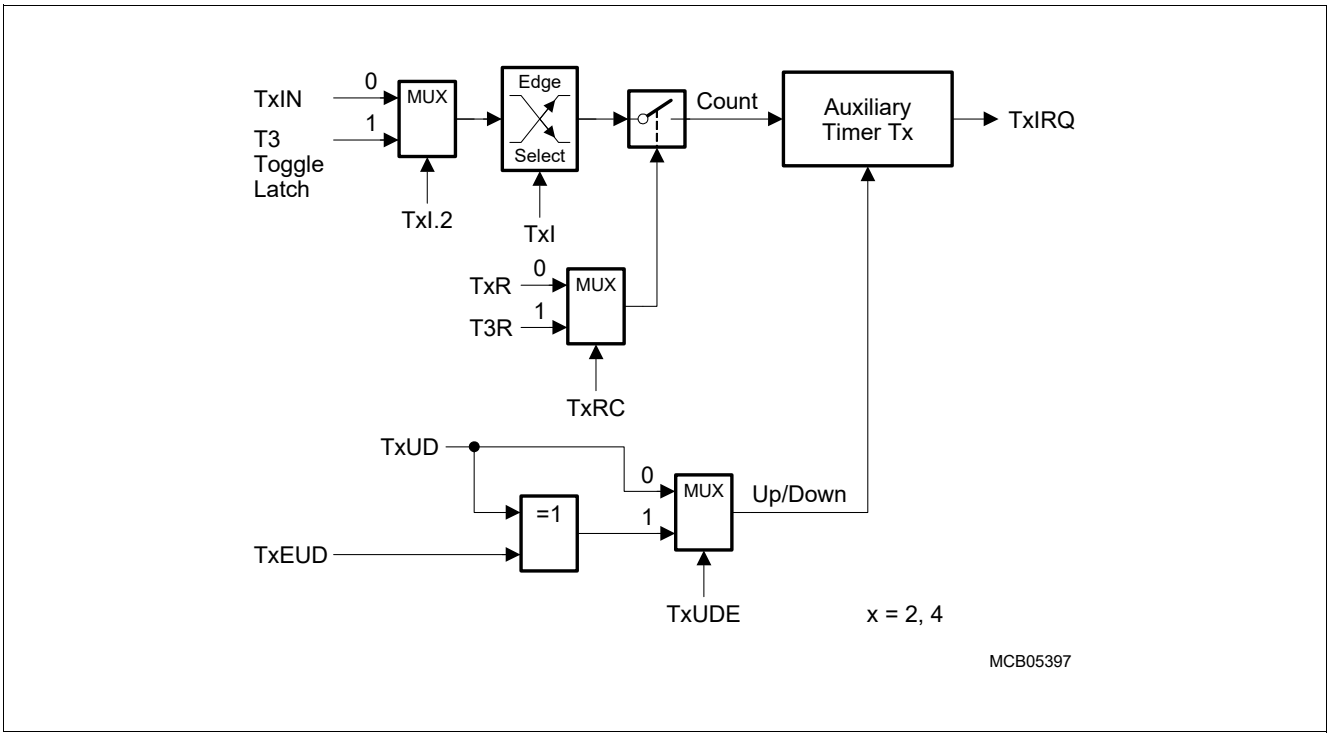


Figure 101 Block Diagram of an Auxiliary Timer in Counter Mode

Note: Only state transitions of T3OTL which are caused by the overflows/underflows of T3 will trigger the counter function of T2/T4. Modifications of T3OTL via software will NOT trigger the counter function of T2/T4.

For counter operation, pin TxIN must be configured as input. The maximum input frequency allowed in Counter Mode depends on the selected prescaler value. To ensure that a transition of the count input signal applied to TxIN is recognized correctly, its level must be held high or low for a minimum number of module clock cycles before it changes. This information can be found in **Section 16.3.5**.

Timer Concatenation

Using the toggle bit T3OTL as a clock source for an auxiliary timer in Counter Mode concatenates the core timer T3 with the respective auxiliary timer. This concatenation forms either a 32-bit or a 33-bit timer/counter, depending on which transition of T3OTL is selected to clock the auxiliary timer.

- **32-bit Timer/Counter:** If both a positive and a negative transition of T3OTL are used to clock the auxiliary timer, this timer is clocked on every overflow/underflow of the core timer T3. Thus, the two timers form a 32-bit timer.
- **33-bit Timer/Counter:** If either a positive or a negative transition of T3OTL is selected to clock the auxiliary timer, this timer is clocked on every second overflow/underflow of the core timer T3. This configuration forms a 33-bit timer (16-bit core timer + T3OTL + 16-bit auxiliary timer).
  As long as bit T3OTL is not modified by software, it represents the state of the internal toggle latch, and can be regarded as part of the 33-bit timer.

The count directions of the two concatenated timers are not required to be the same. This offers a wide variety of different configurations.

T3, which represents the low-order part of the concatenated timer, can operate in Timer Mode, Gated Timer Mode or Counter Mode in this case.

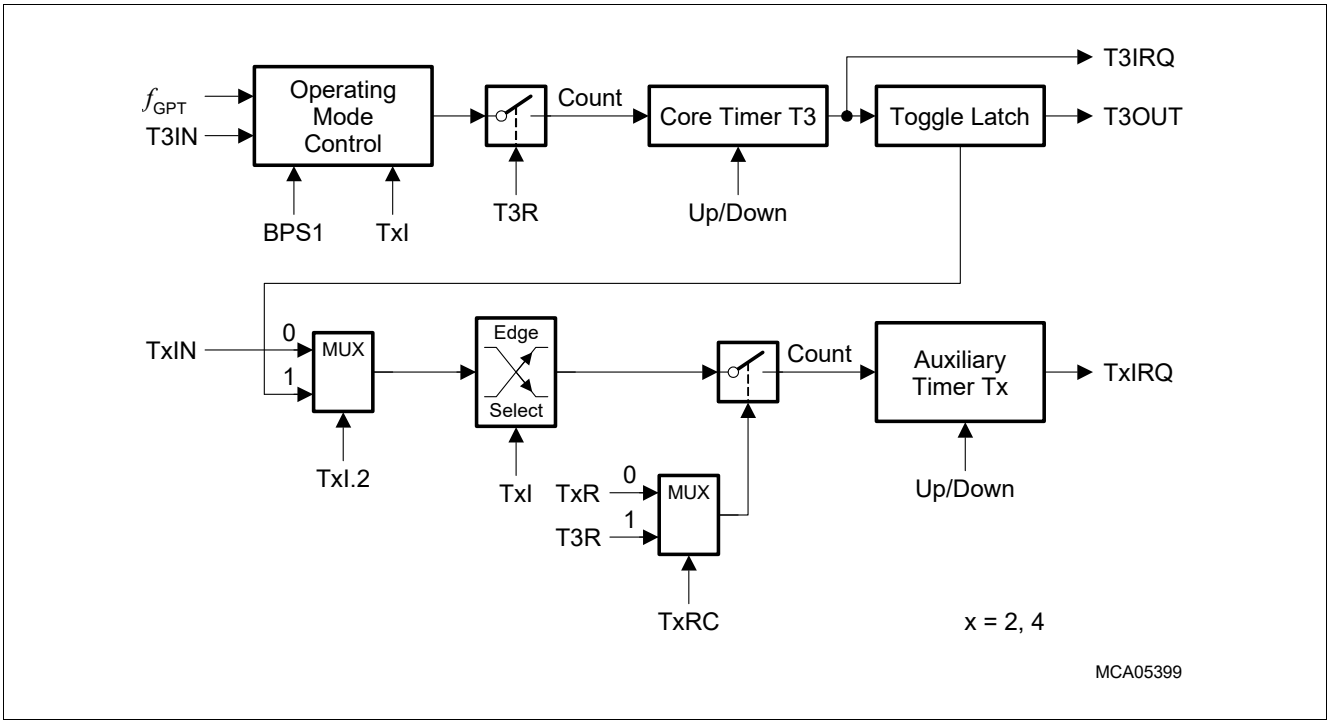


Figure 102 Concatenation of Core Timer T3 and an Auxiliary Timer

For measuring longer time periods, the core timer T3 may be concatenated with an auxiliary timer (T2/T4). The core timer contains the low part, and the auxiliary timer contains the high part of the extended timer value.

Timers T2 and T4 in Capture Mode

Capture mode for an auxiliary timer Tx is selected by setting bitfield TxM in the respective register TxCON to 101_B. In capture mode, the contents of the core timer T3 are latched into an auxiliary timer register in response to a signal transition at the respective auxiliary timer's external input pin TxIN. The capture trigger signal can be a positive, a negative, or both a positive and a negative transition.

The two least significant bits of bitfield TxI select the active transition (see **Table 269**). Bit 2 of TxI is irrelevant for capture mode and must be cleared (TxI.2 = 0).

Note: When programmed for capture mode, the respective auxiliary timer (T2 or T4) stops independently of its run flag T2R or T4R.

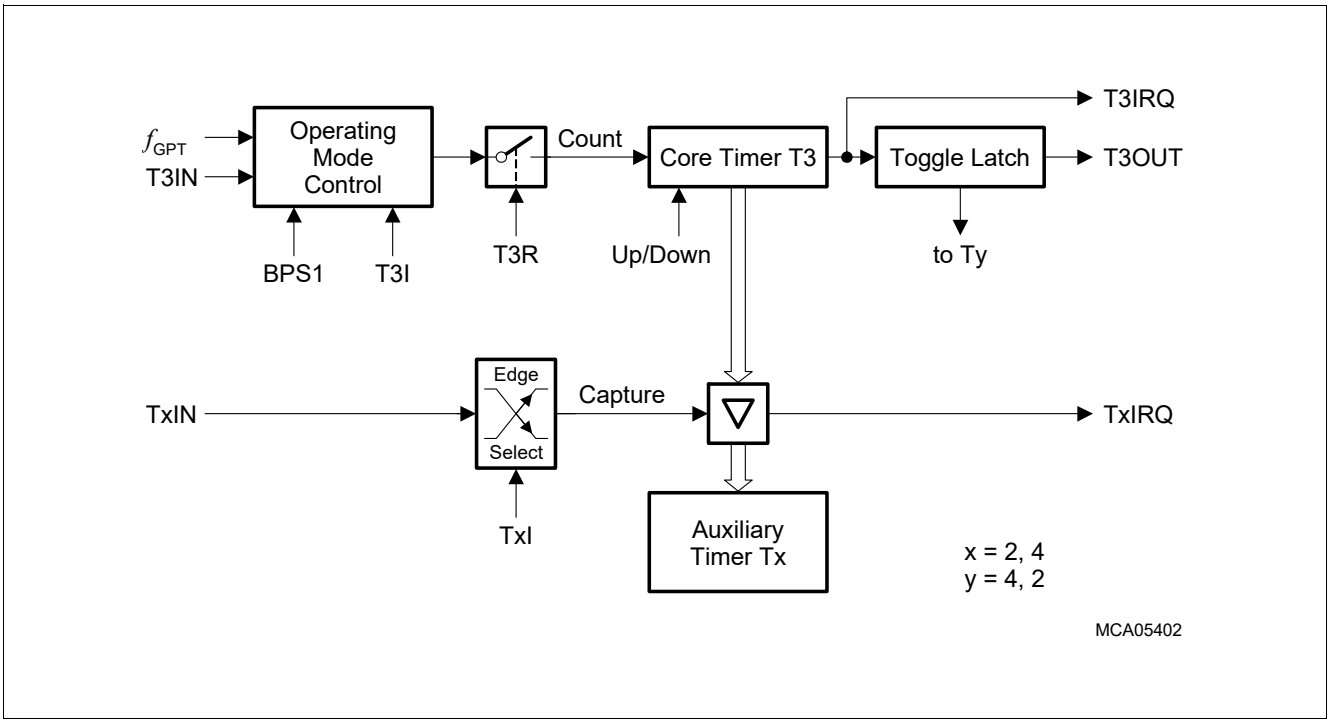


Figure 103 GPT1 Auxiliary Timer in Capture Mode

Upon a trigger (selected transition) at the corresponding input pin TxIN the contents of the core timer are loaded into the auxiliary timer register and the associated interrupt request flag TxIR will be set.

For capture mode operation, the respective timer input pin TxIN must be configured as input. To ensure that a transition of the capture input signal applied to TxIN is recognized correctly, its level must be held high or low for a minimum number of module clock cycles, detailed in **Section 16.3.5**.

Timers T2 and T4 in Incremental Interface Mode

Incremental interface mode for an auxiliary timer Tx is selected by setting bitfield TxM in the respective register TxCON to 110_B or 111_B. In Incremental Interface Mode, the two inputs associated with an auxiliary timer Tx (TxIN, TxEUD) are used to interface to an incremental encoder. Tx is clocked by each transition on one or both of the external input pins to provide 2-fold or 4-fold resolution of the encoder input.

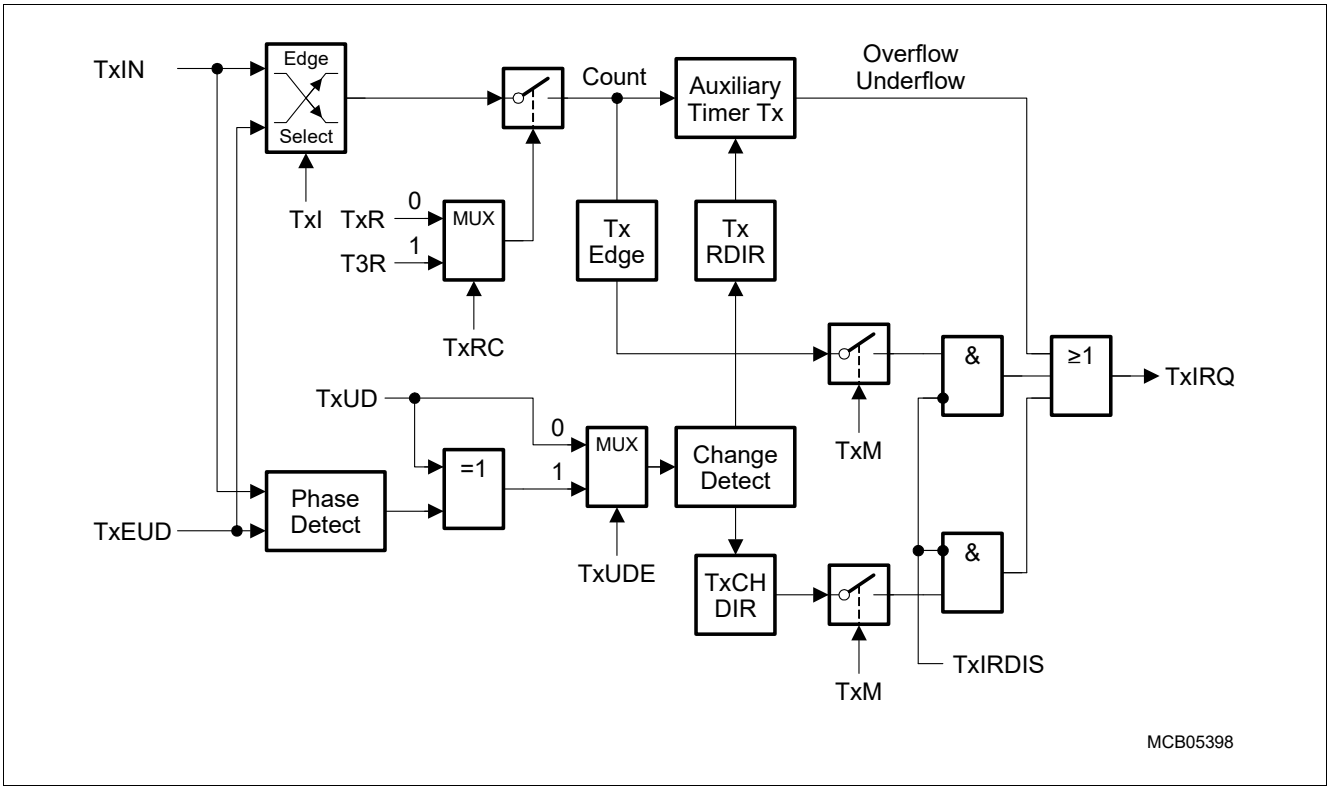


Figure 104 Block Diagram of an Auxiliary Timer in Incremental Interface Mode

The operation of the auxiliary timers T2 and T4 in Incremental Interface Mode and the interrupt generation are the same as described for the core timer T3. The descriptions, figures and tables apply accordingly.

Note: Timers T2 and T4 operating in Incremental Interface Mode automatically provide information on the sensor's current position. For dynamic information (speed, acceleration, deceleration) see ***"Combined Capture Modes" on Page 518****).*

Timers T2 and T4 in Reload Mode

Reload Mode for an auxiliary timer Tx is selected by setting bitfield TxM in the respective register TxCON to 100_B. In reload mode, the core timer T3 is reloaded with the contents of an auxiliary timer register, triggered by one of two different signals. The trigger signal is selected the same way as the clock source for Counter Mode (see **Table 269**), i.e. a transition of the auxiliary timer's input TxIN or the toggle latch T3OTL may trigger the reload.

Note: *When programmed for reload mode, the respective auxiliary timer (T2 or T4) stops independently of its run flag T2R or T4R.*
The timer input pin TxIN must be configured as input if it shall trigger a reload operation.

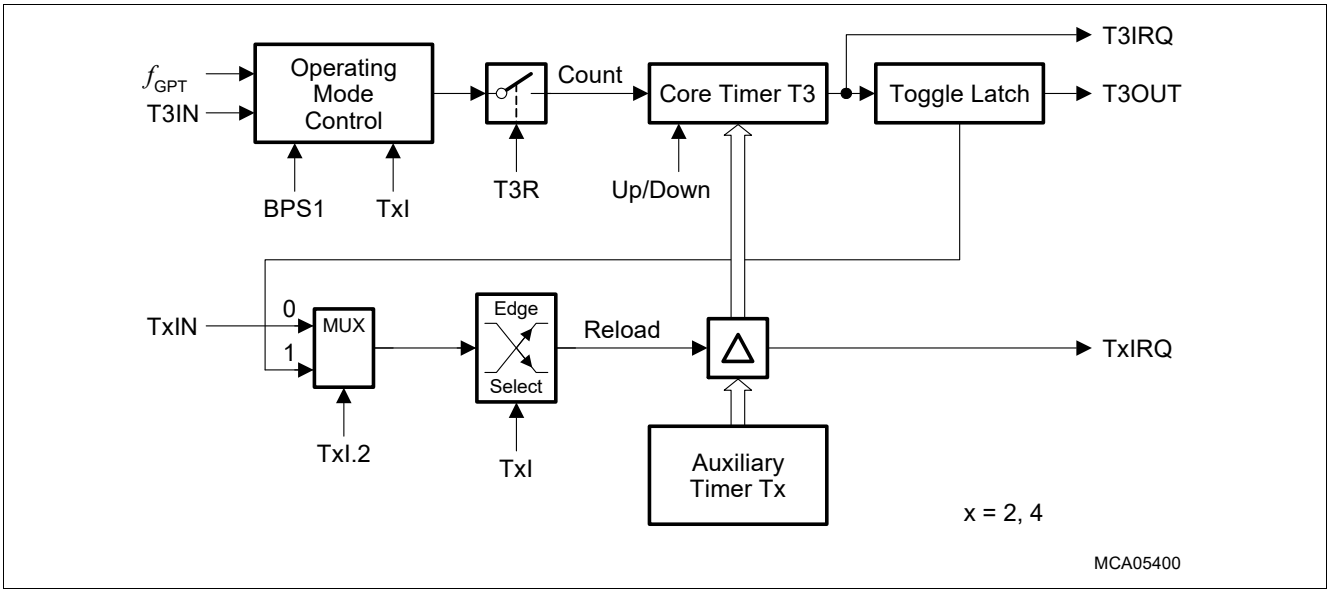


Figure 105 GPT1 Auxiliary Timer in Reload Mode

Upon a trigger signal, T3 is loaded with the contents of the respective timer register (T2 or T4) and the respective interrupt request flag (T2IR or T4IR) is set.

Note: *When a T3OTL transition is selected for the trigger signal, the interrupt request flag T3IR will also be set upon a trigger, indicating T3's overflow or underflow. Modifications of T3OTL via software will NOT trigger the counter function of T2/T4.*

To ensure that a transition of the reload input signal applied to TxIN is recognized correctly, its level must be held high or low for a minimum number of module clock cycles, detailed in **Section 16.3.5**.

The reload mode triggered by the T3 toggle latch can be used in a number of different configurations. The following functions can be performed, depending on the selected active transition:

- If both a positive and a negative transition of T3OTL are selected to trigger a reload, the core timer will be reloaded with the contents of the auxiliary timer each time it overflows or underflows. This is the standard reload mode (reload on overflow/underflow).
- If either a positive or a negative transition of T3OTL is selected to trigger a reload, the core timer will be reloaded with the contents of the auxiliary timer on every second overflow or underflow.
- Using this "single-transition" mode for both auxiliary timers allows to perform very flexible Pulse Width Modulation (PWM). One of the auxiliary timers is programmed to reload the core timer on a positive transition of T3OTL, the other is programmed for a reload on a negative transition of T3OTL. With this combination the core timer is alternately reloaded from the two auxiliary timers.

Figure 106 shows an example for the generation of a PWM signal using the “single-transition” reload mechanism. T2 defines the high time of the PWM signal (reloaded on positive transitions) and T4 defines the low time of the PWM signal (reloaded on negative transitions). The PWM signal can be output on pin T3OUT if T3OE = 1. With this method, the high and low time of the PWM signal can be varied in a wide range.

Note: The output toggle latch T3OTL is accessible via software and may be changed, if required, to modify the PWM signal.
However, this will NOT trigger the reloading of T3.

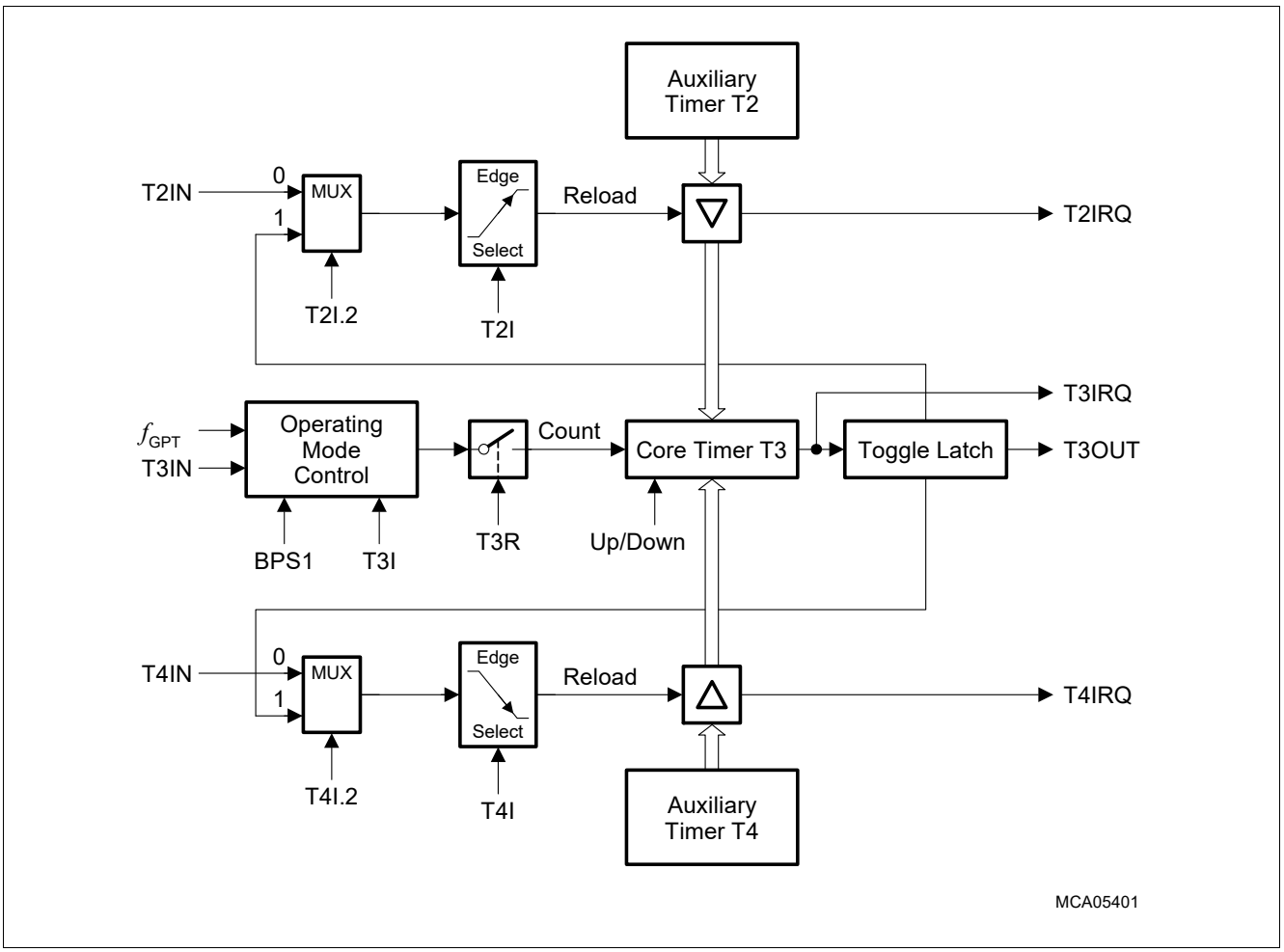


Figure 106 GPT1 Timer Reload Configuration for PWM Generation

Note: Although possible, selecting the same reload trigger event for both auxiliary timers should be avoided. In such a case, both reload registers would try to load the core timer at the same time. If this combination is selected, T2 is disregarded and the contents of T4 is reloaded.

16.3.5 GPT1 Clock Signal Control

All actions within the timer block GPT1 are triggered by transitions of its basic clock. This basic clock is derived from the system clock by a basic block prescaler, controlled by bitfield BPS1 in register T3CON (see **Figure 88**). The count clock can be generated in two different ways:

- **Internal count clock**, derived from GPT1’s basic clock via a programmable prescaler, is used for (gated) Timer Mode.
- **External count clock**, derived from the timer’s input pin(s), is used for Counter Mode.

For both ways, the basic clock determines the maximum count frequency and the timer's resolution:

Table 255 Basic Clock Selection for Block GPT1

Block Prescaler[1]	BPS1 = 01_B	BPS1 = 00_B[2]	BPS1 = 11_B	BPS1 = 10_B
Prescaling Factor for GPT1: F(BPS1)	F(BPS1) = 4	F(BPS1) = 8	F(BPS1) = 16	F(BPS1) = 32
Maximum External Count Frequency	$f_{GPT}/8$	$f_{GPT}/16$	$f_{GPT}/32$	$f_{GPT}/64$
Input Signal Stable Time	$4 \times t_{GPT}$	$8 \times t_{GPT}$	$16 \times t_{GPT}$	$32 \times t_{GPT}$

1) Please note the non-linear encoding of bitfield BPS1.

2) Default after reset.

Note: When initializing the GPT1 block, and the block prescaler BPS1 in register T3CON needs to be set to a value different from its reset value (00_B), it must be initialized first before any mode involving external trigger signals is configured. These modes include counter, incremental interface, capture, and reload mode. Otherwise, unintended count/capture/reload events may occur.
In this case (e.g. when changing BPS1 during operation of the GPT1 block), disable related interrupts before modification of BPS1, and afterwards clear the corresponding service request flags and re-initialize those registers (T2, T3, T4) that might be affected by a count/capture/reload event.

Internal Count Clock Generation

In Timer Mode and Gated Timer Mode, the count clock for each GPT1 timer is derived from the GPT1 basic clock by a programmable prescaler, controlled by bitfield TxI in the respective timer's control register TxCON.

The count frequency f_{Tx} for a timer Tx and its resolution r_{Tx} are scaled linearly with lower clock frequencies, as can be seen from the following formula:

$$f_{Tx} = \frac{f_{GPT}}{F(BPS1) \bullet 2^{<TxI>}} \qquad r_{Tx}[\mu s] = \frac{F(BPS1) \bullet 2^{<TxI>}}{f_{GPT}[MHz]} \tag{16.1}$$

The effective count frequency depends on the common module clock prescaler factor F(BPS1) as well as on the individual input prescaler factor $2^{<TxI>}$. **Table 267** summarizes the resulting overall divider factors for a GPT1 timer that result from these cascaded prescalers.

Table 256 lists GPT1 timer's parameters (such as count frequency, resolution, and period) resulting from the selected overall prescaler factor and the module clock f_{GPT}. Note that some numbers may be rounded.

Table 256 GPT1 Timer Parameters

Module Clock f_{GPT} = 10 MHz			Overall Prescaler Factor	Module Clock f_{GPT} = 40 MHz		
Frequency	**Resolution**	**Period**		**Frequency**	**Resolution**	**Period**
2.5 MHz	400 ns	26.21 ms	4	10.0 MHz	100 ns	6.55 ms
1.25 MHz	800 ns	52.43 ms	8	5.0 MHz	200 ns	13.11 ms
625.0 kHz	1.6 µs	104.9 ms	16	2.5 MHz	400 ns	26.21 ms
312.5 kHz	3.2 µs	209.7 ms	32	1.25 MHz	800 ns	52.43 ms
156.25 kHz	6.4 µs	419.4 ms	64	625.0 kHz	1.6 µs	104.9 ms
78.125 kHz	12.8 µs	838.9 ms	128	312.5 kHz	3.2 µs	209.7 ms
39.06 kHz	25.6 µs	1.678 s	256	156.25 kHz	6.4 µs	419.4 ms

Table 256 GPT1 Timer Parameters (cont'd)

Module Clock f_{GPT} = 10 MHz			Overall Prescaler Factor	Module Clock f_{GPT} = 40 MHz		
Frequency	Resolution	Period		Frequency	Resolution	Period
19.53 kHz	51.2 µs	3.355 s	512	78.125 kHz	12.8 µs	838.9 ms
9.77 kHz	102.4 µs	6.711 s	1024	39.06 kHz	25.6 µs	1.678 s
4.88 kHz	204.8 µs	13.42 s	2048	19.53 kHz	51.2 µs	3.355 s
2.44 kHz	409.6 µs	26.84 s	4096	9.77 kHz	102.4 µs	6.711 s

External Count Clock Input

The external input signals of the GPT1 block are sampled with the GPT1 basic clock (see **Figure 88**). To ensure that a signal is recognized correctly, its current level (high or low) must be held active for at least one complete sampling period, before changing. A signal transition is recognized if two subsequent samples of the input signal represent different levels. Therefore, a minimum of two basic clock periods are required for the sampling of an external input signal. Thus, the maximum frequency of an input signal must not be higher than half the basic clock.

Table 257 summarizes the resulting requirements for external GPT1 input signals.

Table 257 GPT1 External Input Signal Limits

GPT1 Basic Clock = 10 MHz		Input Frequ. Factor	GPT1 Divider BPS1	Input Phase Duration	GPT1 Basic Clock = 40 MHz	
Max. Input Frequency	Min. Level Hold Time				Max. Input Frequency	Min. Level Hold Time
1.25 MHz	400 ns	$f_{GPT}/8$	01_B	$4 \times t_{GPT}$	5.0 MHz	100 ns
625.0 kHz	800 ns	$f_{GPT}/16$	00_B	$8 \times t_{GPT}$	2.5 MHz	200 ns
312.5 kHz	1.6 µs	$f_{GPT}/32$	11_B	$16 \times t_{GPT}$	1.25 MHz	400 ns
156.25 kHz	3.2 µs	$f_{GPT}/64$	10_B	$32 \times t_{GPT}$	625.0 kHz	800 ns

These limitations are valid for all external input signals to GPT1, including the external count signals in Counter Mode and Incremental Interface Mode, the gate input signals in Gated Timer Mode, and the external direction signals.

16.3.6 Interrupt Control for GPT1 Timers

When a timer overflows from $FFFF_H$ to 0000_H (when counting up), or when it underflows from 0000_H to $FFFF_H$ (when counting down), its interrupt request flag in register GPT12E_T2, GPT12E_T3, or GPT12E_T4 will be set. This will cause an interrupt to the respective timer interrupt vector, if the respective interrupt enable bit is set.

In **Reload Mode**, upon a trigger signal, T3 is loaded with the contents of the respective timer (T2 or T4) and the respective interrupt request flag in register GPT12E_T2 or GPT12E_T4 is set.

In **Incremental Interface Mode**, the interrupt request generation can be selected as follows:

- In Rotation Detection Mode (T3M = 110_B), an interrupt request is generated each time the count direction of T3 changes.
- In Edge Detection Mode (T3M = 111_B), an interrupt request is generated each time a count edge for T3 is detected.

In **Capture Mode**, upon a trigger (selected transition) at the corresponding input pin the content of the core timer T3 are loaded into the auxiliary timer register Tx and the associated interrupt request flag in register GPTE12_T2 or GPT12E_T4 will be set.

16.3.7 GPT12 Registers

Table 258 Register Address Space

Module	Base Address	End Address	Note
GPT12E	40010000_H	$40013FFF_H$	

Table 259 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
GPT1 Registers, GPT1 Timer Registers			
GPT12E_T2	Timer T2 Count Register	20_H	see **Table 260**
GPT12E_T3	Timer T3 Count Register	24_H	see **Table 261**
GPT12E_T4	Timer T4 Count Register	28_H	see **Table 262**
GPT1 Registers, GPT1 Core Timer T3 Control Register			
GPT12E_T3CON	Timer T3 Control Register	$0C_H$	see **Table 263**
GPT1 Registers, GPT1 Auxiliary Timers T2/T4 Control Registers			
GPT12E_T2CON	Timer T2 Control Register	08_H	see **Table 264**
GPT12E_T4CON	Timer T4 Control Register	10_H	see **Table 265**
GPT2 Registers, GPT2 Timer Registers			
GPT12E_CAPREL	Capture/Reload Register	$1C_H$	see **Table 277**
GPT12E_T5	Timer 5 Count Register	$2C_H$	see **Table 275**
GPT12E_T6	Timer 6 Count Register	30_H	see **Table 276**
GPT2 Registers, GPT2 Timer Control Registers			
GPT12E_T5CON	Timer T5 Control Register	14_H	see **Table 279**
GPT12E_T6CON	Timer T6 Control Register	18_H	see **Table 278**
Miscellaneous GPT12 Registers,			
GPT12E_ID	Module Identification Register	00_H	see **Table 285**
GPT12E_PISEL	Port Input Select Register	04_H	see **Table 284**

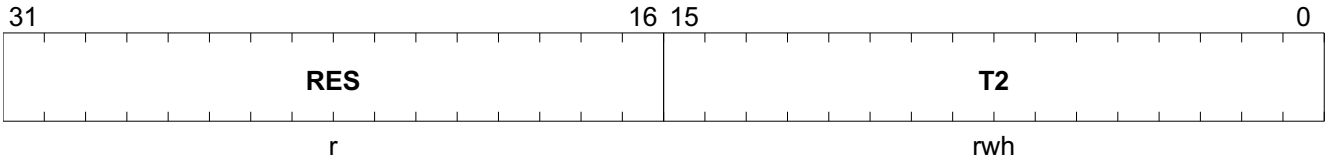
The registers are addressed wordwise.

16.3.8 GPT1 Registers

16.3.8.1 GPT1 Timer Registers

Timer T2 Count Register

GPT12E_T2	Offset	Reset Value
Timer T2 Count Register	20_H	see Table 260



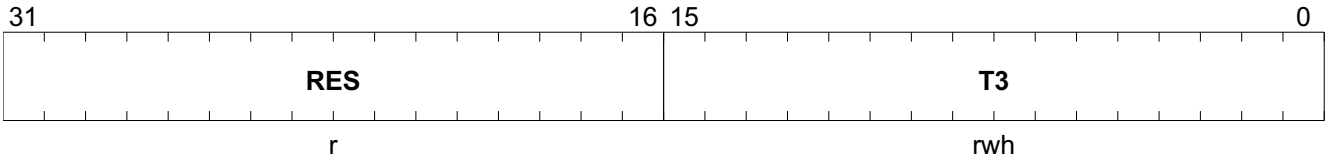
Field	Bits	Type	Description
RES	31:16	r	**Reserved**
T2	15:0	rwh	**Timer T2 Current Value** Contains the current value of the timer T2

Table 260 RESET of GPT12E_T2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Timer T3 Count Register

GPT12E_T3	Offset	Reset Value
Timer T3 Count Register	24_H	see Table 261



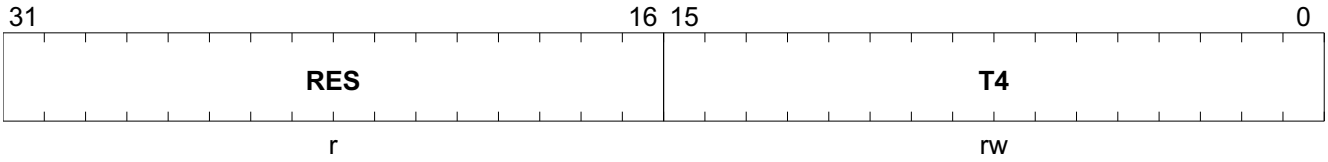
Field	Bits	Type	Description
RES	31:16	r	**Reserved**
T3	15:0	rwh	**Timer T3 Current Value** Contains the current value of the timer T3

Table 261 RESET of GPT12E_T3

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Timer T4 Count Register

GPT12E_T4	Offset	Reset Value
Timer T4 Count Register	28_H	see Table 262



Field	Bits	Type	Description
RES	31:16	r	**Reserved**
T4	15:0	rw	**Timer T4 Current Value** Contains the current value of the timer T4

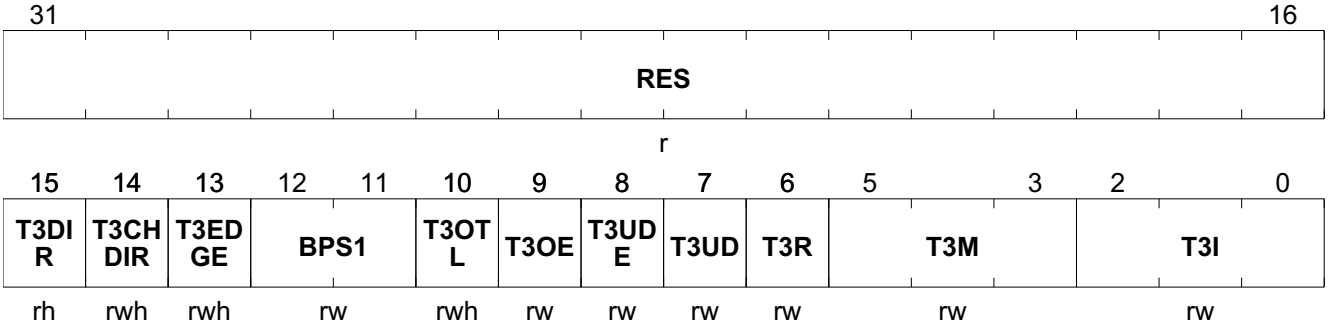
Table 262 RESET of GPT12E_T4

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

16.3.8.2 GPT1 Core Timer T3 Control Register

Timer T3 Control Register

GPT12E_T3CON	Offset	Reset Value
Timer T3 Control Register	$0C_H$	see Table 263



Field	Bits	Type	Description
RES	31:16	r	**Reserved** Read as 0; should be written with 0.
T3DIR	15	rh	**Timer T3 Rotation Direction Flag** 0_B **Up**, Timer T3 counts up 1_B **Down**, Timer T3 counts down
T3CHDIR	14	rwh	**Timer T3 Count Direction Change Flag** This bit is set each time the count direction of timer T3 changes. T3CHDIR must be cleared by software. 0_B **No change**, No change of count direction was detected 1_B **Change**, A change of count direction was detected

Field	Bits	Type	Description
T3EDGE	13	rwh	**Timer T3 Edge Detection Flag** The bit is set each time a count edge is detected. T3EDGE must be cleared by software. 0_B **No count**, No count edge was detected 1_B **Count**, A count edge was detected
BPS1	12:11	rw	**GPT1 Block Prescaler Control** Select basic clock for block GPT1 (see also **Section 16.3.5**) 00_B **8**, $f_{GPT}/8$ 01_B **4**, $f_{GPT}/4$ 10_B **32**, $f_{GPT}/32$ 11_B **16**, $f_{GPT}/16$
T3OTL	10	rwh	**Timer T3 Overflow Toggle Latch** Toggles on each overflow/underflow of T3. Can be set or cleared by software (see separate description)
T3OE	9	rw	**Overflow/Underflow Output Enable** 0_B **Disabled**, Alternate Output Function Disabled 1_B **T3OUT**, State of T3 toggle latch is output on pin T3OUT
T3UDE	8	rw	**Timer T3 External Up/Down Enable**[1)] 0_B **T3UD**, Count direction is controlled by bit T3UD; input T3EUD is disconnected 1_B **T3EUD**, Count direction is controlled by input T3EUD
T3UD	7	rw	**Timer T3 Up/Down Control**[1)] 0_B **Up**, Timer T3 counts up 1_B **Down**, Timer T3 counts down
T3R	6	rw	**Timer T3 Input Run Bit** 0_B **Stop**, Timer T3 stops 1_B **Run**, Timer T3 runs
T3M	5:3	rw	**Timer T3 Input Mode Control** 000_B **Timer Mode**, 001_B **Counter Mode**, 010_B **Gated low**, Gated Timer Mode with gate active low 011_B **Gated high**, Gated Timer Mode with gate active high 100_B **Reserved**, Do not use this combination 101_B **Reserved**, Do not use this combination 110_B **Incremental Interface Mode**, (Rotation Detection Mode) 111_B **Incremental Interface Mode**, (Edge Detection Mode)
T3I	2:0	rw	**Timer T3 Input Parameter Selection** Depends on the operating mode, see respective sections for encoding: **Table 267** for Timer Mode and Gated Timer Mode **Table 268** for Counter Mode **Table 270** for Incremental Interface Mode

1) See **Table 280** for encoding of bits T3UD and T3UDE.

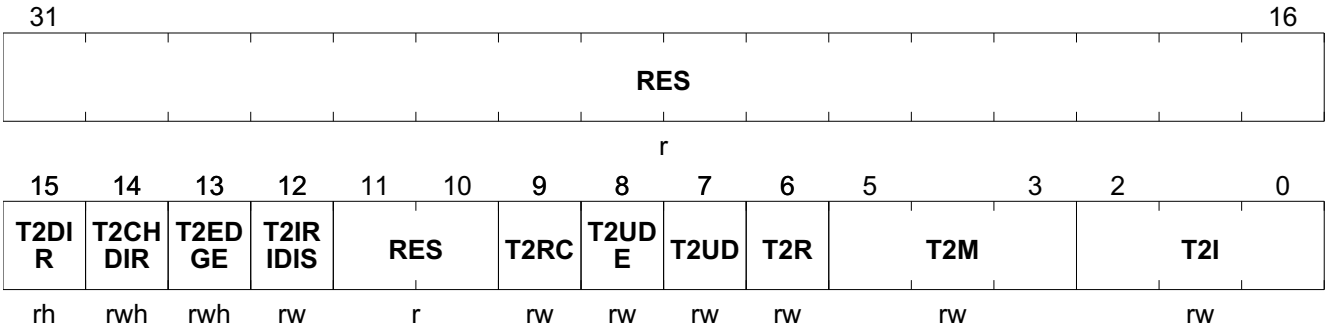
Table 263 RESET of GPT12E_T3CON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

16.3.8.3 GPT1 Auxiliary Timers T2/T4 Control Registers

Timer T2 Control Register

GPT12E_T2CON	Offset	Reset Value
Timer T2 Control Register	08_H	see Table 264



Field	Bits	Type	Description
RES	31:16	r	**Reserved** Read as 0; should be written with 0.
T2DIR	15	rh	**Timer T2 Rotation Direction** 0_B **Up**, Timer T2 counts up 1_B **Down**, Timer T2 counts down
T2CHDIR	14	rwh	**Timer T2 Count Direction Change** This bit is set each time the count direction of timer T2 changes. T2CHDIR must be cleared by software. 0_B **No change**, No change of count direction was detected 1_B **Change**, A change of count direction was detected
T2EDGE	13	rwh	**Timer T2 Edge Detection** The bit is set each time a count edge is detected. T2EDGE must be cleared by software. 0_B **No count**, No count edge was detected 1_B **Count**, A count edge was detected
T2IRIDIS	12	rw	**Timer T2 Interrupt Disable** 0_B **Enabled**, Interrupt generation for T2CHDIR and T2EDGE interrupts in Incremental Interface Mode is enabled 1_B **Disabled**, Interrupt generation for T2CHDIR and T2EDGE interrupts in Incremental Interface Mode is disabled
RES	11:10	r	**Reserved** Read as 0; should be written with 0.
T2RC	9	rw	**Timer T2 Remote Control** 0_B **T2R**, Timer T2 is controlled by its own run bit T2R 1_B **T3R**, Timer T2 is controlled by the run bit T3R of core timer T3, not by bit T2R

Field	Bits	Type	Description
T2UDE	8	rw	**Timer T2 External Up/Down Enable**[1)] 0_B **T2UD**, Count direction is controlled by bit T2UD; input T2EUD is disconnected 1_B **T2EUD**, Count direction is controlled by input T2EUD
T2UD	7	rw	**Timer T2 Up/Down Control**[1)] 0_B **Up**, Timer T2 counts up 1_B **Down**, Timer T2 counts down
T2R	6	rw	**Timer T2 Input Run Bit** 0_B **Stop**, Timer T2 stops 1_B **Run**, Timer T2 runs
T2M	5:3	rw	**Timer T2 Input Mode Control** 000_B **Timer Mode**, 001_B **Counter Mode**, 010_B **Gated low**, Gated Timer Mode with gate active low 011_B **Gated high**, Gated Timer Mode with gate active high 100_B **Reload Mode**, 101_B **Capture Mode**, 110_B **Incremental Interface Mode**, (Rotation Detection Mode) 111_B **Incremental Interface Mode**, (Edge Detection Mode)
T2I	2:0	rw	**Timer T2 Input Parameter Selection** Depends on the operating mode, see respective sections for encoding: **Table 267** for Timer Mode and Gated Timer Mode **Table 268** for Counter Mode **Table 270** for Incremental Interface Mode

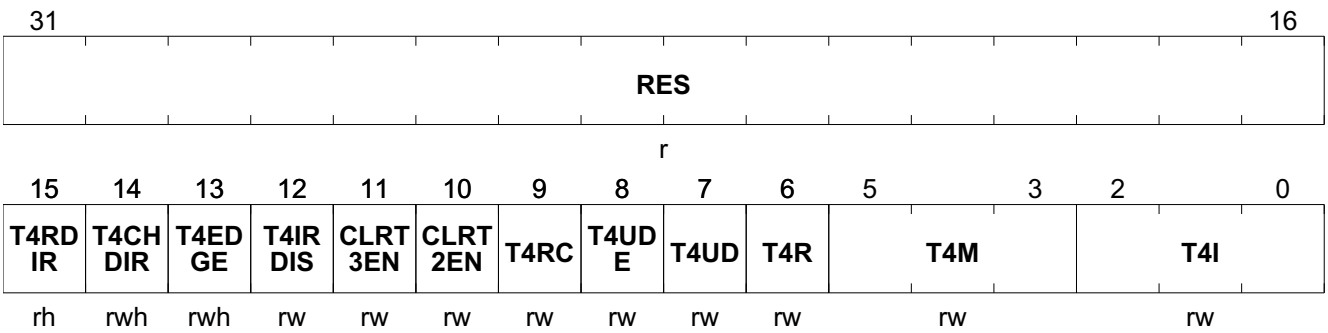
1) See **Table 280** for encoding of bits T3UD and T3UDE.

Table 264 RESET of GPT12E_T2CON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Timer T4 Control Register

GPT12E_T4CON	Offset	Reset Value
Timer T4 Control Register	10_H	see Table 265



Field	Bits	Type	Description
RES	31:16	r	**Reserved** Read as 0; should be written with 0.
T4RDIR	15	rh	**Timer T4 Rotation Direction** 0_B **Up**, Timer T4 counts up 1_B **Down**, Timer T4 counts down
T4CHDIR	14	rwh	**Timer T4 Count Direction Change** The bit is set each time a count direction of timer T4 changes. T4EDGE must be cleared by software 0_B **No change**, No change in count direction was detected 1_B **Change**, A change in count direction was detected
T4EDGE	13	rwh	**Timer T4 Edge Direction** The bit is set each time a count edge is detected. T4EDGE has to be cleared by software 0_B **No count**, No count edge was detected 1_B **Count**, A count edge was detected
T4IRDIS	12	rw	**Timer T4 Interrupt Disable** 0_B **Enabled**, Interrupt generation for T4CHDIR and T4EDGE interrupts in Incremental Interface Mode is enabled 1_B **Disabled**, Interrupt generation for T4CHDIR and T4EDGE interrupts in Incremental Interface Mode is disabled
CLRT3EN	11	rw	**Clear Timer T3 Enable** Enables the automatic clearing of timer T3 upon a falling edge of the selected T4In input. 0_B **No effect**, No effect of T4IN on Timer T3 1_B **Clear**, A falling edge on T4In clears timer T3
CLRT2EN	10	rw	**Clear Timer T2 Enable** Enables the automatic clearing of timer T2 upon a falling edge of the selected T4EUD input. 0_B **No effect**, No effect of T4EUD on timer T2 1_B **Clear**, A falling edge on T4EUD clears timer T2
T4RC	9	rw	**Timer T4 Remote Control** 0_B **T4R**, Timer T4 is controlled by its own run bit T4R 1_B **T3R**, Timer T4 is controlled by the run bit T3R of core timer T3, but not by bit T4R
T4UDE	8	rw	**Timer T4 External Up/Down Enable**[1)] 0_B **T4UD**, Count direction is controlled by bit T4UD; input T4EUD is disconnected 1_B **T4EUD**, Count direction is controlled by input T4EUD
T4UD	7	rw	**Timer T2 Up/Down Control**[1)] 0_B **Up**, Timer T2 counts up 1_B **Down**, Timer T2 counts down
T4R	6	rw	**Timer T4 Input Run Bit** 0_B **Stop**, Timer T4 stops 1_B **Run**, Timer T4 runs

Field	Bits	Type	Description
T4M	5:3	rw	**Timer T4 Mode Control (Basic Operating Mode)** 000_B **Timer Mode**, 001_B **Counter Mode**, 010_B **Gated low**, Gated Timer Mode with gate active low 011_B **Gated high**, Gated Timer Mode with gate active high 100_B **Reload Mode**, 101_B **Capture Mode**, 110_B **Incremental Interface Mode**, (Rotation Detection Mode) 111_B **Incremental Interface Mode**, (Edge Detection Mode)
T4I	2:0	rw	**Timer T4 Input Parameter Selection** Depends on the operating mode, see respective sections for encoding: **Table 267** for Timer Mode and Gated Timer Mode **Table 268** for Counter Mode **Table 270** for Incremental Interface Mode

1) See **Table 280** for encoding of bits T3UD and T3UDE.

Table 265 RESET of GPT12E_T4CON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

16.3.8.4 Encoding

Encoding of GPT1 Timer Count Direction Control

Table 266 GPT1 Timer Count Direction Control

Pin TxEUD	Bit TxUDE	Bit TxUD	Count Direction	Bit TxRDIR
X	0	0	Count Up	0
X	0	1	Count Down	1
0	1	0	Count Up	0
1	1	0	Count Down	1
0	1	1	Count Down	1
1	1	1	Count Up	0

Timer Mode and Gated Timer Mode: Encoding of GPT1 Overall Prescaler Factor

Table 267 GPT1 Overall Prescaler Factors for Internal Count Clock (Timer Mode and Gated Timer Mode)

Individual Prescaler for Tx	Common Prescaler for Module Clock[1)]			
	BPS1 = 01_B	**BPS1 = 00_B**	**BPS1 = 11_B**	**BPS1 = 10_B**
TxI = 000_B	4	8	16	32
TxI = 001_B	8	16	32	64

Table 267 GPT1 Overall Prescaler Factors for Internal Count Clock (Timer Mode and Gated Timer Mode) (cont'd)

Individual Prescaler for Tx	Common Prescaler for Module Clock[1]			
	BPS1 = 01_B	**BPS1 = 00_B**	**BPS1 = 11_B**	**BPS1 = 10_B**
TxI = 010_B	16	32	64	128
TxI = 011_B	32	64	128	256
TxI = 100_B	64	128	256	512
TxI = 101_B	128	256	512	1024
TxI = 110_B	256	512	1024	2048
TxI = 111_B	512	1024	2048	4096

1) Please note the non-linear encoding of bitfield BPS1.

Counter Mode: Encoding of GPT1 Input Edge Selection

Table 268 GPT1 Core Timer T3 Input Edge Selection (Counter Mode)

T3I	Triggering Edge for Counter Increment/Decrement
000_B	None. Counter T3 is disabled
001_B	Positive transition (rising edge) on T3IN
010_B	Negative transition (falling edge) on T3IN
011_B	Any transition (rising or falling edge) on T3IN
$1XX_B$	Reserved. Do not use this combination

Table 269 GPT1 Auxiliary Timers T2/T4 Input Edge Selection (Counter Mode)

T2I/T4I	Triggering Edge for Counter Increment/Decrement
$X00_B$	None. Counter Tx is disabled
001_B	Positive transition (rising edge) on TxIN
010_B	Negative transition (falling edge) on TxIN
011_B	Any transition (rising or falling edge) on TxIN
101_B	Positive transition (rising edge) of T3 toggle latch T3OTL
110_B	Negative transition (falling edge) of T3 toggle latch T3OTL
111_B	Any transition (rising or falling edge) of T3 toggle latch T3OTL

Incremental Interface Mode: Encoding of Input Edge Selection

Table 270 GPT1 Core Timer T3 Input Edge Selection (Incremental Interface Mode)

T3I	Triggering Edge for Counter Increment/Decrement
000_B	None. Counter T3 stops.
001_B	Any transition (rising or falling edge) on T3IN.
010_B	Any transition (rising or falling edge) on T3EUD.
011_B	Any transition (rising or falling edge) on any T3 input (T3IN or T3EUD).
$1XX_B$	Reserved. Do not use this combination.

16.3.8.5 GPT1 Timer Interrupt Control Registers

The Interrupt Control and Status register are located in the SCU.

16.4 Timer Block GPT2

From a programmer's point of view, the GPT2 block is represented by a set of SFRs as summarized below. Those portions of port and direction registers which are used for alternate functions by the GPT2 block are shaded.

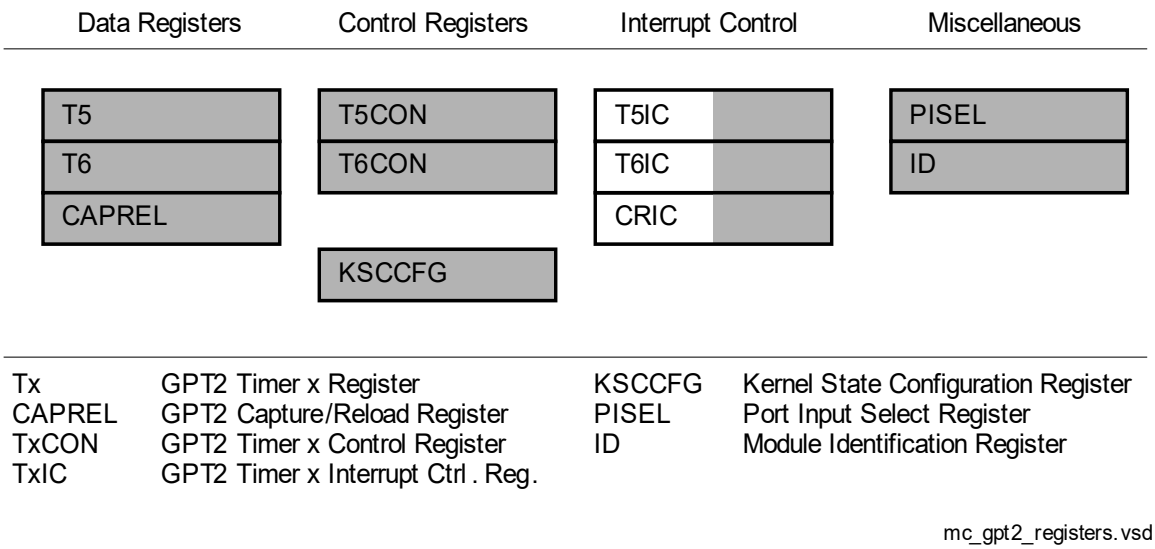


Figure 107 SFRs Associated with Timer Block GPT2

Both timers of block GPT2 (T5, T6) can run in one of 3 basic modes: Timer Mode, Gated Timer Mode, or Counter Mode. All timers can count up or down. Each timer of GPT2 is controlled by a separate control register TxCON.

Each timer has an input pin TxIN (alternate pin function) associated with it, which serves as the gate control in Gated Timer Mode, or as the count input in Counter Mode. The count direction (up/down) may be programmed via software or may be dynamically altered by a signal at the External Up/Down control input TxEUD (alternate pin function). An overflow/underflow of core timer T6 is indicated by the Output Toggle Latch T6OTL, whose state may be output on the associated pin T6OUT (alternate pin function). The auxiliary timer T5 may additionally be concatenated with core timer T6 (through T6OTL).

The Capture/Reload register CAPREL can be used to capture the contents of timer T5, or to reload timer T6. A special mode facilitates the use of register CAPREL for both functions at the same time. This mode allows frequency multiplication. The capture function is triggered by the input pin CAPIN, or by GPT1 timer's T3 input lines T3IN and T3EUD. The reload function is triggered by an overflow or underflow of timer T6. Overflows/underflows of timer T6 may also clock the timers of the CAPCOM units.

The current contents of each timer can be read or modified by the CPU by accessing the corresponding timer count registers T5 or T6, located in the SFR space (see **Section 16.4.8.1**). When any of the timer registers is written to by the CPU in the state immediately preceding a timer increment, decrement, reload, or capture operation, the CPU write operation has priority in order to guarantee correct results.

The interrupts of GPT2 are controlled through the GPTM1IEN and GPTM1IRC. These registers are not part of the GPT2 block.

The input and output lines of GPT2 are connected to pins. The control registers for the port functions are located in the respective port modules.

*Note: The timing requirements for external input signals can be found in **Section 16.4.6**, **Section 16.6.1** summarizes the module interface signals, including pins.*

16.4.1 GPT2 Core Timer T6 Control

The current contents of the core timer T6 are reflected by its count register T6. This register can also be written to by the CPU, for example, to set the initial start value.

The core timer T6 is configured and controlled via its control register T6CON.

Timer T6 Run Control

The core timer T6 can be started or stopped by software through bit T6R (timer T6 run bit). This bit is relevant in all operating modes of T6. Setting bit T6R will start the timer, clearing bit T6R stops the timer.

In Gated Timer Mode, the timer will only run if T6R = 1 and the gate is active (high or low, as programmed).

Note: When bit T5RC in timer control register T5CON is set, bit T6R will also control (start and stop) the Auxiliary Timer T5.

Count Direction Control

The count direction of the GPT2 timers (core timer and auxiliary timer) can be controlled either by software or by the external input pin TxEUD (Timer Tx External Up/Down Control Input). These options are selected by bits TxUD and TxUDE in the respective control register TxCON. When the up/down control is provided by software (bit TxUDE = 0), the count direction can be altered by setting or clearing bit TxUD. When bit TxUDE = 1, pin TxEUD is selected to be the controlling source of the count direction. However, bit TxUD can still be used to reverse the actual count direction, as shown in **Table 280**. The count direction can be changed regardless of whether or not the timer is running.

Note: When pin TxEUD is used as external count direction control input, it must be configured as input.

Timer T6 Output Toggle Latch

The overflow/underflow signal of timer T6 is connected to a block named ‘Toggle Latch’, shown in the Timer Mode diagrams. **Figure 108** illustrates the details of this block. An overflow or underflow of T6 will clock two latches: The first latch represents bit T6OTL in control register T6CON. The second latch is an internal latch toggled by T6OTL’s output. Both latch outputs are connected to the input control block of the auxiliary timer T5. The output level of the shadow latch will match the output level of T6OTL, but is delayed by one clock cycle. When the T6OTL value changes, this will result in a temporarily different output level from T6OTL and the shadow latch, which can trigger the selected count event in T5.

When software writes to T6OTL, both latches are set or cleared simultaneously. In this case, both signals to the auxiliary timers carry the same level and no edge will be detected. Bit T6OE (overflow/underflow output enable) in register T6CON enables the state of T6OTL to be monitored via an external pin T6OUT. When T6OTL is linked to an external port pin (must be configured as output), T6OUT can be used to control external HW. If T6OE = 1, pin T6OUT outputs the state of T6OTL. If T6OE = 0, pin T6OUT outputs a high level (while it selects the timer output signal).

As can be seen from **Figure 108**, when latch T6OTL is modified by software to determine the state of the output line, also the internal shadow latch is set or cleared accordingly. Therefore, no trigger condition is detected by T5 in this case.

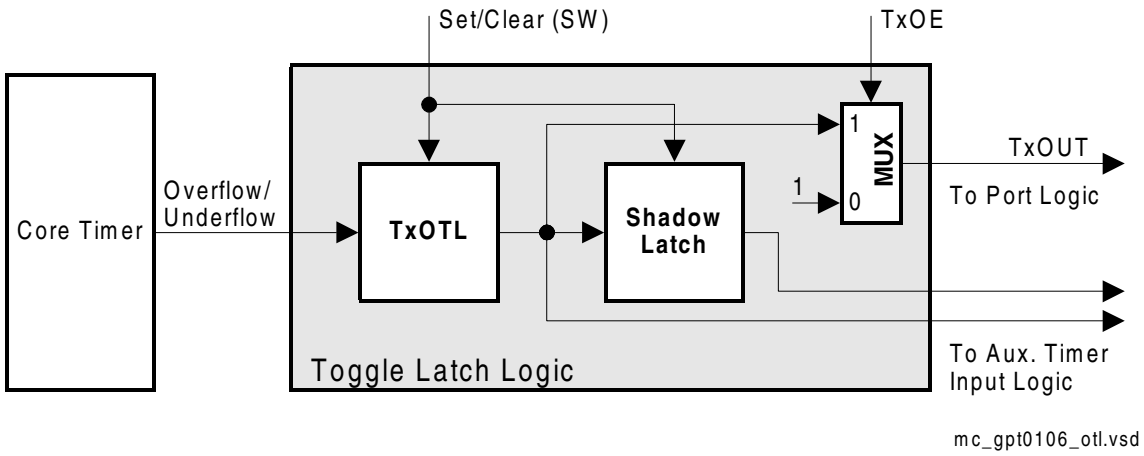


Figure 108 Block Diagram of the Toggle Latch Logic of Core Timer T6 (x = 6)

Note: T6 is also used to clock the timers in the CAPCOM units. For this purpose, there is a direct internal connection between the T6 overflow/underflow line and the CAPCOM timers (signal T6OUF).

16.4.2 GPT2 Core Timer T6 Operating Modes

Timer T6 can operate in one of several modes.

Timer T6 in Timer Mode

Timer mode for the core timer T6 is selected by setting bitfield T6M in register T6CON to 000_B. In this mode, T6 is clocked with the module's input clock f_{GPT} divided by two programmable prescalers controlled by bitfields BPS2 and T6I in register T6CON. Please see **Section 16.4.6** for details on the input clock options.

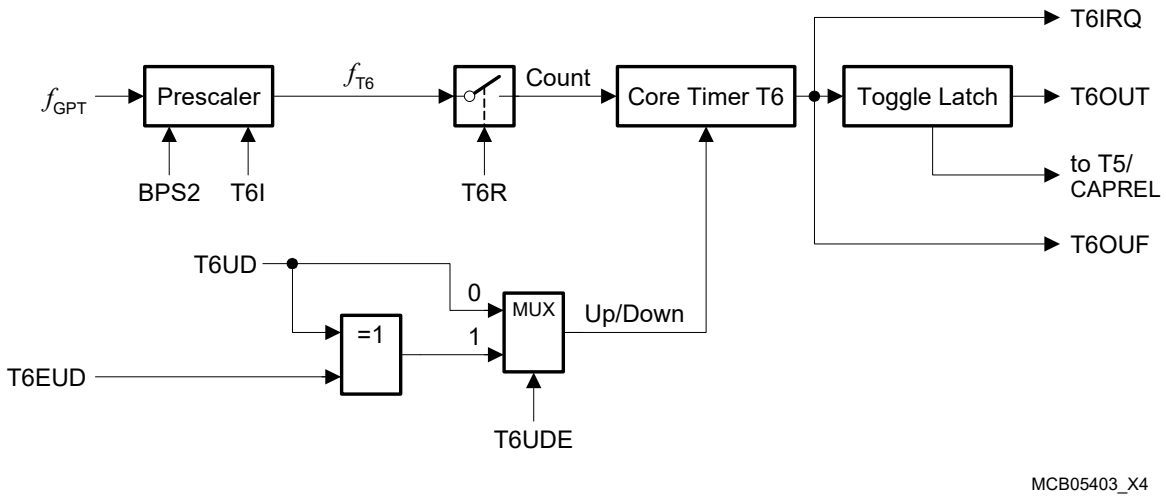


Figure 109 Block Diagram of Core Timer T6 in Timer Mode

Timer T6 in Gated Timer Mode

Gated Timer Mode for the core timer T6 is selected by setting bitfield T6M in register T6CON to 010_B or 011_B. Bit T6M.0 (T6CON.3) selects the active level of the gate input. The same options for the input frequency are available in Gated Timer Mode as in Timer Mode (see **Section 16.4.6**). However, the input clock to the timer in this mode is gated by the external input pin T6IN (Timer T6 External Input).
To enable this operation, the associated pin T6IN must be configured as input.

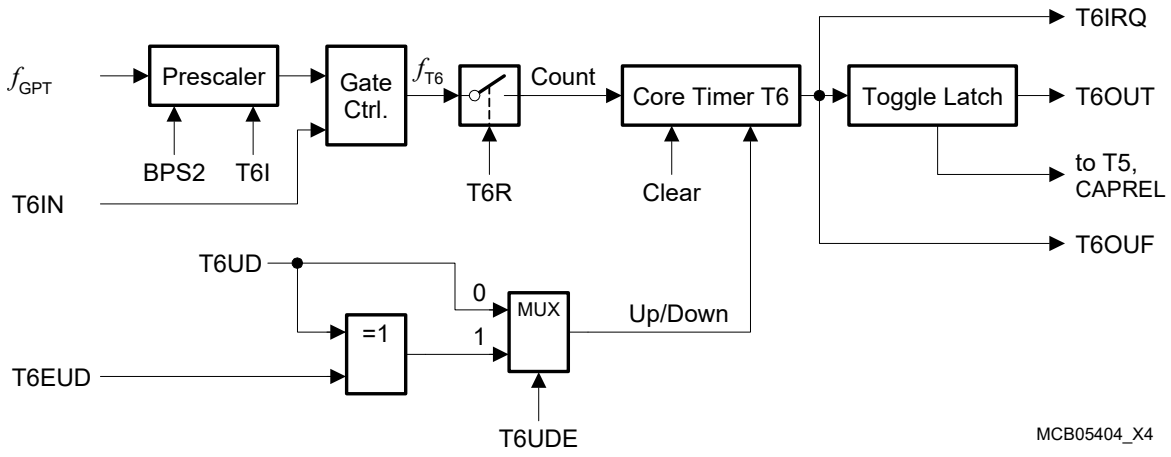


Figure 110 Block Diagram of Core Timer T6 in Gated Timer Mode

If T6M = 010_B, the timer is enabled when T6IN shows a low level. A high level at this line stops the timer. If T6M = 011_B, line T6IN must have a high level in order to enable the timer. Additionally, the timer can be turned on or off by software using bit T6R. The timer will only run if T6R is 1 and the gate is active. It will stop if either T6R is 0 or the gate is inactive.

Note: A transition of the gate signal at pin T6IN does not cause an interrupt request.

Timer T6 in Counter Mode

Counter Mode for the core timer T6 is selected by setting bitfield T6M in register T6CON to 001_B. In Counter Mode, timer T6 is clocked by a transition at the external input pin T6IN. The event causing an increment or decrement of the timer can be a positive, a negative, or both a positive and a negative transition at this line. Bitfield T6I in control register T6CON selects the triggering transition (see **Table 283**).

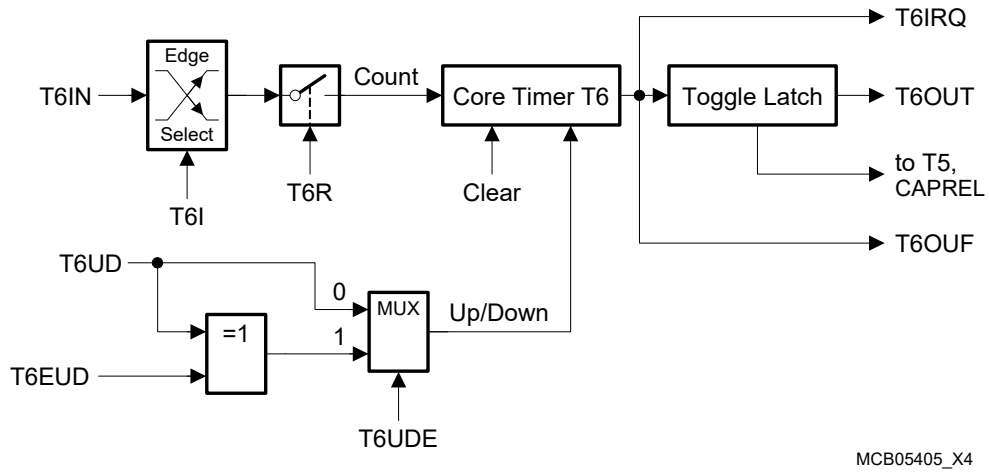


Figure 111 Block Diagram of Core Timer T6 in Counter Mode

For Counter Mode operation, pin T6IN must be configured as input. The maximum input frequency allowed in Counter Mode depends on the selected prescaler value. To ensure that a transition of the count input signal applied to T6IN is recognized correctly, its level must be held high or low for a minimum number of module clock cycles before it changes. This information can be found in **Section 16.4.6**.

16.4.3 GPT2 Auxiliary Timer T5 Control

Auxiliary timer T5 can be configured for Timer Mode, Gated Timer Mode, or Counter Mode with the same options for the timer frequencies and the count signal as the core timer T6. In addition to these 3 counting modes, the auxiliary timer can be concatenated with the core timer. The contents of T5 may be captured to register CAPREL upon an external or an internal trigger. The start/stop function of the auxiliary timers can be remotely controlled by the T6 run control bit. Several timers may thus be controlled synchronously.

The current contents of the auxiliary timer are reflected by its count register T5. This register can also be written to by the CPU, for example, to set the initial start value.

The individual configurations for timer T5 are determined by its control register T5CON. Some bits in this register also control the function of the CAPREL register. Note that functions which are present in all timers of block GPT2 are controlled in the same bit positions and in the same manner in each of the specific control registers.

Note: The auxiliary timer has no output toggle latch and no alternate output function.

Timer T5 Run Control

The auxiliary timer T5 can be started or stopped by software in two different ways:

- Through the associated timer run bit (T5R). In this case it is required that the respective control bit T5RC = 0.
- Through the core timer's run bit (T6R). In this case the respective remote control bit must be set (T5RC = 1).

The selected run bit is relevant in all operating modes of T5. Setting the bit will start the timer, clearing the bit stops the timer.

In Gated Timer Mode, the timer will only run if the selected run bit is set and the gate is active (high or low, as programmed).

Note: If remote control is selected T6R will start/stop timer T6 and the auxiliary timer T5 synchronously.

16.4.4 GPT2 Auxiliary Timer T5 Operating Modes

The operation of the auxiliary timer in the basic operating modes is almost identical with the core timer's operation, with very few exceptions. Additionally, some combined operating modes can be selected.

Timer T5 in Timer Mode

Timer Mode for the auxiliary timer T5 is selected by setting its bitfield T5M in register T5CON to 000_B.

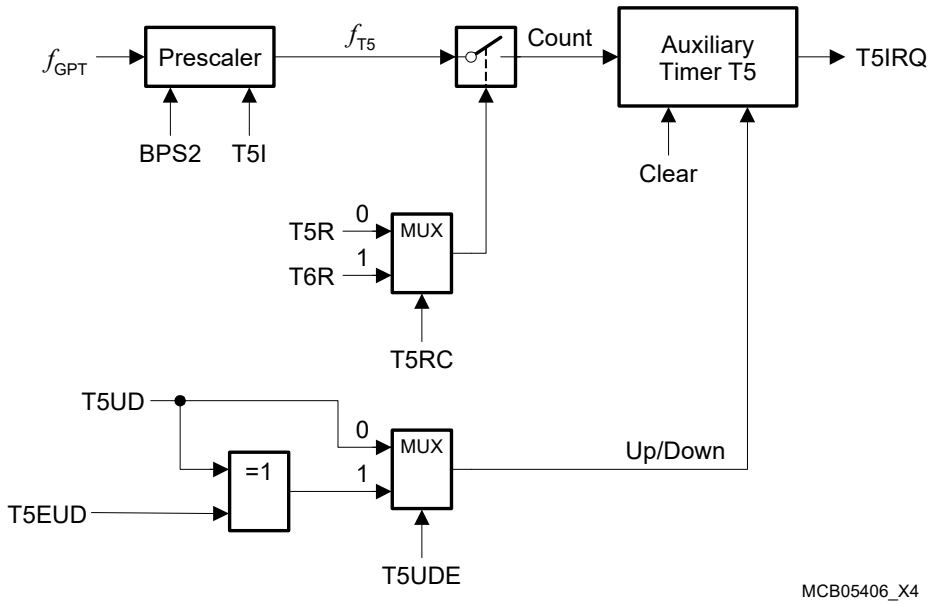


Figure 112 Block Diagram of Auxiliary Timer T5 in Timer Mode

Timer T5 in Gated Timer Mode

Gated Timer Mode for the auxiliary timer T5 is selected by setting bitfield T5M in register T5CON to 010_B or 011_B. Bit T5M.0 (T5CON.3) selects the active level of the gate input.

Note: A transition of the gate signal at line T5IN does not cause an interrupt request.

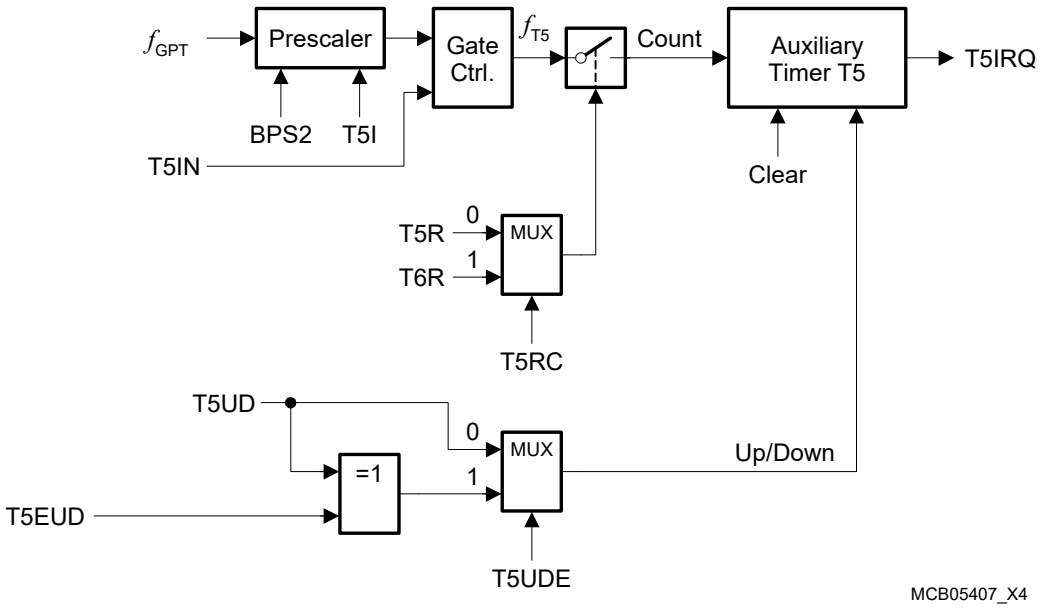


Figure 113 Block Diagram of Auxiliary Timer T5 in Gated Timer Mode

Note: There is no output toggle latch for T5.
Start/stop of the auxiliary timer can be controlled locally or remotely.

Timer T5 in Counter Mode

Counter Mode for auxiliary timer T5 is selected by setting bitfield T5M in register T5CON to 001_B. In Counter Mode, the auxiliary timer can be clocked either by a transition at its external input line T5IN, or by a transition of timer T6's toggle latch T6OTL. The event causing an increment or decrement of a timer can be a positive, a negative, or both a positive and a negative transition at either the respective input pin or at the toggle latch. Bitfield T5I in control register T5CON selects the triggering transition (see **Table 282**).

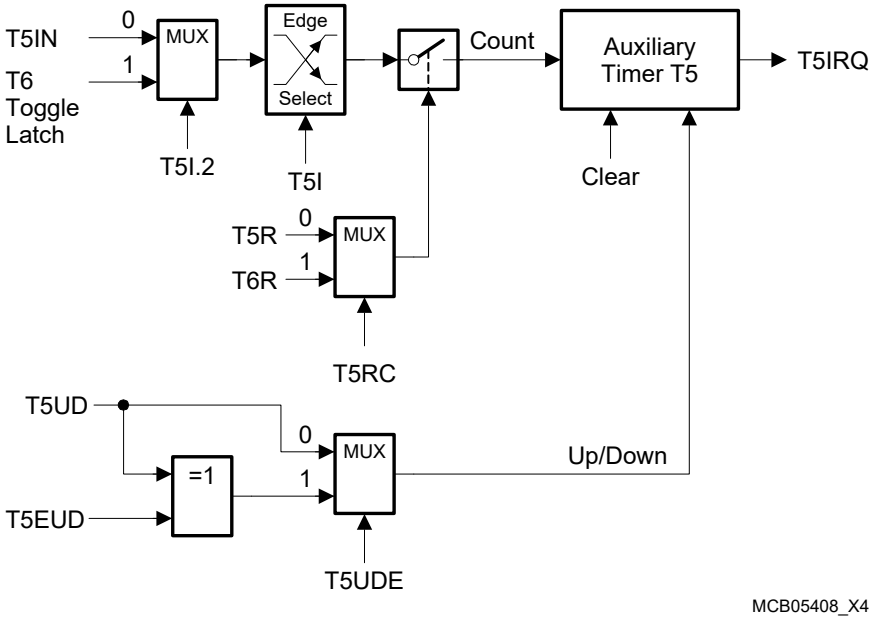


Figure 114 Block Diagram of Auxiliary Timer T5 in Counter Mode

Note: Only state transitions of T6OTL which are caused by the overflows/underflows of T6 will trigger the counter function of T5. Modifications of T6OTL via software will NOT trigger the counter function of T5.

For counter operation, pin T5IN must be configured as input. The maximum input frequency allowed in Counter Mode depends on the selected prescaler value. To ensure that a transition of the count input signal applied to T5IN is recognized correctly, its level must be held high or low for a minimum number of module clock cycles before it changes. This information can be found in **Section 16.4.6**.

Timer Concatenation

Using the toggle bit T6OTL as a clock source for the auxiliary timer in Counter Mode concatenates the core timer T6 with the auxiliary timer T5. This concatenation forms either a 32-bit or a 33-bit timer/counter, depending on which transition of T6OTL is selected to clock the auxiliary timer.

- **32-bit Timer/Counter:** If both a positive and a negative transition of T6OTL are used to clock the auxiliary timer, this timer is clocked on every overflow/underflow of the core timer T6. Thus, the two timers form a 32-bit timer.
- **33-bit Timer/Counter:** If either a positive or a negative transition of T6OTL is selected to clock the auxiliary timer, this timer is clocked on every second overflow/underflow of the core timer T6. This configuration forms a 33-bit timer (16-bit core timer + T6OTL + 16-bit auxiliary timer).
  As long as bit T6OTL is not modified by software, it represents the state of the internal toggle latch, and can be regarded as part of the 33-bit timer.

The count directions of the two concatenated timers are not required to be the same. This offers a wide variety of different configurations.

T6, which represents the low-order part of the concatenated timer, can operate in Timer Mode, Gated Timer Mode or Counter Mode in this case.

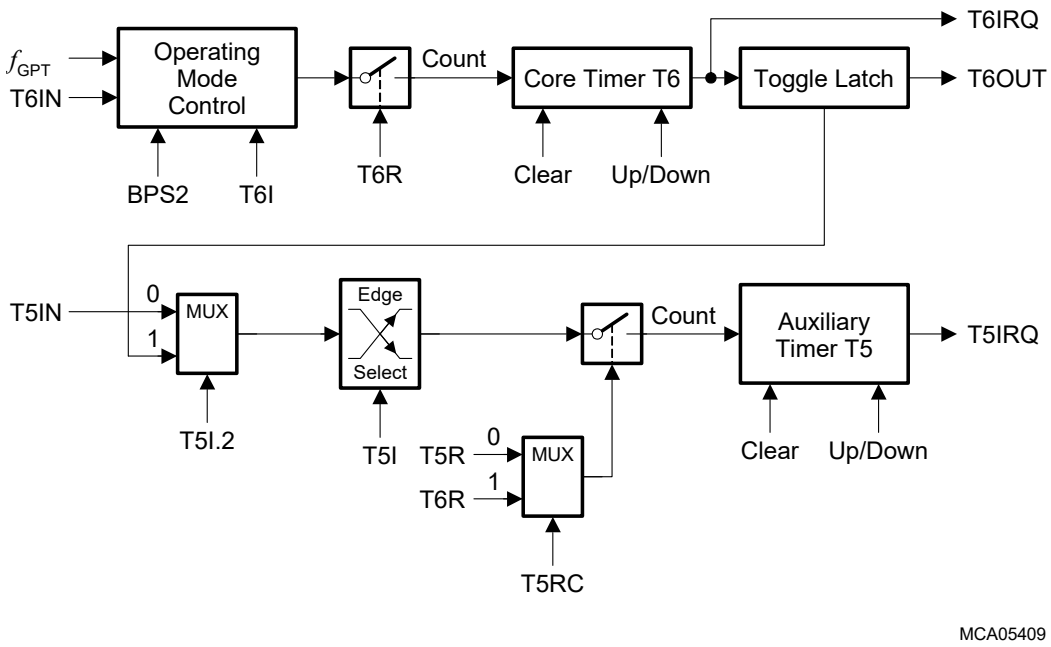


Figure 115 Concatenation of Core Timer T6 and Auxiliary Timer T5

16.4.5 GPT2 Register CAPREL Operating Modes

The Capture/Reload register CAPREL can be used to capture the contents of timer T5, or to reload timer T6. A special mode facilitates the use of register CAPREL for both functions at the same time. This mode allows frequency multiplication. The capture function is triggered by CAPIN, by T3IN and T3EUD, or by read GPT1 timers. The reload function is triggered by an overflow or underflow of timer T6.

In addition to the capture function, the capture trigger signal can also be used to clear the contents of timers T5 and T6 individually.

The functions of register CAPREL are controlled via several bit(field)s in the timer control registers T5CON and T6CON.

Capture/Reload Register CAPREL in Capture Mode

Capture mode for register CAPREL is selected by setting bit T5SC in control register T5CON (set bitfield CI in register T5CON to a non-zero value to select a trigger signal). In capture mode, the contents of the auxiliary timer T5 are latched into register CAPREL in response to a signal transition at the selected external input pin(s). Bit CT3 selects the external input line CAPIN or the input lines T3IN and/or T3EUD of GPT1 timer T3 as the source for a capture trigger. Either a positive, a negative, or both a positive and a negative transition at line CAPIN can be selected to trigger the capture function, or transitions on input T3IN or input T3EUD or both inputs, T3IN and T3EUD. The active edge is controlled by bitfield CI in register T5CON. **Table 271** summarizes these options.

Table 271 CAPREL Register Input Edge Selection

CT3	CI	Triggering Signal/Edge for Capture Mode
X	00_B	None. Capture Mode is disabled.
0	01_B	Positive transition (rising edge) on CAPIN.[1)]
0	10_B	Negative transition (falling edge) on CAPIN.
0	11_B	Any transition (rising or falling edge) on CAPIN.
1	01_B	Any transition (rising or falling edge) on T3IN.
1	10_B	Any transition (rising or falling edge) on T3EUD.
1	11_B	Any transition (rising or falling edge) on T3IN or T3EUD.

1) Rising edge must be selected if capturing is triggered by the internal GPT1 read signals (see register PISEL and **“Combined Capture Modes” on Page 518**).

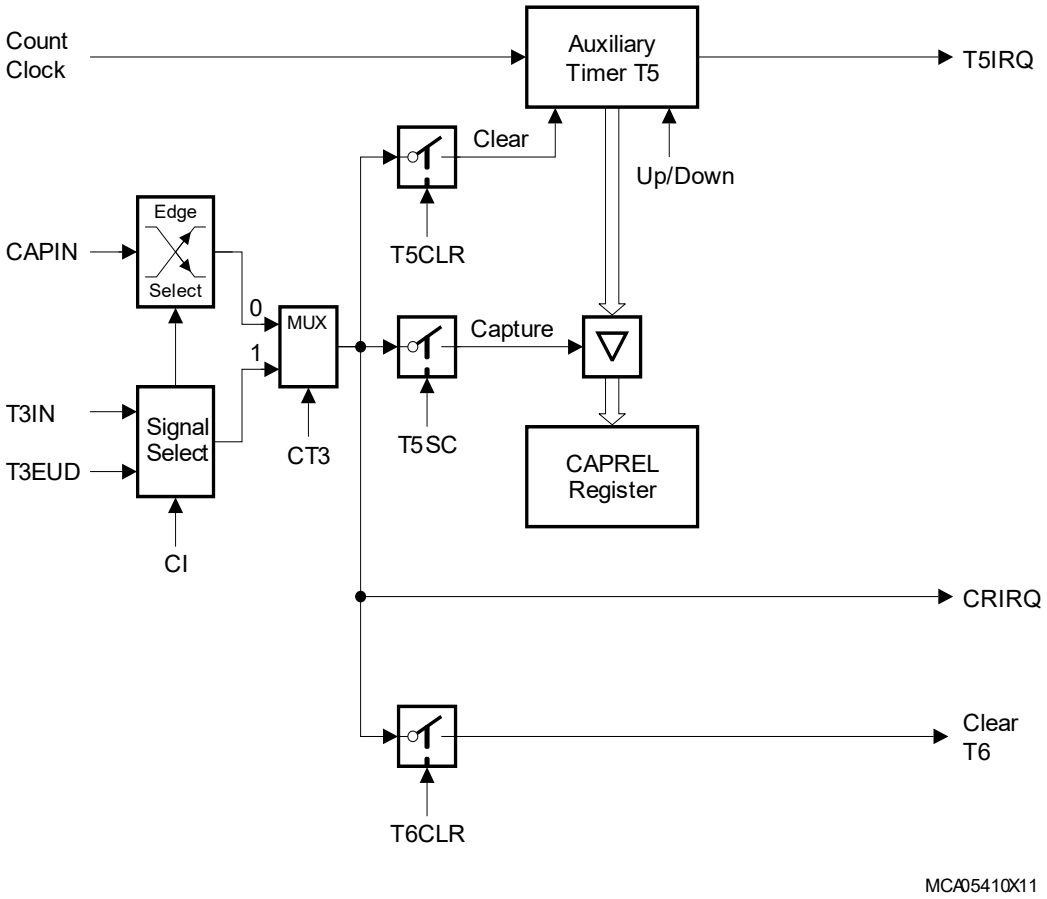


Figure 116 Capture/Reload Register CAPREL in Capture Mode

When a selected trigger is detected, the contents of the auxiliary timer T5 are latched into register CAPREL and the interrupt request line CRIRQ is activated. The same event can optionally clear timer T5 and/or timer T6. This option is enabled by bit T5CLR in register T5CON and bit T6CLR in register T6CON, respectively. If TxCLR = 0 the contents of timer Tx is not affected by a capture. If TxCLR = 1 timer Tx is cleared after the current timer T5 value has been latched into register CAPREL.

Note: Bit T5SC only controls whether or not a capture is performed. If T5SC is cleared the external input pin(s) can still be used to clear timer T5 and/or T6, or as external interrupt input(s). This interrupt is controlled by the CAPREL interrupt control register GPTM1IEN and GPTM1IRC.

When capture triggers T3IN or T3EUD are enabled (CT3 = 1), register CAPREL captures the contents of T5 upon transitions of the selected input(s). These values can be used to measure T3's input signals. This is useful, for example, when T3 operates in Incremental Interface Mode, in order to derive dynamic information (speed, acceleration) from the input signals.

For capture mode operation, the selected pins CAPIN, T3IN, or T3EUD must be configured as input. To ensure that a transition of a trigger input signal applied to one of these inputs is recognized correctly, its level must be held high or low for a minimum number of module clock cycles, detailed in **Section 16.4.6**.

Capture/Reload Register CAPREL in Reload Mode

Reload mode for register CAPREL is selected by setting bit T6SR in control register T6CON. In reload mode, the core timer T6 is reloaded with the contents of register CAPREL, triggered by an overflow or underflow of T6. This will not activate the interrupt request line CRIRQ associated with the CAPREL register. However, interrupt request line T6IRQ will be activated, indicating the overflow/underflow of T6.

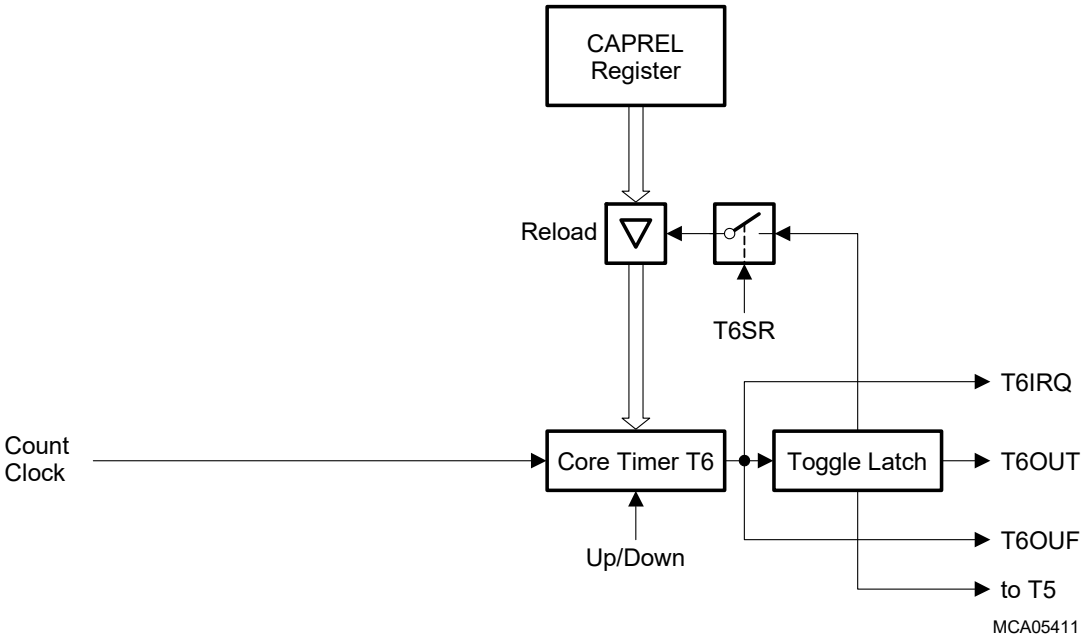


Figure 117 Capture/Reload Register CAPREL in Reload Mode

Capture/Reload Register CAPREL in Capture-And-Reload Mode

Since the reload function and the capture function of register CAPREL can be enabled individually by bits T5SC and T6SR, the two functions can be enabled simultaneously by setting both bits. This feature can be used to generate an output frequency that is a multiple of the input frequency.

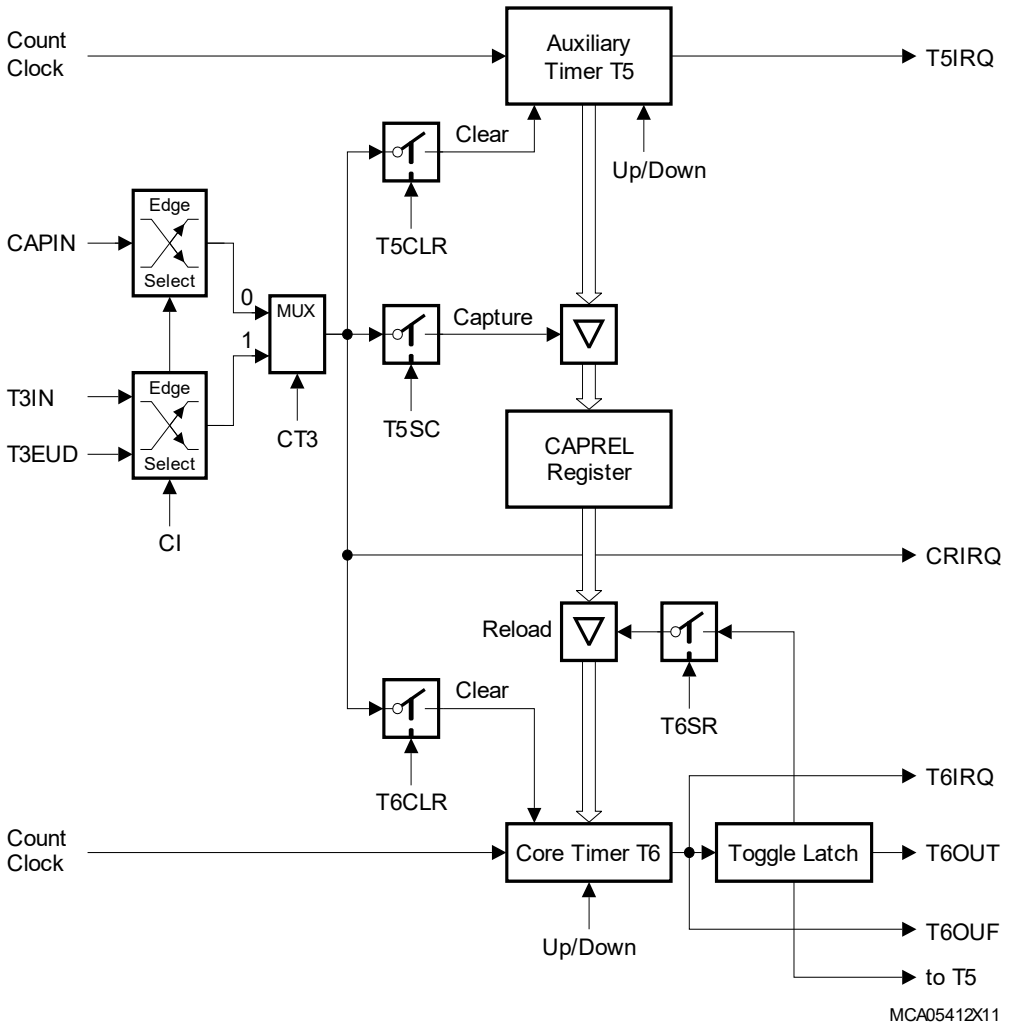


Figure 118 Capture/Reload Register CAPREL in Capture-And-Reload Mode

This combined mode can be used to detect consecutive external events which may occur aperiodically, but where a finer resolution, that means, more ‘ticks’ within the time between two external events is required.

For this purpose, the time between the external events is measured using timer T5 and the CAPREL register. Timer T5 runs in Timer Mode counting up with a frequency of e.g. $f_{GPT}/32$. The external events are applied to pin CAPIN. When an external event occurs, the contents of timer T5 are latched into register CAPREL and timer T5 is cleared (T5CLR = 1). Thus, register always contains the correct time between two events, measured in timer T5 increments. Timer T6, which runs in Timer Mode counting down with a frequency of e.g. $f_{GPT}/4$, uses the value in register CAPREL to perform a reload on underflow. This means, the value in register CAPREL represents the time between two underflows of timer T6, now measured in timer T6 increments. Since (in this example) timer T6 runs 8 times faster than timer T5, it will underflow 8 times within the time between two external events. Thus, the underflow signal of timer T6 generates 8 ‘ticks’. Upon each underflow, the interrupt request line T6IRQ will be activated and bit T6OTL will be toggled. The state of T6OTL may be output on pin T6OUT. This signal has 8 times more transitions than the signal which is applied to pin CAPIN.

Note: The underflow signal of Timer T6 can furthermore be used to clock one or more of the timers of the CAPCOM units, which gives the user the possibility to set compare events based on a finer resolution than that of the external events. This connection is accomplished via signal T6OUF.

Capture Correction

A certain deviation of the output frequency is generated by the fact that timer T5 will count actual time units (e.g. T5 running at 1 MHz will count up to the value $64_H/100_D$ for a 10 kHz input signal), while T6OTL will only toggle upon an underflow of T6 (i.e. the transition from 0000_H to $FFFF_H$). In the above mentioned example, T6 would count down from 64_H, so the underflow would occur after 101 timing ticks of T6. The actual output frequency then is 79.2 kHz, instead of the expected 80 kHz.

This deviation can be compensated for by using T6 overflows. In this case, T5 counts down and T6 counts up. Upon a signal transition on pin CAPIN, the count value in T5 is captured into CAPREL and T5 is cleared to 0000_H. In its next clock cycle, T5 underflows to $FFFF_H$, and continues to count down with the following clocks. T6 is reloaded from CAPREL upon an overflow, and continues to count up with its following clock cycles (8 times faster in the above example). In this case, T5 and T6 count the same number of steps with their respective internal count frequency.

In the above example, T5 running at 1 MHz will count down to the value $FF9C_H/-100_D$ for a 10 kHz input signal applied at CAPIN, while T6 counts up from $FF9C_H$ through $FFFF_H$ to 0000_H. So the overflow occurs after 100 timing ticks of T6, and the actual output frequency at T6OUT then is the expected 80 kHz.

However, in this case CAPREL does not directly contain the time between two CAPIN events, but rather its 2's complement. Software will have to convert this value, if it is required for the operation.

Combined Capture Modes

For incremental interface applications in particular, several timer features can be combined to obtain dynamic information such as speed, acceleration, or deceleration. The current position itself can be obtained directly from the timer register (T2, T3, T4).

The time information to determine the dynamic parameters is generated by capturing the contents of the free-running timer T5 into register CAPREL. Two trigger sources for this event can be selected:

- Capture trigger on sensor signal transitions
- Capture trigger on position read operations

Capturing on sensor signal transitions is available for timer T3 inputs. This mode is selected by setting bit CT3 and selecting the intended signal(s) via bitfield CI in register T5CON. CAPREL then indicates the time between two selected transitions (measured in T5 counts).

Capturing on position read operations is available for timers T2, T3, and T4. This mode is selected by clearing bit CT3 and selecting the rising edge via bitfield CI in register T5CON. Bitfield ISCAPIN in register PISEL then selects either a read access from T3 or a read access from any of T2 or T3 or T4. CAPREL then indicates the time between two read accesses.

These operating modes directly support the measurement of position and rotational speed. Acceleration and deceleration can then be determined by evaluating subsequent speed measurements.

16.4.6 GPT2 Clock Signal Control

All actions within the timer block GPT2 are triggered by transitions of its basic clock. This basic clock is derived from the module clock f_{GPT} by a basic block prescaler, controlled by bitfield BPS2 in register T6CON (see **Figure 89**). The count clock can be generated in two different ways:

- **Internal count clock**, derived from GPT2's basic clock via a programmable prescaler, is used for (gated) Timer Mode.
- **External count clock**, derived from the timer's input pin(s), is used for Counter Mode.

For both ways, the basic clock determines the maximum count frequency and the timer's resolution:

Table 272 Basic Clock Selection for Block GPT2

Block Prescaler[1)]	BPS2 = 01_B	BPS2 = 00_B[2)]	BPS2 = 11_B	BPS2 = 10_B
Prescaling Factor for GPT2: F(BPS2)	F(BPS2) = 2	F(BPS2) = 4	F(BPS2) = 8	F(BPS2) = 16
Maximum External Count Frequency	$f_{GPT}/4$	$f_{GPT}/8$	$f_{GPT}/16$	$f_{GPT}/32$
Input Signal Stable Time	$2 \times t_{GPT}$	$4 \times t_{GPT}$	$8 \times t_{GPT}$	$16 \times t_{GPT}$

1) Please note the non-linear encoding of bitfield BPS2.
2) Default after reset.

Note: When initializing the GPT2 block, and the block prescaler BPS2 in T6CON needs to be set to a value different from its reset value (00_B), it must be initialized first before any mode involving external trigger signals is configured. These modes include counter, capture, and reload mode. Otherwise, unintended count/capture/reload events may occur.
In this case (e.g. when changing BPS2 during operation of the GPT2 block), disable related interrupts before modification of BPS2, and afterwards clear the corresponding service request flags and re-initialize those registers (T5, T6, CAPREL) that might be affected by a count/capture/reload event.

Internal Count Clock Generation

In Timer Mode and Gated Timer Mode, the count clock for each GPT2 timer is derived from the GPT2 basic clock by a programmable prescaler, controlled by bitfield TxI in the respective timer's control register TxCON.

The count frequency f_{Tx} for a timer Tx and its resolution r_{Tx} are scaled linearly with lower clock frequencies, as can be seen from the following formula:

$$f_{Tx} = \frac{f_{GPT}}{F(BPS2) \bullet 2^{<TxI>}} \qquad r_{Tx}[\mu s] = \frac{F(BPS2) \bullet 2^{<TxI>}}{f_{GPT}[MHz]} \tag{16.2}$$

The effective count frequency depends on the common module clock prescaler factor F(BPS2) as well as on the individual input prescaler factor $2^{<TxI>}$. **Table 281** summarizes the resulting overall divider factors for a GPT2 timer that result from these cascaded prescalers.

Table 273 lists GPT2 timer's parameters (such as count frequency, resolution, and period) resulting from the selected overall prescaler factor and the module clock f_{GPT}. Note that some numbers may be rounded.

Table 273 GPT2 Timer Parameters

System Clock = 10 MHz			Overall Divider Factor	System Clock = 40 MHz		
Frequency	Resolution	Period		Frequency	Resolution	Period
5.0 MHz	200 ns	13.11 ms	2	20.0 MHz	50 ns	3.28 ms
2.5 MHz	400 ns	26.21 ms	4	10.0 MHz	100 ns	6.55 ms
1.25 MHz	800 ns	52.43 ms	8	5.0 MHz	200 ns	13.11 ms
625.0 kHz	1.6 µs	104.9 ms	16	2.5 MHz	400 ns	26.21 ms
312.5 kHz	3.2 µs	209.7 ms	32	1.25 MHz	800 ns	52.43 ms
156.25 kHz	6.4 µs	419.4 ms	64	625.0 kHz	1.6 µs	104.9 ms
78.125 kHz	12.8 µs	838.9 ms	128	312.5 kHz	3.2 µs	209.7 ms
39.06 kHz	25.6 µs	1.678 s	256	156.25 kHz	6.4 µs	419.4 ms
19.53 kHz	51.2 µs	3.355 s	512	78.125 kHz	12.8 µs	838.9 ms
9.77 kHz	102.4 µs	6.711 s	1024	39.06 kHz	25.6 µs	1.678 s
4.88 kHz	204.8 µs	13.42 s	2048	19.53 kHz	51.2 µs	3.355 s

External Count Clock Input

The external input signals of the GPT2 block are sampled with the GPT2 basic clock (see **Figure 89**). To ensure that a signal is recognized correctly, its current level (high or low) must be held active for at least one complete sampling period, before changing. A signal transition is recognized if two subsequent samples of the input signal represent different levels. Therefore, a minimum of two basic clock periods are required for the sampling of an external input signal. Thus, the maximum frequency of an input signal must not be higher than half the basic clock.

Table 274 summarizes the resulting requirements for external GPT2 input signals.

Table 274 GPT2 External Input Signal Limits

GPT2 Basic Clock = 10 MHz		Input Frequ. Factor	GPT2 Divider BPS2	Input Phase Duration	GPT2 Basic Clock = 40 MHz	
Max. Input Frequency	Min. Level Hold Time				Max. Input Frequency	Min. Level Hold Time
2.5 MHz	200 ns	$f_{GPT}/4$	01_B	$2 \times t_{GPT}$	10.0 MHz	50 ns
1.25 MHz	400 ns	$f_{GPT}/8$	00_B	$4 \times t_{GPT}$	5.0 MHz	100 ns
625.0 kHz	800 ns	$f_{GPT}/16$	11_B	$8 \times t_{GPT}$	2.5 MHz	200 ns
312.5 kHz	1.6 µs	$f_{GPT}/32$	10_B	$16 \times t_{GPT}$	1.25 MHz	400 ns

These limitations are valid for all external input signals to GPT2, including the external count signals in Counter Mode and the gate input signals in Gated Timer Mode.

16.4.7 Interrupt Control for GPT2 Timers and CAPREL

When a timer overflows from $FFFF_H$ to 0000_H (when counting up), or when it underflows from 0000_H to $FFFF_H$ (when counting down), its interrupt request flag in register GPT2_T5 or GPT2_T6I will be set. This will cause an interrupt to the respective timer interrupt vector, if the respective interrupt enable bit is set.

Whenever a transition according to the selection in bit field CI is detected at pin CAPIN, interrupt request flag in register GPT12_CR is set. Setting any request flag will cause an interrupt to the respective timer or CAPREL interrupt vector, if the respective interrupt enable bit is set.

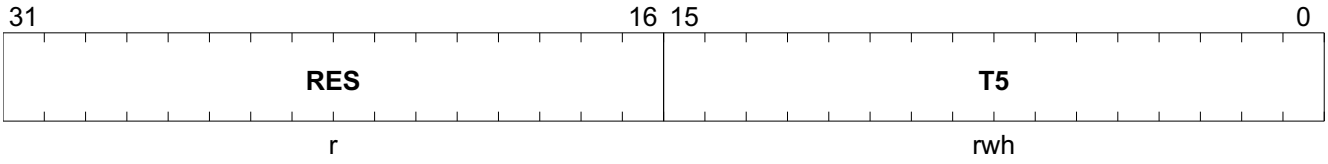
There is an interrupt control register for each of the two timers (T5, T6) and for the CAPREL register. All interrupt control registers have the same structure described in section Interrupt Control.

16.4.8 GPT2 Registers

16.4.8.1 GPT2 Timer Registers

Timer 5 Count Register

GPT12E_T5	Offset	Reset Value
Timer 5 Count Register	$2C_H$	see Table 275



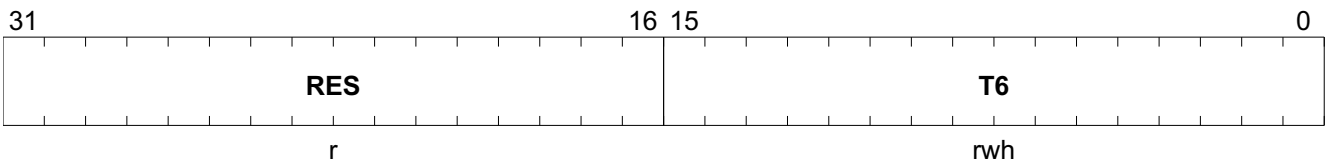
Field	Bits	Type	Description
RES	31:16	r	Reserved
T5	15:0	rwh	Timer T5 Current Value Contains the current value of the timer T2

Table 275 RESET of GPT12E_T5

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Timer 6 Count Register

GPT12E_T6	Offset	Reset Value
Timer 6 Count Register	30_H	see Table 276



Field	Bits	Type	Description
RES	31:16	r	Reserved

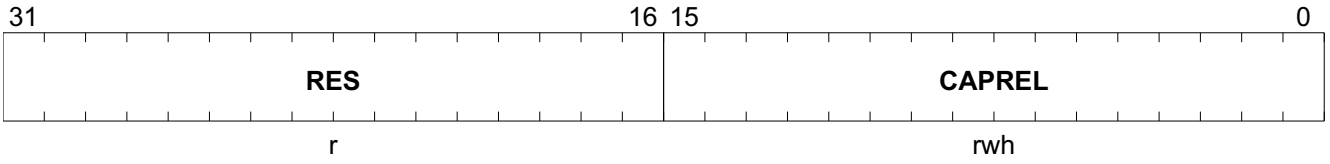
Field	Bits	Type	Description
T6	15:0	rwh	**Timer T6 Current Value** Contains the current value of the timer T6

Table 276 RESET of GPT12E_T6

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Capture/Reload Register

GPT12E_CAPREL	Offset	Reset Value
Capture/Reload Register	**$1C_H$**	**see Table 277**



Field	Bits	Type	Description
RES	31:16	r	**Reserved**
CAPREL	15:0	rwh	**Current reload value or Captured value** Contains the current value of the timer CAPREL register

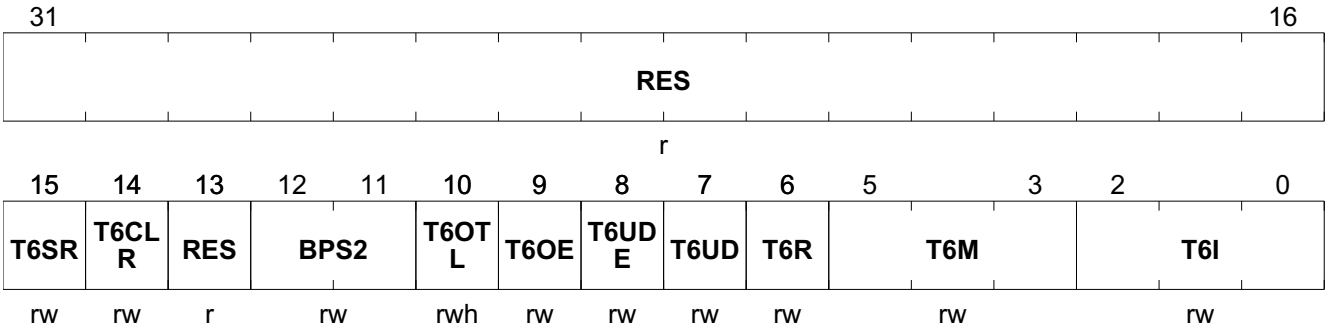
Table 277 RESET of GPT12E_CAPREL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

16.4.8.2 GPT2 Timer Control Registers

GPT2 Core Timer T6 Control Register

GPT12E_T6CON	Offset	Reset Value
Timer T6 Control Register	**18_H**	**see Table 278**



Field	Bits	Type	Description
RES	31:16	r	**Reserved**
T6SR	15	rw	**Timer T6 Reload Mode Enable** 0_B **Disabled**, Reload from register CAPREL disabled 1_B **Enabled**, Reload from register CAPREL enabled
T6CLR	14	rw	**Timer T6 Clear Enable Bit** 0_B **Not cleared**, Timer T6 is not cleared on a capture event 1_B **Cleared**, Timer T6 is cleared on a capture event
RES	13	r	**Reserved**
BPS2	12:11	rw	**GPT2 Block Prescaler Control** Select basic clock for block GPT1 (see also **Section 16.4.6**) 00_B **4**, $f_{GPT}/4$ 01_B **2**, $f_{GPT}/2$ 10_B **16**, $f_{GPT}/16$ 11_B **8**, $f_{GPT}/8$
T6OTL	10	rwh	**Timer T6 Overflow Toggle Latch** Toggles on each overflow/underflow of T6. Can be set or cleared by software (see separate description)
T6OE	9	rw	**Overflow/Underflow Output Enable** 0_B **Disabled**, Alternate Output Function Disabled 1_B **T6OUT**, State of T6 toggle latch is output on pin T6OUT
T6UDE	8	rw	**Timer T6 External Up/Down Enable**[1)] 0_B **T6UD**, Count direction is controlled by bit T6UD; input T6EUD is disconnected 1_B **T6EUD**, Count direction is controlled by input T6EUD
T6UD	7	rw	**Timer T6 Up/Down Control**[1)] 0_B **Up**, Timer T3 counts up 1_B **Down**, Timer T3 counts down
T6R	6	rw	**Timer T6 Input Run Bit** 0_B **Stop**, Timer T3 stops 1_B **Run**, Timer T3 runs

Field	Bits	Type	Description
T6M	5:3	rw	**Timer T6 Mode Control** 000_B **Timer Mode**, 001_B **Counter Mode**, 010_B **Gated low**, Gated Timer Mode with gate active low 011_B **Gated high**, Gated Timer Mode with gate active high 100_B **Reserved**, Do not use this combination 101_B **Reserved**, Do not use this combination 110_B **Reserved**, Do not use this combination 111_B **Reserved**, Do not use this combination
T6I	2:0	rw	**Timer T6 Input Parameter Selection** Depends on the operating mode, see respective sections for encoding: **Table 281** for Timer Mode and Gated Timer Mode **Table 283** for Counter Mode

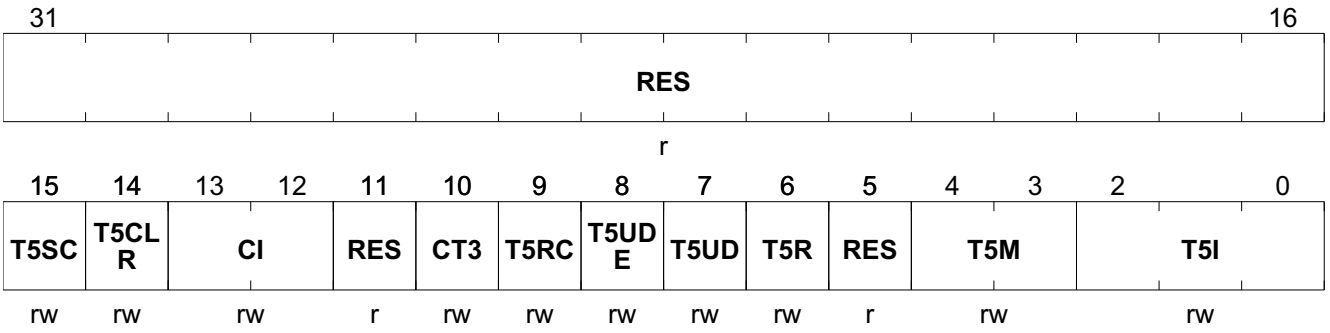
1) See **Table 280** for encoding of bits T6UD and T6UDE.

Table 278 RESET of GPT12E_T6CON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

GPT2 Auxiliary Timer T5 Control Register

GPT12E_T5CON	Offset	Reset Value
Timer T5 Control Register	14_H	see Table 279



Field	Bits	Type	Description
RES	31:16	r	**Reserved**
T5SC	15	rw	**Timer T5 Capture Mode Enable** 0_B **Disabled**, Capture into register CAPREL disabled 1_B **Enabled**, Capture into register CAPREL enabled
T5CLR	14	rw	**Timer T5 Clear Enable Bit** 0_B **Not cleared**, Timer T5 is not cleared on a capture event 1_B **Cleared**, Timer T5 is cleared on a capture event

Field	Bits	Type	Description
CI	13:12	rw	**Register CAPREL Capture Trigger Selection**[1)] 00_B **Disabled**, Capture disabled 01_B **Positive**, Positive transition (rising edge) on CAPIN[2)] or any transition on T3IN 10_B **Negative**, Negative transition (falling edge) on CAPIN or any transition on T3EUD 11_B **Any**, Any transition (rising or falling edge) on CAPIN or any transition on T3IN or T3EUD
RES	11	r	**Reserved**
CT3	10	rw	**Timer T3 Capture Trigger Enable** 0_B **CAPIN**, Capture trigger from input line CAPIN 1_B **T3IN**, Capture trigger from T3 input lines T3IN and/or T3EUD
T5RC	9	rw	**Timer T5 Remote Control** 0_B **T5R**, Timer T5 is controlled by its own run bit T5R 1_B **T6R**, Timer T5 is controlled by the run bit T6R of core timer T6, not by bit T5R
T5UDE	8	rw	**Timer T5 External Up/Down Enable**[3)] 0_B **T5UD**, Count direction is controlled by bit T5UD; input T5EUD is disconnected 1_B **T5EUD**, Count direction is controlled by input T5EUD
T5UD	7	rw	**Timer T2 Up/Down Control**[3)] 0_B **Up**, Timer T5 counts up 1_B **Down**, Timer T5 counts down
T5R	6	rw	**Timer T5 Run Bit** 0_B **Stop**, Timer T5 stops 1_B **Run**, Timer T5 runs
RES	5	r	**Reserved** Contains the current value of the CAPREL register
T5M	4:3	rw	**Timer T5 Input Mode Control** 00_B **Timer Mode**, 01_B **Counter Mode**, 10_B **Gated low**, Gated Timer Mode with gate active low 11_B **Gated high**, Gated Timer Mode with gate active high
T5I	2:0	rw	**Timer T5 Input Parameter Selection** Depends on the operating mode, see respective sections for encoding: **Table 281** for Timer Mode and Gated Timer Mode **Table 282** for Counter Mode

1) To define the respective trigger source signal, also bit CT3 must be regarded (see **Table 271**).

2) Rising edge must be selected if capturing is triggered by the internal GPT1 read signals (see register PISEL and **"Combined Capture Modes" on Page 518**).

3) See **Table 280** for encoding of bits T5UD and T5UDE.

Table 279 RESET of GPT12E_T5CON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

16.4.8.3 Encoding

Encoding of Timer Count Direction Control

Table 280 GPT2 Timer Count Direction Control

Pin TxEUD	Bit TxUDE	Bit TxUD	Count Direction
X	0	0	Count Up
X	0	1	Count Down
0	1	0	Count Up
1	1	0	Count Down
0	1	1	Count Down
1	1	1	Count Up

Timer Mode and Gated Timer Mode: Encoding of Overall Prescaler Factor

Table 281 GPT2 Overall Prescaler Factors for Internal Count Clock (Timer Mode and Gated Timer Mode)

Individual Prescaler for Tx	Common Prescaler for Module Clock[1)]			
	BPS2 = 01_B	**BPS2 = 00_B**	**BPS2 = 11_B**	**BPS2 = 10_B**
TxI = 000_B	2	4	8	16
TxI = 001_B	4	8	16	32
TxI = 010_B	8	16	32	64
TxI = 011_B	16	32	64	128
TxI = 100_B	32	64	128	256
TxI = 101_B	64	128	256	512
TxI = 110_B	128	256	512	1024
TxI = 111_B	256	512	1024	2048

1) Please note the non-linear encoding of bitfield BPS2.

Counter Mode: Encoding of Input Edge Selection

Table 282 GPT2 Auxiliary Timer T5 Input Edge Selection (Counter Mode)

T5I	Triggering Edge for Counter Increment/Decrement
$X00_B$	None. Counter T5 is disabled
001_B	Positive transition (rising edge) on T5IN
010_B	Negative transition (falling edge) on T5IN
011_B	Any transition (rising or falling edge) on T5IN

Table 282 GPT2 Auxiliary Timer T5 Input Edge Selection (Counter Mode) (cont'd)

T5I	Triggering Edge for Counter Increment/Decrement
101_B	Positive transition (rising edge) of T6 toggle latch T6OTL
110_B	Negative transition (falling edge) of T6 toggle latch T6OTL
111_B	Any transition (rising or falling edge) of T6 toggle latch T6OTL

Table 283 GPT2 Core Timer T6 Input Edge Selection (Counter Mode)

T6I	Triggering Edge for Counter Increment/Decrement
000_B	None. Counter T6 is disabled
001_B	Positive transition (rising edge) on T6IN
010_B	Negative transition (falling edge) on T6IN
011_B	Any transition (rising or falling edge) on T6IN
$1XX_B$	Reserved. Do not use this combination

16.4.8.4 GPT2 Timer and CAPREL Interrupt Control Registers

The Interrupt control register for GPT2 and CAPREL are located in the SCU.

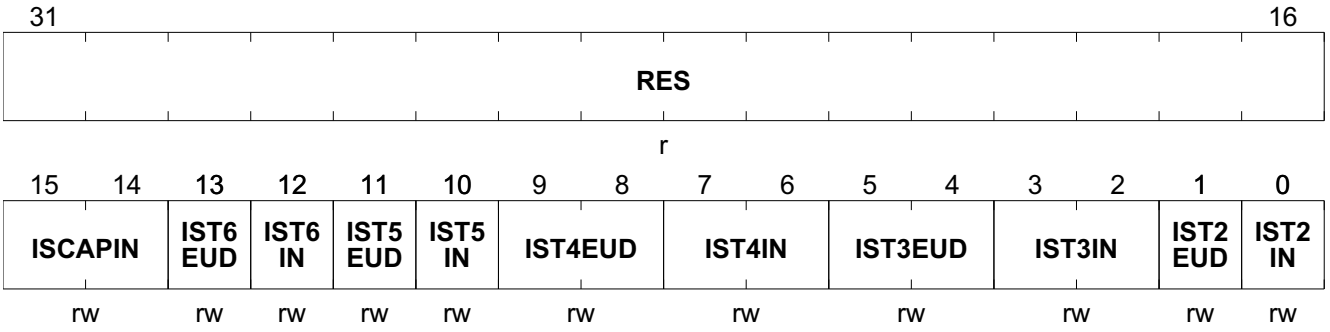
16.5 Miscellaneous GPT12 Registers

The following registers are not assigned to a specific timer block. They control general functions and/or give general information.

Port Input Select Register

Register PISEL selects timer input signal from several sources under software control.

GPT12E_PISEL	Offset	Reset Value
Port Input Select Register	04_H	see Table 284



Field	Bits	Type	Description
RES	31:16	r	**Reserved**
ISCAPIN	15:14	rw	**Input Select for CAPIN** 00_B **CAPINA**, Signal CAPINA is selected 01_B **CAPINB**, Signal CAPINB is selected 10_B **CAPINC**, Signal CAPINC (Read trigger from T3) is selected 11_B **CAPIND**, Signal CAPIND (Read trigger from T2 or T3 or T4) is selected
IST6EUD	13	rw	**Input Select for T6EUD** 0_B **T6EUDA**, Signal T6EUDA is selected 1_B **T6EUDB**, Signal T6EUDB is selected
IST6IN	12	rw	**Input Select for T6IN** 0_B **T6INA**, Signal T6INA is selected 1_B **T6INB**, Signal T6INB is selected
IST5EUD	11	rw	**Input Select for T5EUD** 0_B **T5EUDA**, Signal T5EUDA is selected 1_B **T5EUDB**, Signal T5EUDB is selected
IST5IN	10	rw	**Input Select for T5IN** 0_B **T5INA**, Signal T5INA is selected 1_B **T5INB**, Signal T5INB is selected
IST4EUD	9:8	rw	**Input Select for TEUD** 00_B **T4EUDA**, Signal T4EUDA is selected 01_B **T4EUDB**, Signal T4EUDB is selected 10_B **T4EUDC**, Signal T4EUDC is selected 11_B **T4EUDD**, Signal T4EUDD is selected

Field	Bits	Type	Description
IST4IN	7:6	rw	**Input Select for T4IN** 00_B **T4INA**, Signal T4INA is selected 01_B **T4INB**, Signal T4INB is selected 10_B **T4INC**, Signal T4INC is selected 11_B **T4IND**, Signal T4IND is selected
IST3EUD	5:4	rw	**Input Select for T3EUD** 00_B **T3EUDA**, Signal T3EUDA is selected 01_B **T3EUDB**, Signal T3EUDB is selected 10_B **T3EUDC**, Signal T3EUDC is selected 11_B **T3EUDD**, Signal T3EUDD is selected
IST3IN	3:2	rw	**Input Select for T3IN** 00_B **T3INA**, Signal T3INA is selected 01_B **T3INB**, Signal T3INB is selected 10_B **T3INC**, Signal T3INC is selected 11_B **T3IND**, Signal T3IND is selected
IST2EUD	1	rw	**Input Select for T2EUD** 0_B **T2EUDA**, Signal T2EUDA is selected 1_B **T2EUDB**, Signal T2EUDB is selected
IST2IN	0	rw	**Input Select for T2IN** 0_B **T2INA**, Signal T2INA is selected 1_B **T2INB**, Signal T2INB is selected

Table 284 RESET of GPT12E_PISEL

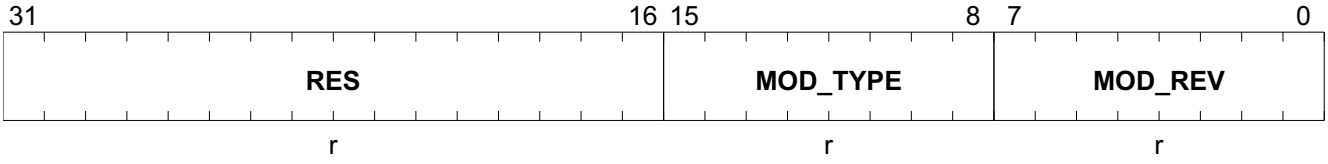
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Note: PISEL's reset value represents the connections available in previous versions.

Module Identification Register

Register ID indicates the module version.

GPT12E_ID	Offset	Reset Value
Module Identification Register	**00_H**	**see Table 285**



Field	Bits	Type	Description
RES	31:16	r	**Reserved**

Field	Bits	Type	Description
MOD_TYPE	15:8	r	**Module Identification Number** This bitfield defines the module identification number (58_H = GPT12E)
MOD_REV	7:0	r	**Module Revision Number** MOD:_REV defines the revision number. The value of a module revision starts with 01_H (first revision)

Table 285 RESET of GPT12E_ID

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00005804_H	RESET_TYPE_3		

16.6 Implementation of the GPT12 Module

This chapter describes the implementation of the GPT12 module in the TLE985xQX device.

16.6.1 Module Connections

Besides the described intra-module connections, the timer unit blocks GPT1 and GPT2 are connected to their environment in two basic ways:

- **Internal connections** interface the timers with on-chip resources such as clock generation unit, interrupt controller, or other timers.
  The GPT module is clocked with the TLE985xQX system clock, so $f_{GPT} = f_{SYS}$.
- **External connections** interface the timers with external resources via port pins.

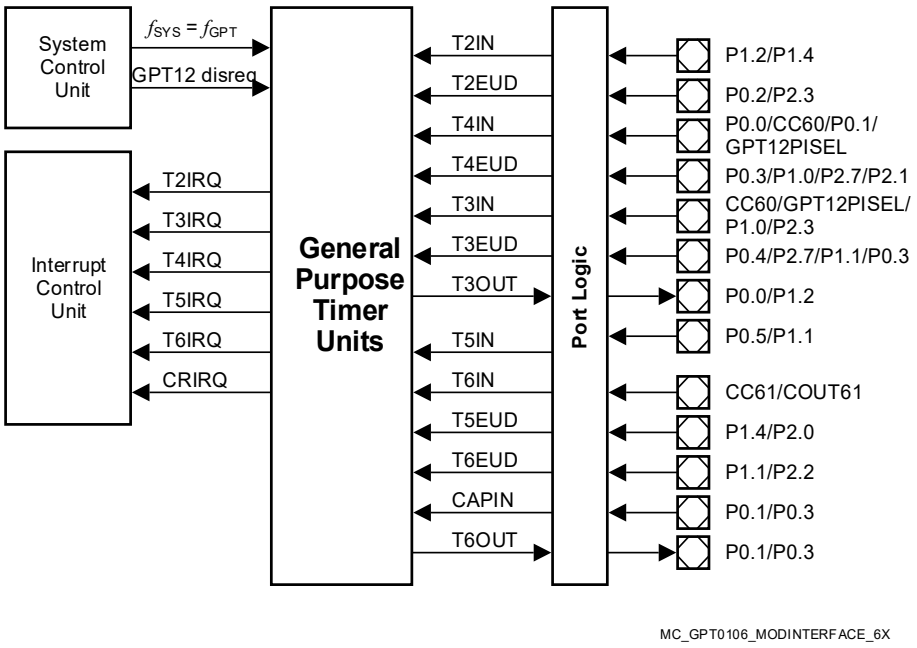


Figure 119 GPT Module Interfaces

Note: The GPT12E output signal 'T6OFL' is connected to the CAPCOM2 input 'TOUF' and to the GSC.

The following table **Table 286** (GPT12) shows the digital connections of the GPT12 module with other modules or pins in the TLE985xQX device.

Table 286 GPT12 Digital Connections in TLE985xQX

Signal	from/to Module	I/O to GPT	Can be used to/as
T2INA	P1.2	I	count input signals for timer T2
T2INB	P1.4	I	
T2EUDA	P0.2	I	direction input signals for timer T2
T2EUDB	P2.3	I	
T2IRQ	ICU/SCU	O	interrupt request from timer T2

Table 286 GPT12 Digital Connections in TLE985xQX

Signal	from/to Module	I/O to GPT	Can be used to/as
T3INA	CC60	I	count input signals for timer T3
T3INB	GPT12PISEL	I	
T3INC	P1.0	I	
T3IND	P2.3	I	
T3EUDA	P0.4	I	direction input signals for timer T3
T3EUDB	P2.7	I	
T3EUDC	P1.1	I	
T3EUDD	P0.3	I	
T3OUT_0, _1	P0.0	O	count output signal for timer T3
	P1.2	O	
T3IRQ	ICU/SCU	O	interrupt request from timer T3
T4INA	P0.0	I	count input signals for timer T4
T4INB	CC60	I	
T4INC	P0.1	I	
T4IND	GPT12PISEL	I	
T4EUDA	P0.3	I	direction input signals for timer T4
T4EUDB	P1.0	I	
T4EUDC	P2.7	I	
T4EUDD	P2.1	I	
T4IRQ	ICU/SCU	O	interrupt request from timer T4
T5INA	P0.5	I	count input signals for timer T5
T5INB	P1.1	I	
T5EUDA	P1.4	I	direction input signals for timer T5
T5EUDB	P2.0	I	
T5IRQ	ICU/SCU	O	interrupt request from timer T5
T6INA	CC61	I	count input signals for timer T6
T6INB	COUT61	I	
T6EUDA	P1.1	I	direction input signals for timer T6
T6EUDB	P2.2	I	
T6OUT_1, _0	P0.3	O	count output signal for timer T6
	P0.1	O	
T6IRQ	ICU/SCU	O	interrupt request from timer T6
T6OFL	P0.3	O	over/under-flow signal from timer T6
CAPINA	P0.1	I	input capture signals
CAPINB	P0.3	I	
CAPINC	read trigger from T3	I	

Table 286 GPT12 Digital Connections in TLE985xQX

Signal	from/to Module	I/O to GPT	Can be used to/as
CAPIND	read trigger from T2 or T3 or T4	I	
CRIRQ	ICU/SCU	O	interrupt request from capture control

Port Control

Port pins to be used for timer input signals must be switched to input (bitfield PC in the respective port control register must be $0xxx_B$) and must be selected via register PISEL.

Port pins to be used for timer output signals must be switched to output and the alternate timer output signal must be selected (bitfield PC in the respective port control register must be $1xxx_B$).

Note: For a description of the port control registers, please refer to chapter “Parallel Ports”.

Interrupts

The Mod_Name 12 has six interrupt request lines.

Interrupt nodes to be used for timer interrupt requests must be enabled and programmed to a specific interrupt level.

Debug Details

While the module GPT is disabled, its registers can still be read. While disabled the following registers can be written: PISEL, T5CON.

17 Timer2 and Timer21

This chapter describes the Timer2 and Timer21. Each timer is a 16-bit timer which additionally can function as a counter. Each Timer 2 module also provides a single channel 16-bit capture.

17.1 Features

- 16-bit auto-reload mode
  - selectable up or down counting
- One channel 16-bit capture mode
- Baud-rate generator for U(S)ART

17.2 Introduction

Two functionally identical timers are implemented: Timer 2 and 21. The description refers to Timer 2 only, but applies to Timer 21 as well.

The timer modules are general purpose 16-bit timer. Timer 2 can function as a timer or counter in each of its modes. As a timer, it counts with an input clock of $f_{sys}/12$ (if prescaler is disabled). As a counter, Timer 2 counts 1-to-0 transitions on pin T2. In the counter mode, the maximum resolution for the count is $f_{sys}/24$ (if prescaler is disabled).

Note: "Timer 2" is generally referred in the following description which is applicable to each of the Timer2 and Timer21.

17.2.1 Timer2 and Timer21 Modes Overview

Table 287 Port Registers

Mode	Description
Auto-reload	**Up/Down Count Disabled** • Count up only • Start counting from 16-Bit reload value, overflow at $FFFF_H$ • Reload event configurable for trigger by overflow condition only, or by negative/positive edge at input pin T2EX as well • Programmable reload value in register RC2 • Interrupt is generated with reload events.
Auto-reload	**Up/Down Count Enabled** • Count up or down, direction determined by level at input pin T2EX • No interrupt is generated • Count up – Start counting from 16-Bit reload value, overflow at $FFFF_H$ – Reload event triggered by overflow condition – Programmable reload value in register RC2 • Count down – Start counting from $FFFF_H$, underflow at value defined in register RC2 – Reload event triggered by underflow condition – Reload value fixed at $FFFF_H$
Channel capture	• Count up only • Start counting from 0000_H, overflow at $FFFF_H$ • Reload event triggered by overflow condition • Reload value fixed at 0000_H • Capture event triggered by falling/rising edge at pin T2EX • Captured timer value stored in register RC2 • Interrupt is generated with reload or capture event

Timer 2 can be started by using TR2 bit by hardware or software. Timer 2 can be started by setting TR2 bit by software. If bit T2RHEN is set, Timer 2 can be started by hardware. Bit T2REGS defines the event on pin T2EX: falling edge or rising edge, that can set the run bit TR2 by hardware. Timer 2 can only be stopped by resetting TR2 bit by software.

17.3 Functional Description

17.3.1 Auto-Reload Mode

The auto-reload mode is selected when the bit CP_RL2 in register T2CON is zero. In the auto-reload mode, Timer 2 counts to an overflow value and then reloads its register contents with a 16-bit start value for a fresh

counting sequence. The overflow condition is indicated by setting bit TF2 in the T2CON register. This will then generate an interrupt request to the core. The overflow flag TF2 must be cleared by software.

The auto-reload mode is further classified into two categories depending upon the DCEN control bit.

17.3.1.1 Up/Down Count Disabled

If DCEN = 0, the up-down count selection is disabled. The timer, therefore, functions as a pure up counter/timer only. The operational block diagram is shown in **Figure 120**.

In this mode, if EXEN2 = 0, the timer starts to count up to a maximum of $FFFF_H$, once TR2 is set. Upon overflow, bit TF2 is set and the timer register is reloaded with the 16-bit reload value of the RC2 register. This reload value is chosen by software, prior to the occurrence of an overflow condition. A fresh count sequence is started and the timer counts up from this reload value as in the previous count sequence.

If EXEN2 = 1, the timer counts up to a maximum of $FFFF_H$ once TR2 is set. A 16-bit reload of the timer registers from register RC2 is triggered either by an overflow condition or by a negative/positive edge (chosen by T2MOD.EDGESEL) at input pin T2EX. If an overflow caused the reload, the overflow flag TF2 is set. If a negative/positive transition at pin T2EX caused the reload, bit EXF2 is set. In either case, an interrupt is generated to the core and the timer proceeds to its next count sequence. The EXF2 flag, similar to the TF2, must be cleared by software.

If bit T2RHEN is set, Timer 2 is started by first falling edge/rising edge at pin T2EX, which is defined by bit T2REGS. If bit EXEN2 is set, bit EXF2 is also set at the same point when Timer2 is started with the same falling edge/rising edge at pin T2EX, which is defined by bit EDGESEL. The reload will happen with the following negative/positive transitions at pin T2EX, which is defined by bit EDGESEL.

Note: In counter mode, if the reload via T2EX and the count clock T2 are detected simultaneously, the reload takes precedence over the count. The counter increments its value with the following T2 count clock.

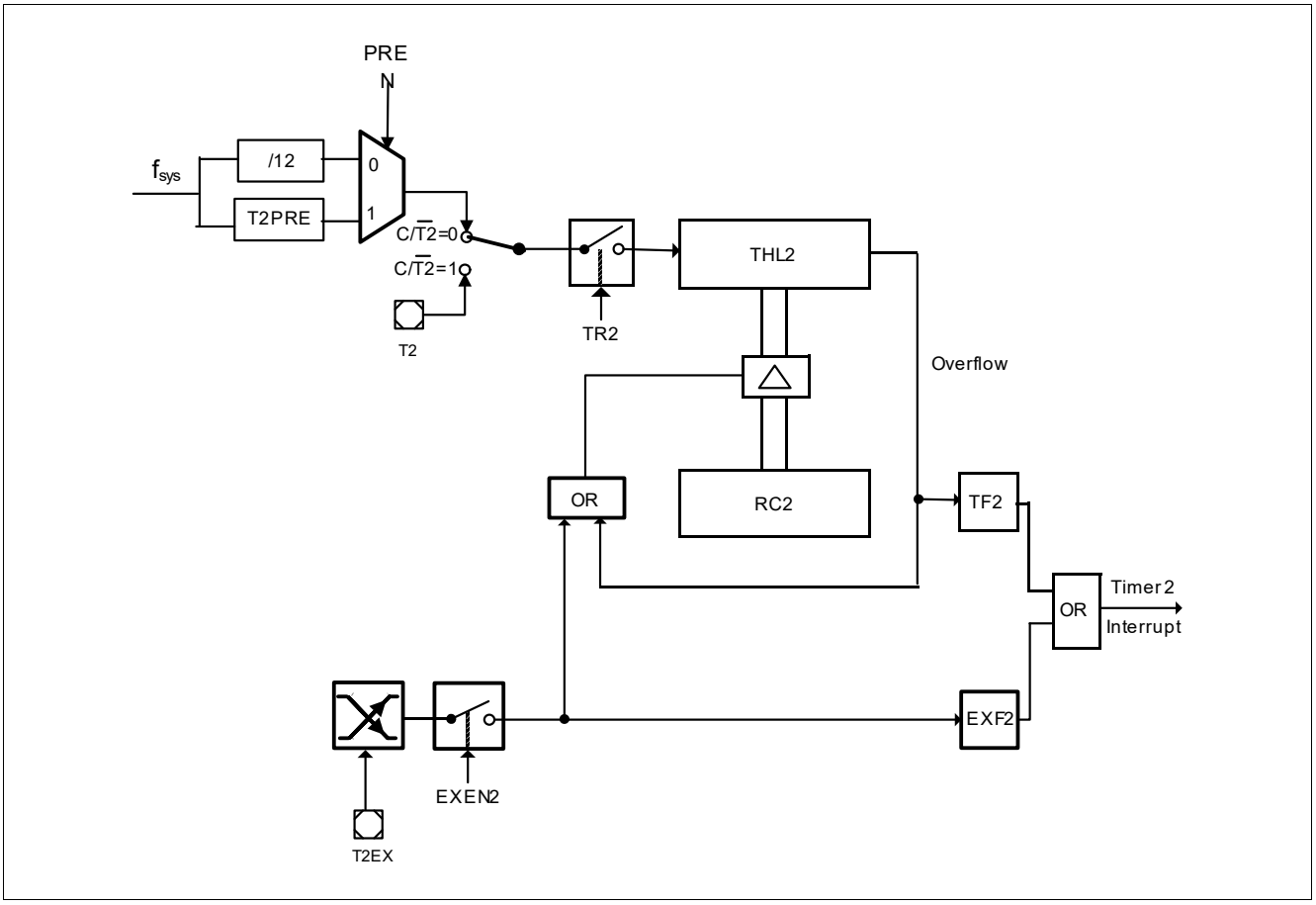


Figure 120 Auto-Reload Mode (DCEN = 0)

17.3.1.2 Up/Down Count Enabled

If DCEN = 1, the up-down count selection is enabled. The direction of count is determined by the level at input pin T2EX. The operational block diagram is shown in **Figure 121**.

A logic 1 at pin T2EX sets the Timer 2 to up counting mode. The timer, therefore, counts up to a maximum of $FFFF_H$. Upon overflow, bit TF2 is set and the timer register is reloaded with a 16-bit reload value of the RC2 register. A fresh count sequence is started and the timer counts up from this reload value as in the previous count sequence. This reload value is chosen by software, prior to the occurrence of an overflow condition.

A logic 0 at pin T2EX sets the Timer 2 to down counting mode. The timer counts down and underflows when the THL2 value reaches the value stored at register RC2. The underflow condition sets the TF2 flag and causes $FFFF_H$ to be reloaded into the THL2 register. A fresh down counting sequence is started and the timer counts down as in the previous counting sequence.

If bit T2RHEN is set, Timer 2 can only be started either by rising edge (T2REGS = 1) at pin T2EX and then do the up counting, or be started by falling edge (T2REGS = 0) at pin T2EX and then do the down counting.

In this mode, bit EXF2 toggles whenever an overflow or an underflow condition is detected. This flag, however, does not generate an interrupt request.

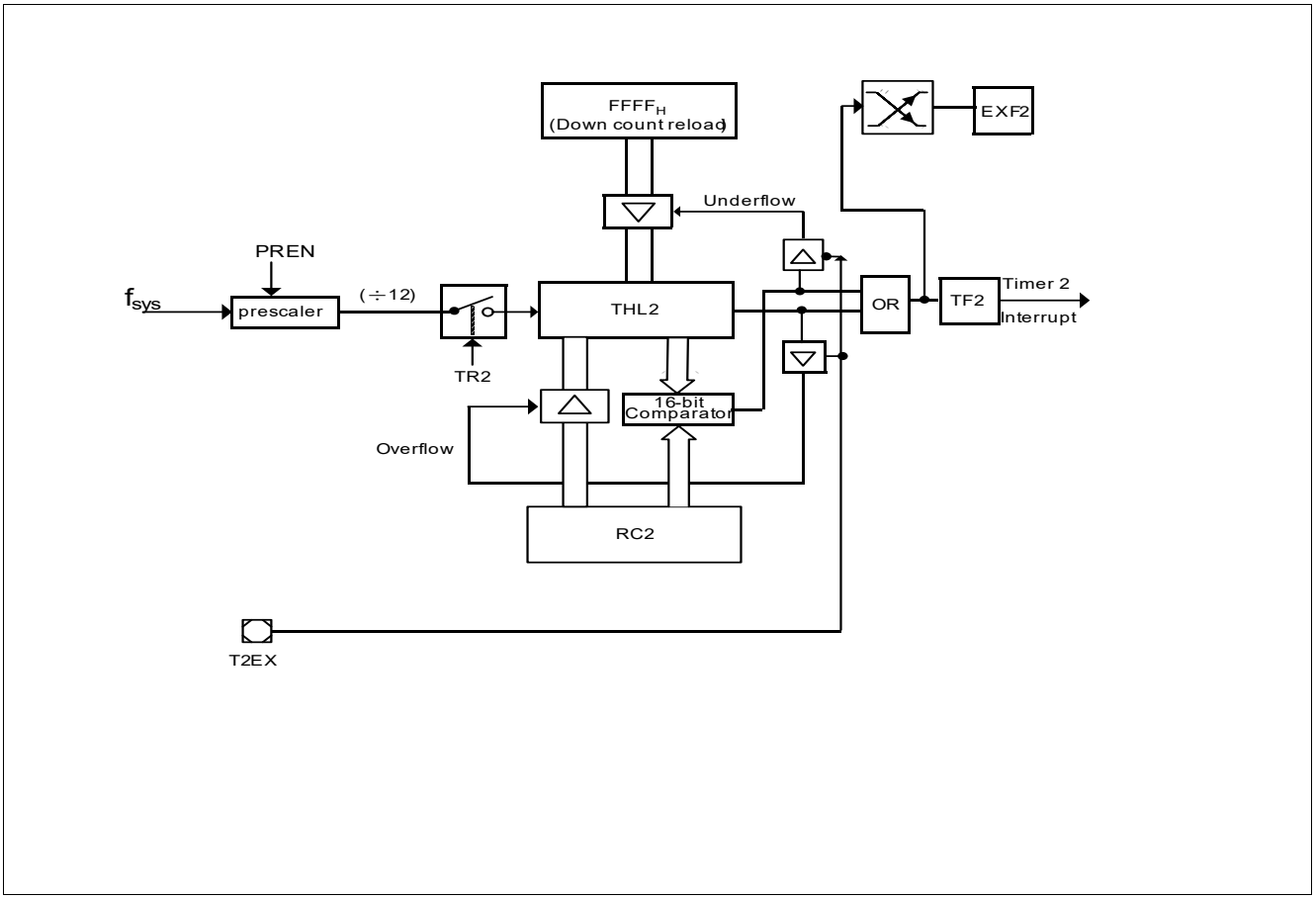


Figure 121 Auto-Reload Mode (DCEN = 1)

17.3.2 Capture Mode

In order to enter the 16-bit capture mode, bits CP_RL2 and EXEN2 in register T2CON must be set. In this mode, the down count function must remain disabled. The timer functions as a 16-bit timer or counter and always counts up to $FFFF_H$ and overflows. Upon an overflow condition, bit TF2 is set and the timer reloads its registers with 0000_H. The setting of TF2 generates an interrupt request to the core.

Additionally, with a falling/rising edge on pin T2EX (chosen by T2MOD.EDGESEL) the contents of the timer register (THL2) are captured into the RC2 register. The external input is sampled in every f_{sys} cycle. When a sampled input shows a low (high) level in one f_{sys} cycle and a high (low) in the next f_{sys} cycle, a transition is recognized. If the capture signal is detected while the counter is being incremented, the counter is first incremented before the capture operation is performed. This ensures that the latest value of the timer register is always captured.

If bit T2RHEN is set, Timer 2 is started by first falling edge/rising edge at pin T2EX, which is defined by bit T2REGS. If bit EXEN2 is set, bit EXF2 is also set at the same point when Timer2 is started with the same falling edge/rising edge at pin T2EX, which is defined by bit EDGESEL. The capture will happen with the following negative/positive transitions at pin T2EX, which is defined by bit EDGESEL.

When the capture operation is completed, bit EXF2 is set and can be used to generate an interrupt request. **Figure 122** describes the capture function of Timer 2.

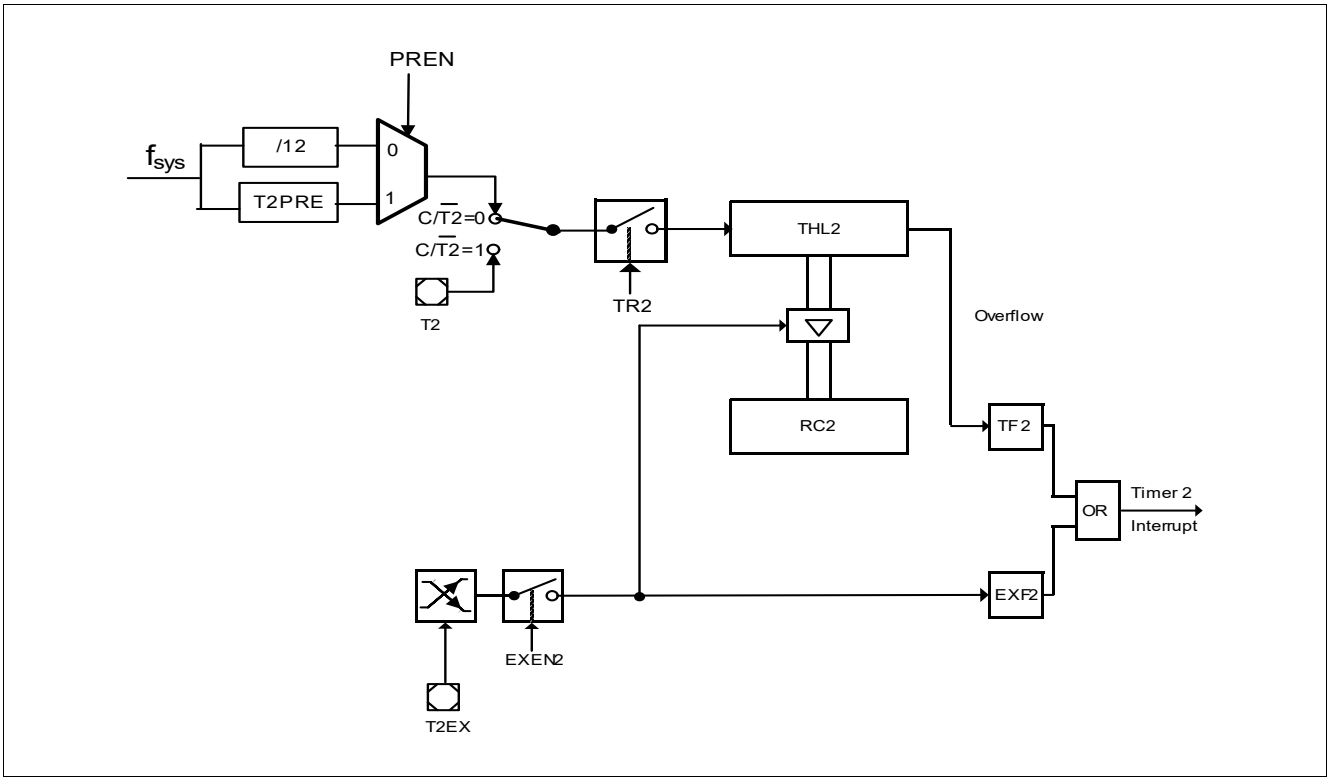


Figure 122 Capture Mode

17.3.3 Count Clock

The count clock for the auto-reload mode is chosen by the bit C_T2 in register T2CON. If C_T2 = 0, a count clock of $f_{sys}/12$ (if prescaler is disabled) is used for the count operation.

If C_T2 = 1, Timer 2 behaves as a counter that counts 1-to-0 transitions of input pin T2. The counter samples pin T2 over 2 f_{sys} cycles. If a 1 was detected during the first clock and a 0 was detected in the following clock, then the counter increments by one. Therefore, the input levels should be stable for at least 1 clock.

If bit T2RHEN is set, Timer 2 can be started by the falling edge/rising edge on pin T2EX, which is defined by bit T2REGS.

Note: If pin T2 is not connected, counting clock function on pin T2 cannot be used.

17.3.4 Interrupt Generation

When an interrupt event happened, the corresponding interrupt flag bit EXF2/TF2 is set. If enabled by the related interrupt enable bit EXF2EN/TF2EN in register T2CON1, an interrupt for the interrupt event EXF2/TF2 will be generated.

Note: When the timer/counter is stopped and while the module remains enabled, it is possible for an external event at T2EX to generate an interrupt. For this to occur, bit EXEN2 in SFR T2CON must be set. In this case, a dummy reload or capture happens depending on the CP_RL2 bit selection. The resulting interrupt could therefore be used in the product as an external falling/rising edge triggered interrupt.

17.4 Timer 2 Register Definition

All Timer 2 and Timer 21 register names described in the following sections will be referenced in other chapters with the module name prefix “T2_” and “T21_”, respectively.

Table 288 Register Address Space

Module	Base Address	End Address	Note
T2	48004000_H	$48004FFF_H$	Timer2
T21	48005000_H	$48005FFF_H$	Timer21

Table 289 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Timer 2 Register Definition, Mode Register			
T2_MOD	Timer 2 Mode Register	04_H	see **Table 290**
Timer 2 Register Definition, Control Register			
T2_CON	Timer 2 Control Register	00_H	see **Table 291**
T2_CON1	Timer 2 Control Register 1	$1C_H$	see **Table 293**
T2_ICLR	Timer 2 Interrupt Clear Register	18_H	see **Table 292**
Timer 2 Register Definition, Timer 2 Reload/Capture Register			
T2_RC	Timer 2 Reload/Capture Register	08_H	see **Table 294**
Timer 2 Register Definition, Timer 2 Count Register			
T2_CNT	Timer 2 Count Register	10_H	see **Table 295**

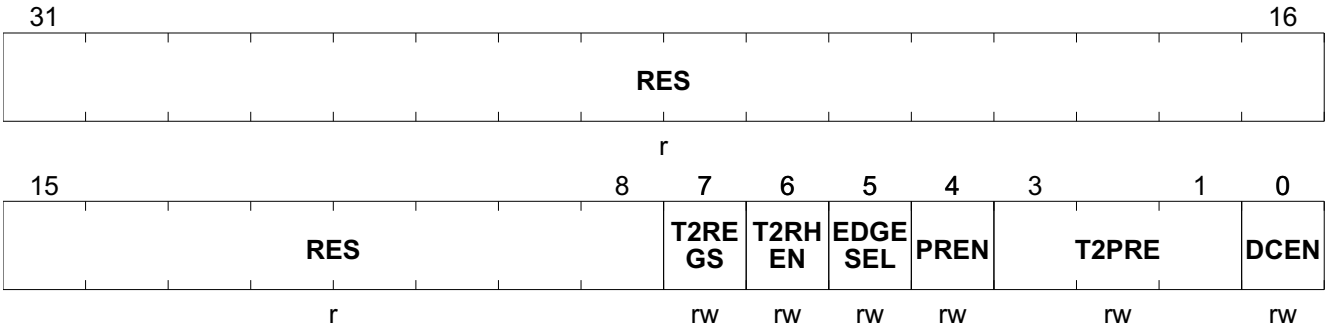
The registers are addressed wordwise.

17.4.1 Mode Register

The T2MOD is used to configure Timer 2 for various modes of operation.

Timer 2 Mode Register

T2_MOD	Offset	Reset Value
Timer 2 Mode Register	04_H	see Table 290



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
T2REGS	7	rw	**Edge Select for Timer 2 External Start** 0_B **FALLING**, The falling edge at Pin T2EX is selected. 1_B **RISING**, The rising edge at Pin T2EX is selected.
T2RHEN	6	rw	**Timer 2 External Start Enable** 0_B **DISABLED**, Timer 2 External Start is disabled. 1_B **ENABLED**, Timer 2 External Start is enabled.
EDGESEL	5	rw	**Edge Select in Capture Mode/Reload Mode** 0_B **FALLING**, The falling edge at Pin T2EX is selected. 1_B **RISING**, The rising edge at Pin T2EX is selected.
PREN	4	rw	**Prescaler Enable** 0_B **DISABLED**, Prescaler is disabled and the 2 or 12 divider takes effect. 1_B **ENABLED**, Prescaler is enabled (see T2PRE bit) and the 2 or 12 divider is bypassed.
T2PRE	3:1	rw	**Timer 2 Prescaler Bit** Selects the input clock for Timer 2 which is derived from the peripheral clock. 000_B **DIV1**, $f_{T2} = f_{sys}$ 001_B **DIV2**, $f_{T2} = f_{sys} / 2$ 010_B **DIV4**, $Df_{T2} = f_{sys} / 4$ 011_B **DIV8**, $f_{T2} = f_{sys} / 8$ 100_B **DIV16**, $f_{T2} = f_{sys} / 16$ 101_B **DIV32**, $f_{T2} = f_{sys} / 32$ 110_B **DIV64**, $f_{T2} = f_{sys} / 64$ 111_B **DIV128**, $f_{T2} = f_{sys} / 128$

Field	Bits	Type	Description
DCEN	0	rw	**Up/Down Counter Enable** 0_B **DISABLED**, Up/Down Counter function is disabled 1_B **ENABLED**, Up/Down Counter function is enabled and controlled by pin T2EX (Up = 1, Down = 0)

Table 290 Reset of T2_MOD

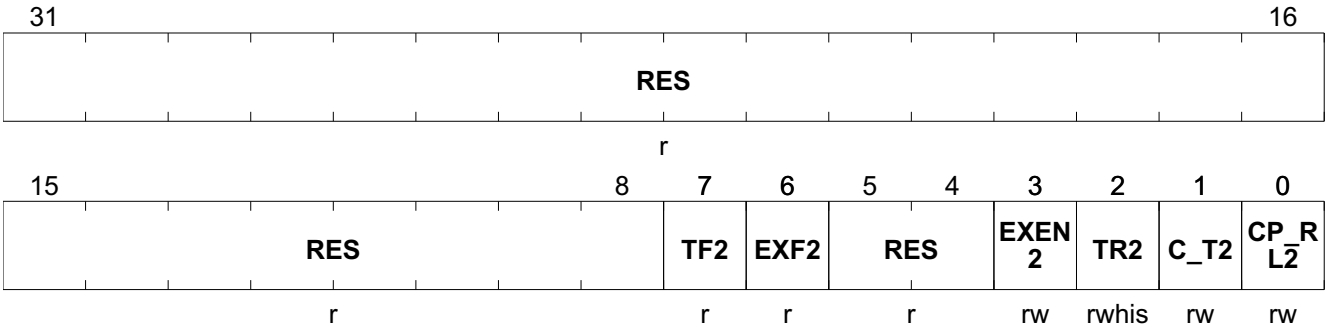
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

17.4.2 Control Register

Control register is used to control the operating modes and interrupt of Timer 2.

Timer 2 Control Register

T2_CON	Offset	Reset Value
Timer 2 Control Register	00_H	see Table 291



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
TF2	7	r	**Timer 2 Overflow/Underflow Flag** Set by a Timer 2 overflow/underflow. Must be cleared by software.
EXF2	6	r	**Timer 2 External Flag** In Capture/Reload Mode, this bit is set by hardware when a negative/positive transition occurs at pin T2EX, if bit EXEN2 = 1. This bit must be cleared by software. *Note: When bit DCEN = 1 in auto-reload mode, no interrupt request to the core is generated.*
RES	5:4	r	**Reserved** Returns 0 if read; should be written with 0.
EXEN2	3	rw	**Timer 2 External Enable Control** 0_B **DISABLED**, External events are disabled. 1_B **ENABLED**, External events are enabled in Capture/Reload Mode.
TR2	2	rwhis	**Timer 2 Start/Stop Control** 0_B **STOP**, Timer 2. 1_B **START**, Timer 2.
C_T2	1	rw	**Timer or Counter Select** 0_B **Timer**, function selected. 1_B **Count**, upon negative edge at pin T2.

Timer2 and Timer21

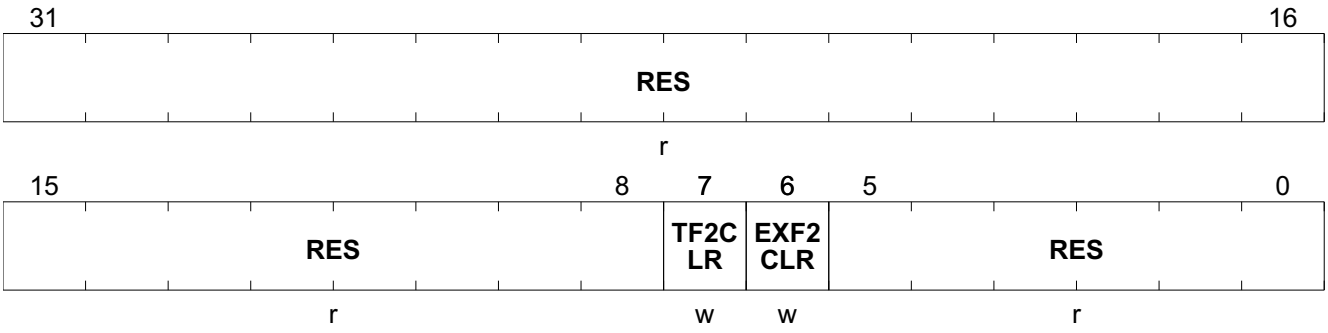
Field	Bits	Type	Description
CP_RL2	0	rw	**Capture/Reload Select** 0_B **Reload**, upon overflow or upon negative/positive transition at pin T2EX (when EXEN2 = 1). 1_B **Capture**, Timer 2 data register contents on the negative/positive transition at pin T2EX, provided EXEN2 = 1.The negative or positive transition at Pin T2EX is selected by bit EDGESEL.

Table 291 Reset of T2_CON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Timer 2 Interrupt Clear Register

T2_ICLR	Offset	Reset Value
Timer 2 Interrupt Clear Register	18_H	see Table 292



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Always read as 0
TF2CLR	7	w	**Overflow/Underflow Interrupt Clear Flag** 0_B **N/A**, Overflow/underflow interrupt is not cleared. 1_B **Clear**, Overflow/underflow interrupt
EXF2CLR	6	w	**External Interrupt Clear Flag** 0_B **N/A**, External interrupt is not cleared. 1_B **Clear**, External interrupt
RES	5:0	r	**Reserved** Always read as 0

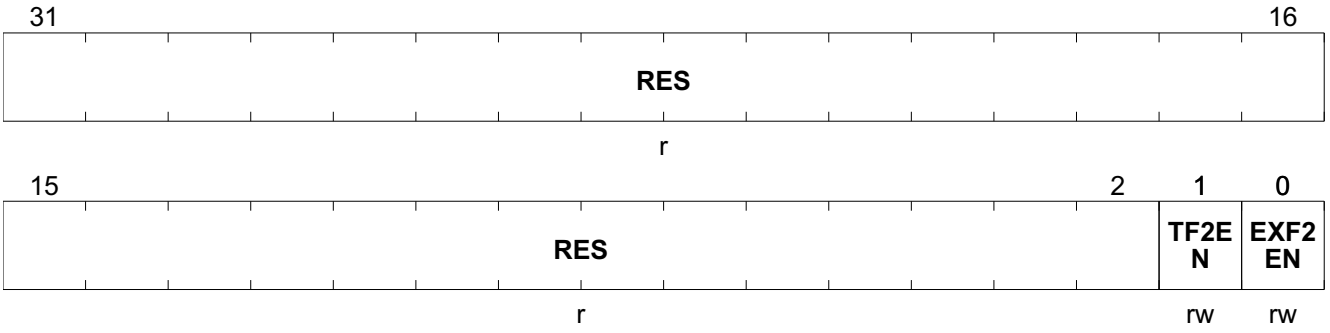
Table 292 Reset of T2_ICLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Timer2 and Timer21

Timer 2 Control Register 1

T2_CON1	Offset	Reset Value
Timer 2 Control Register 1	$1C_H$	see Table 293



Field	Bits	Type	Description
RES	31:2	r	**Reserved** Always read as 0
TF2EN	1	rw	**Overflow/Underflow Interrupt Enable** 0_B **DISABLE**, Overflow/underflow interrupt. 1_B **ENABLE**, Overflow/underflow interrupt.
EXF2EN	0	rw	**External Interrupt Enable** 0_B **DISABLE**, External interrupt. 1_B **ENABLE**, External interrupt

Table 293 Reset of T2_CON1

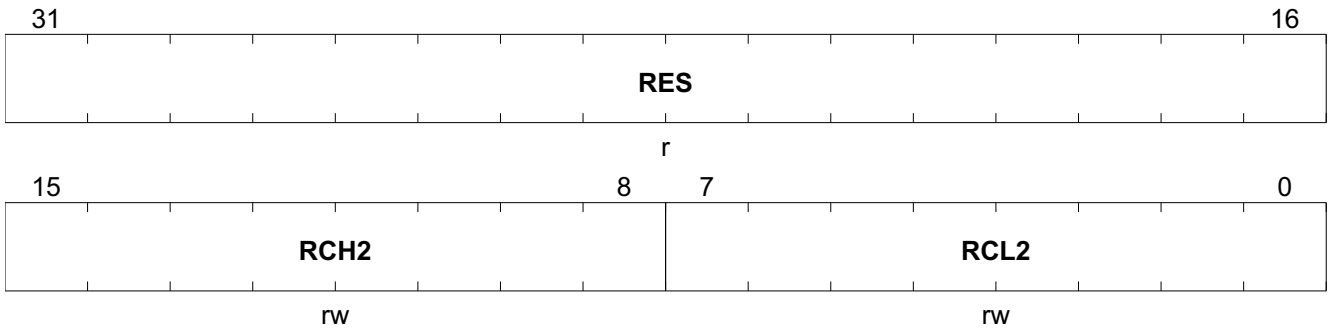
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

17.4.3 Timer 2 Reload/Capture Register

The RC2 register is used for a 16-bit reload of the timer count upon an overflow or a capture of the current timer count depending on the mode selected.

Timer 2 Reload/Capture Register, Low Byte

T2_RC	Offset	Reset Value
Timer 2 Reload/Capture Register	08_H	see Table 294



Field	Bits	Type	Description
RES	31:16	r	**Reserved** Always read as 0
RCH2	15:8	rw	**Reload/Capture Value** Note: Reload/Capture Value can be set by software (highest priority) and is updated by hardware during capture mode. These contents are loaded into the timer register upon an overflow condition, if CP_RL2 = 0.If CP_RL2 = 1, this register is loaded with the current timer count upon a negative/positive transition at pin T2EX when EXEN2 = 1.
RCL2	7:0	rw	**Reload/Capture Value** Note: Reload/Capture Value can be set by software (highest priority) and is updated by hardware during capture mode. These contents are loaded into the timer register upon an overflow condition, if CP_RL2 = 0.If CP_RL2 = 1, this register is loaded with the current timer count upon a negative/positive transition at pin T2EX when EXEN2 = 1.

Table 294 Reset of T2_RC

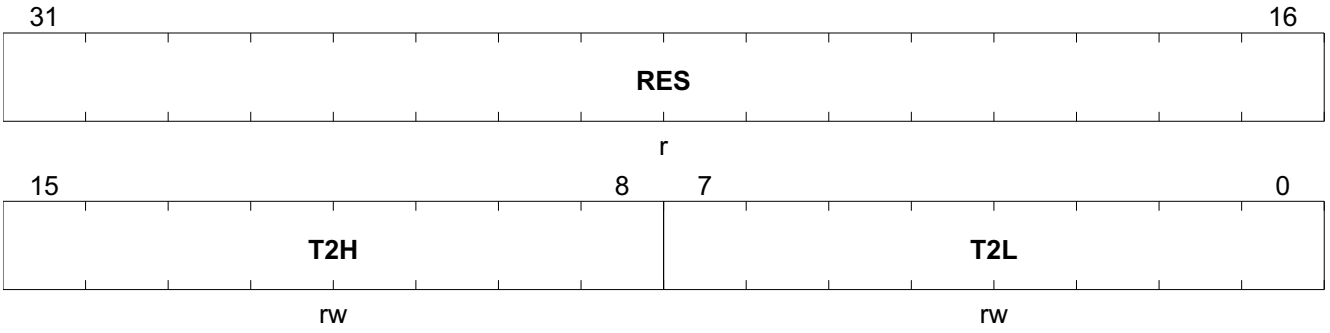
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

17.4.4 Timer 2 Count Register

The T2_CNT register holds the current 16-bit value of the Timer 2 count.

Timer 2 Register

T2_CNT	Offset	Reset Value
Timer 2 Count Register	10_H	see Table 295



Field	Bits	Type	Description
RES	31:16	r	**Reserved** Always read as 0
T2H	15:8	rw	**Timer 2 Value** These bits indicate the current timer value T2[15:8]. Note: Timer 2 can be updated by software (highest priority) and is updated by hardware if T2R is set
T2L	7:0	rw	**Timer 2 Value** These bits indicate the current timer value T2[7:0]. Note: Timer 2 can be updated by software (highest priority) and is updated by hardware if T2R is set

Table 295 Reset of T2_CNT

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

17.5 Timer2 and Timer21 Implementation Details

This section describes:

- the TLE985xQX module related interfaces such as port connections and interrupt control
- all TLE985xQX module related registers with their addresses

17.5.1 Interfaces of the Timer2 and Timer21

Overviews of the Timer2 and Timer21 kernel I/O interfaces and interrupt signals are shown in **Figure 123** and **Figure 124**.

Timer2 and Timer21 can be suspended when Debug Mode enters Monitor Mode and has the Debug Suspend signal activated, provided the timer suspend bits, T2SUSP and T21SUSP (in SCU SFR MODSUSP) are set. Refer to SCU chapter.

The interrupt request of the Timer2 and Timer21 is not connected directly to the CPU's Interrupt Controller, but via the System Control Unit (SCU). The General Purpose IO (GPIO) Port provides the interface from the Timer2 and Timer21 to the external world.

The external trigger and counter inputs of the two Timer 2 modules can be selected from several different sources. This selection is performed by the SCU via the corresponding input control and select bits in SFR MODPISEL1 and MODPISEL2.

In the TLE985xQX, Timer2 and Timer21 allow additionally to trigger ADC1 conversions through the t2(1)_adc_trigger signals. These trigger signals are generated while the timer is working in timer mode (C_T2 = 0).

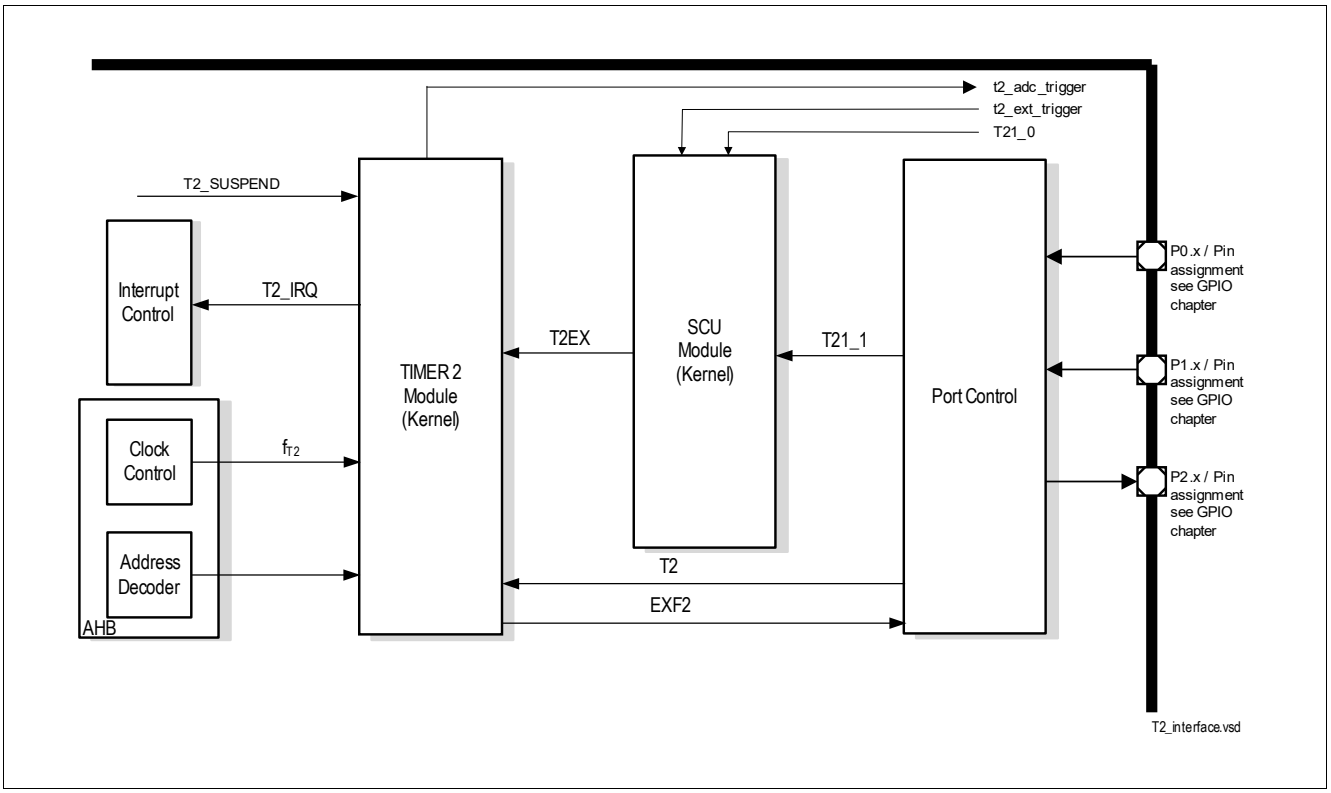


Figure 123 Timer 2 Module I/O Interface

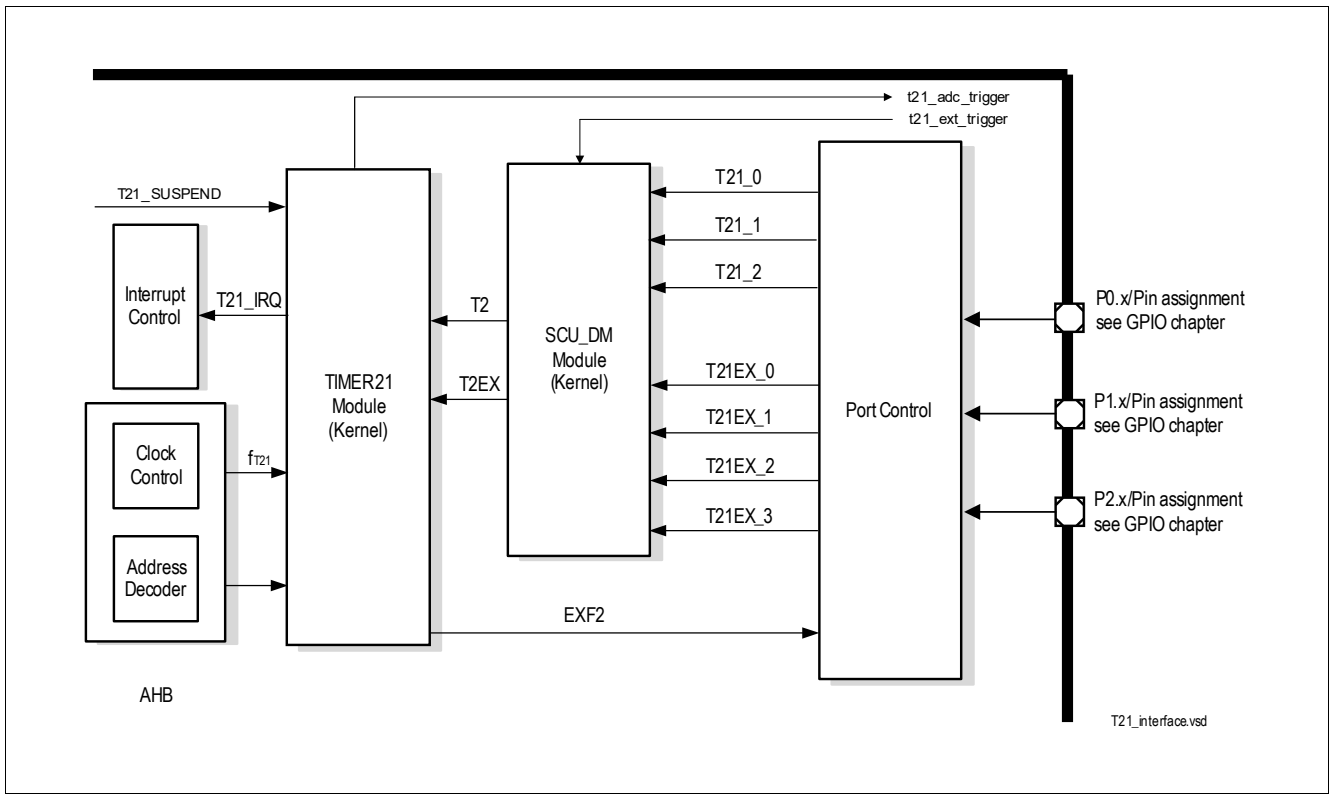


Figure 124 Timer 21 Module I/O Interface

18 Capture/Compare Unit 6 (CCU6)

The CCU6 is a high-resolution 16-bit capture and compare unit with application specific modes, mainly for AC drive control. Special operating modes support the control of Brushless DC-motors using Hall sensors or Back-EMF detection. Furthermore, block commutation and control mechanisms for multi-phase machines are supported.

It also supports inputs to start several timers synchronously, an important feature in devices with several CCU6 modules.

This chapter is structured as follows:

- Functional description of the CCU6 kernel (see **Section 18.2**)
  - Introduction (see **Section 18.2**)
  - Operating T12 (see **Section 18.3**)
  - Operating T13 (see **Section 18.4**)
  - Trap handling (see **Section 18.5**)
  - Multi-Channel mode (see **Section 18.6**)
  - Hall sensor mode (see **Section 18.7**)
  - Interrupt handling (see **Section 18.8**)
  - General module operation (see **Section 18.9**)
- CCU6 kernel registers description (see **Section 18.10**)
- TLE985xQX implementation specific details (see **Section 18.11**)

18.1 Feature Set Overview

This section gives an overview over the different building blocks and their main features.

Timer 12 Block Features

- Three capture/compare channels, each channel can be used either as capture or as compare channel
- Generation of a three-phase PWM supported (six outputs, individual signals for High Side and low-side switches)
- 16-bit resolution, maximum count frequency = peripheral clock
- Dead-time control for each channel to avoid short-circuits in the power stage
- Concurrent update of T12 registers
- Center-aligned and edge-aligned PWM can be generated
- Single-shot mode supported
- Start can be controlled by external events
- Capability of counting external events
- Multiple interrupt request sources
- Hysteresis-like control mode

Timer 13 Block Features

- One independent compare channel with one output
- 16-bit resolution, maximum count frequency = peripheral clock
- Concurrent update of T13 registers

- Can be synchronized to T12
- Interrupt generation at period-match and compare-match
- Single-shot mode supported
- Start can be controlled by external events
- Capability of counting external events

Additional Specific Functions

- Block commutation for Brushless DC-drives implemented
- Position detection via Hall-sensor pattern
- Noise filter supported for position input signals
- Automatic rotational speed measurement and commutation control for block commutation
- Integrated error handling
- Fast emergency stop without CPU load via external signal ($\overline{\text{CTRAP}}$)
- Control modes for multi-channel AC-drives
- Output levels can be selected and adapted to the power stage

18.2 Introduction

The CCU6 unit is made up of a Timer T12 Block with three capture/compare channels and a Timer T13 Block with one compare channel. The T12 channels can independently generate PWM signals or accept capture triggers, or they can jointly generate control signal patterns to drive AC-motors or inverters.

A rich set of status bits, synchronized updating of parameter values via shadow registers, and flexible generation of interrupt request signals provide means for efficient software-control.

Note: The capture/compare module itself is named CCU6 (capture/compare unit 6). A capture/compare channel inside this module is named CC6x.

18.2.1 Block Diagram

The Timer T12 can work in capture and/or compare mode for its three channels. The modes can also be combined (e.g. a channel works in compare mode, whereas another channel works in capture mode). The Timer T13 can work in compare mode only. The multi-channel control unit generates output patterns which can be modulated by T12 and/or T13. The modulation sources can be selected and combined for the signal modulation.

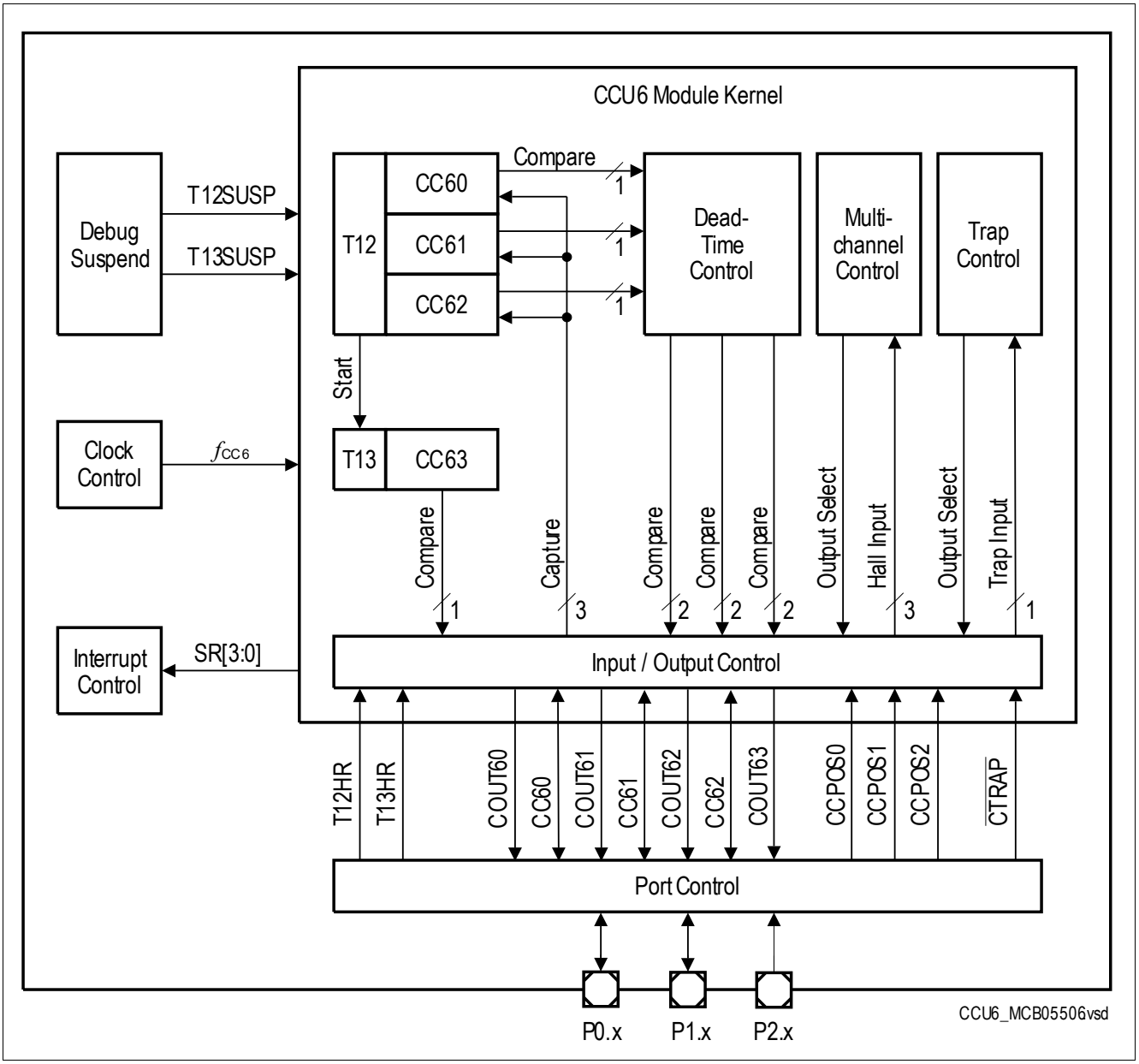


Figure 125 CCU6 Block Diagram

18.3 Operating Timer T12

The timer T12 block is the main unit to generate the 3-phase PWM signals. A 16-bit counter is connected to 3 channel registers via comparators, which generate a signal when the counter contents match one of the channel register contents. A variety of control functions facilitate the adaptation of the T12 structure to different application needs.

Besides the 3-phase PWM generation, the T12 block offers options for individual compare and capture functions, as well as dead-time control and hysteresis-like compare mode.

This section provides information about:

- T12 overview (see **Section 18.3.1**)
- Counting scheme (see **Section 18.3.2**)
- Compare modes (see **Section 18.3.3**)
- Compare mode output path (see **Section 18.3.4**)
- Capture modes (see **Section 18.3.5**)
- Shadow transfer (see **Section 18.3.6**)
- T12 operating mode selection (see **Section 18.3.7**)

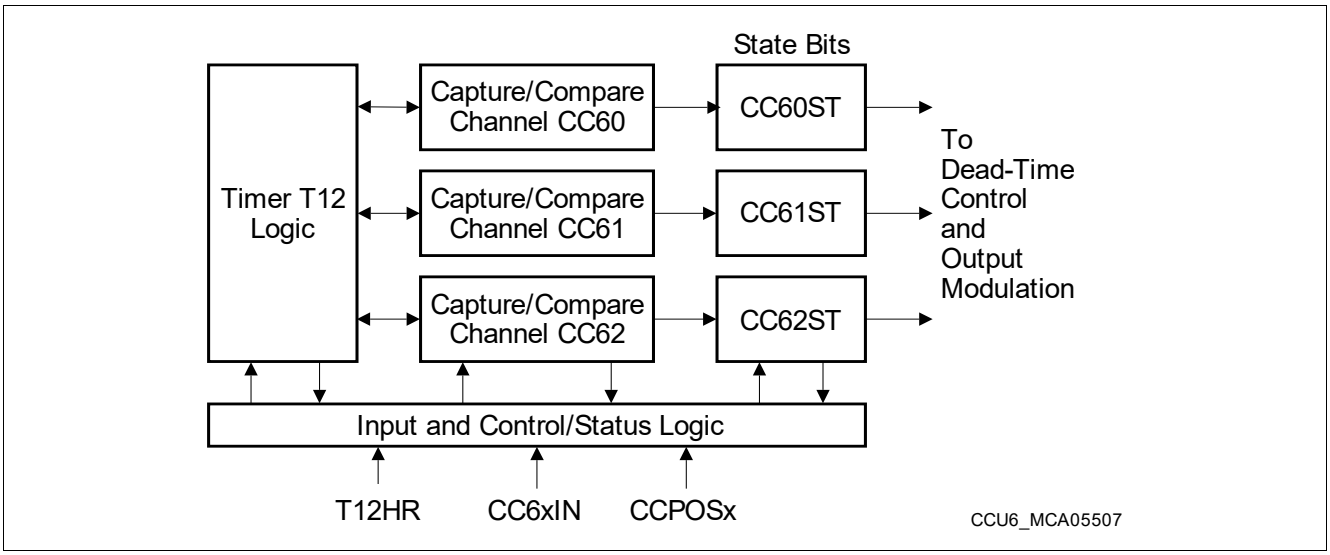


Figure 126 Overview Diagram of the Timer T12 Block

18.3.1 T12 Overview

Figure 127 shows a detailed block diagram of Timer T12. The functions of the timer T12 block are controlled by bits in registers TCTR0, TCTR2, and PISEL0.

Timer T12 receives its input clock (f_{T12}) from the module clock f_{CC6} via a programmable prescaler and an optional 1/256 divider or from an input signal T12HR. These options are controlled via bit fields T12CLK and T12PRE (see **Table 296**). T12 can count up or down, depending on the selected operation mode. A direction flag, CDIR, indicates the current counting direction.

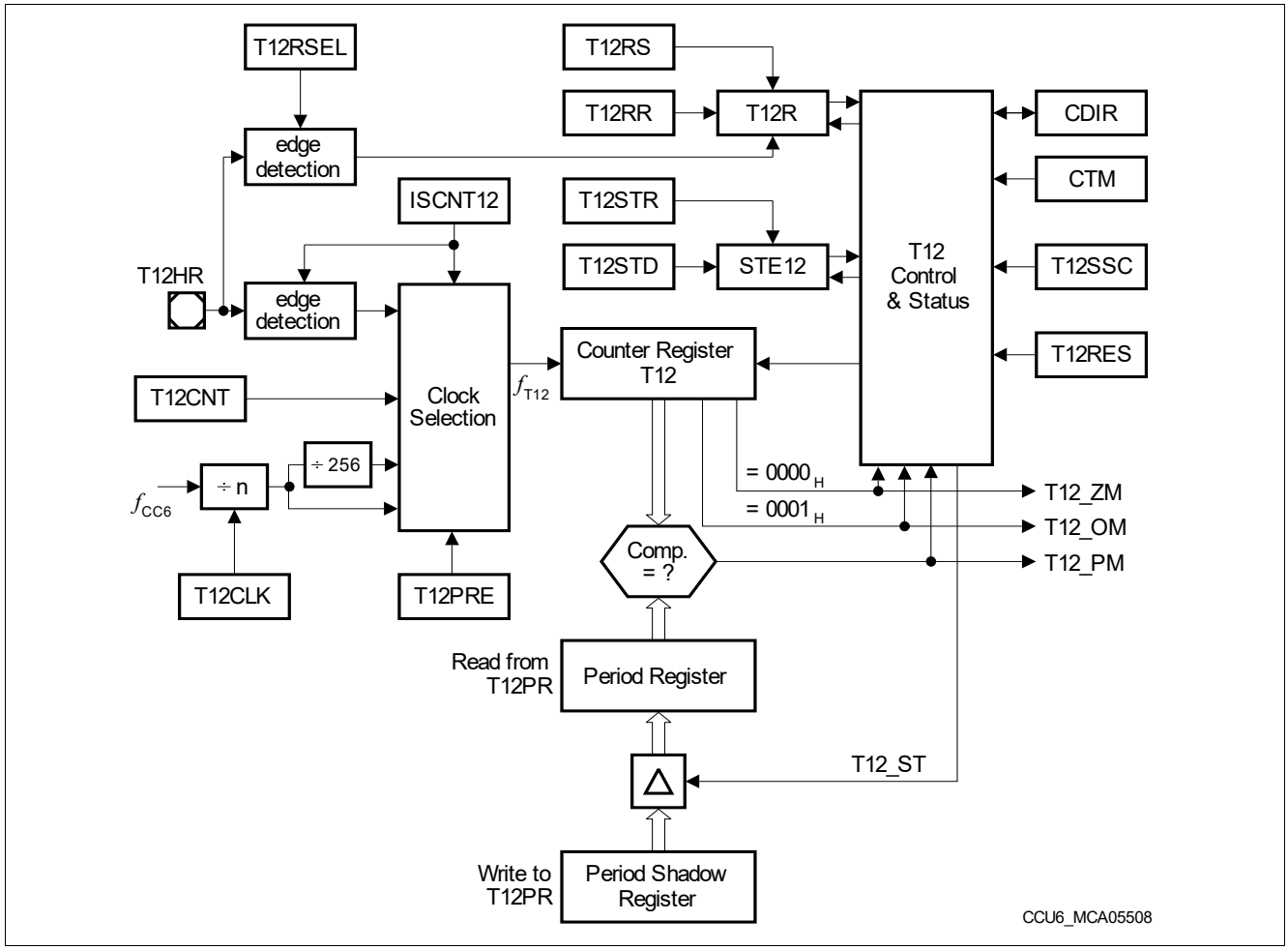


Figure 127 Timer T12 Logic and Period Comparators

Via a comparator, the T12 counter register T12 is connected to a Period Register T12PR. This register determines the maximum count value for T12.

In Edge-Aligned mode, T12 is cleared to 0000_H after it has reached the period value defined by T12PR. In Center-Aligned mode, the count direction of T12 is set from 'up' to 'down' after it has reached the period value (please note that in this mode, T12 exceeds the period value by one before counting down). In both cases, signal T12_PM (T12 Period Match) is generated. The Period Register receives a new period value from its Shadow Period Register.

A read access to T12PR delivers the current period value at the comparator, whereas a write access targets the Shadow Period Register to prepare another period value. The transfer of a new period value from the Shadow Period Register into the Period Register (see **Section 18.3.6**) is controlled via the 'T12 Shadow Transfer' control signal, T12_ST. The generation of this signal depends on the operating mode and on the shadow

transfer enable bit STE12. Providing a shadow register for the period value as well as for other values related to the generation of the PWM signal allows a concurrent update by software for all relevant parameters.

Two further signals indicate whether the counter contents are equal to 0000_H (T12_ZM = zero match) or 0001_H (T12_OM = one match). These signals control the counting and switching behavior of T12.

The basic operating mode of T12, either Edge-Aligned mode (**Figure 128**) or Center-Aligned mode (**Figure 129**), is selected via bit CTM. A Single-Shot control bit, T12SSC, enables an automatic stop of the timer when the current counting period is finished (see **Figure 130** and **Figure 131**).

The start or stop of T12 is controlled by the Run bit T12R that can be modified by bits in register TCTR4. The run bit can be set/cleared by software via the associated set/clear bits T12RS or T12RR, it can be set by a selectable edge of the input signal T12HR (TCTR2.T12RSEL), or it is cleared by hardware according to preselected conditions.

The timer T12 run bit T12R must not be set while the applied T12 period value is zero. Timer T12 can be cleared via control bit T12RES. Setting this write-only bit does only clear the timer contents, but has no further effects, for example, it does not stop the timer.

The generation of the T12 shadow transfer control signal, T12_ST, is enabled via bit STE12. This bit can be set or reset by software indirectly through its associated set/clear control bits T12STR and T12STD.

While Timer T12 is running, write accesses to the count register T12 are not taken into account. If T12 is stopped and the Dead-Time counters are 0, write actions to register T12 are immediately taken into account.

18.3.2 T12 Counting Scheme

This section describes the clocking and counting capabilities of T12.

18.3.2.1 Clock Selection

In **Timer Mode** (PISEL2.ISCNT12 = 00_B), the input clock f_{T12} of Timer T12 is derived from the internal module clock f_{CC6} through a programmable prescaler and an optional 1/256 divider. The resulting prescaler factors are listed in **Table 296**. The prescaler of T12 is cleared while T12 is not running (TCTR0.T12R = 0) to ensure reproducible timings and delays.

Table 296 Timer T12 Input Frequency Options

T12CLK	Resulting Input Clock f_{T12} Prescaler Off (T12PRE = 0)	Resulting Input Clock f_{T12} Prescaler On (T12PRE = 1)
000_B	f_{CC6}	f_{CC6} / 256
001_B	f_{CC6} / 2	f_{CC6} / 512
010_B	f_{CC6} / 4	f_{CC6} / 1024
011_B	f_{CC6} / 8	f_{CC6} / 2048
100_B	f_{CC6} / 16	f_{CC6} / 4096
101_B	f_{CC6} / 32	f_{CC6} / 8192
110_B	f_{CC6} / 64	f_{CC6} / 16384
111_B	f_{CC6} / 128	f_{CC6} / 32768

In **Counter Mode**, timer T12 counts one step:

- If a 1 is written to TCTR4.T12CNT and PISEL2.ISCNT12 = 01_B
- If a rising edge of input signal T12HR is detected and PISEL2.ISCNT12 = 10_B
- If a falling edge of input signal T12HR is detected and PISEL2.ISCNT12 = 11_B

18.3.2.2 Edge-Aligned / Center-Aligned Mode

In **Edge-Aligned Mode** (CTM = 0), timer T12 is always counting upwards (CDIR = 0). When reaching the value given by the period register (period-match T12_PM), the value of T12 is cleared with the next counting step (saw tooth shape).

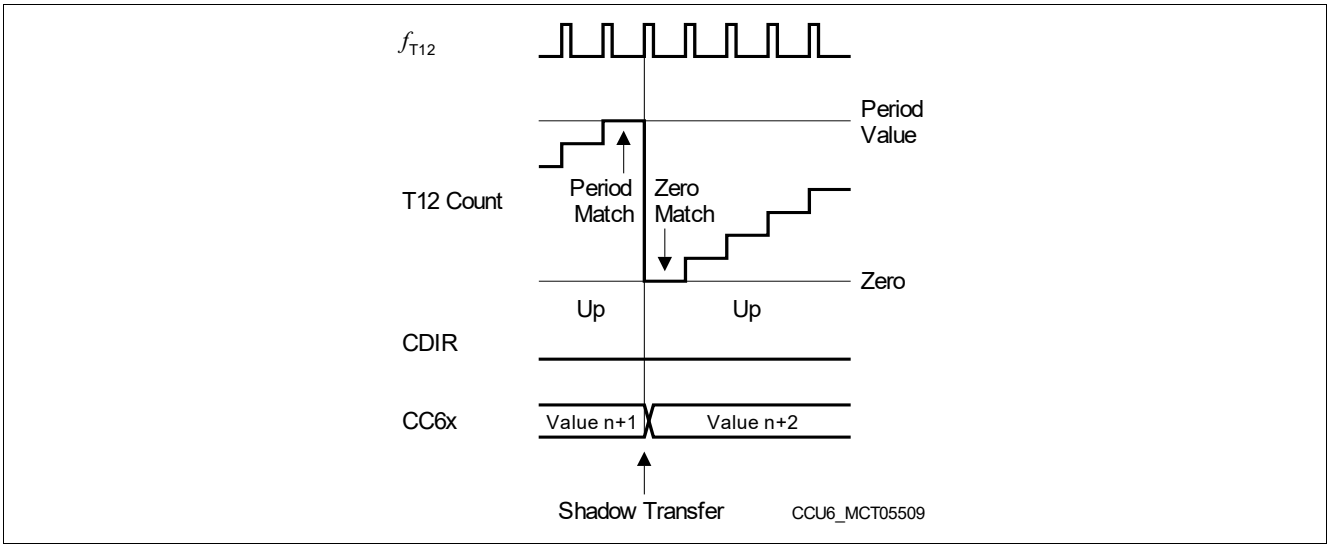


Figure 128 T12 Operation in Edge-Aligned Mode

As a result, in Edge-Aligned mode, the timer period is given by:

$T12_{PER}$ = <Period-Value> + 1; in T12 clocks (f_{T12}) (18.1)

In **Center-Aligned Mode** (CTM = 1), timer T12 is counting upwards or downwards (triangular shape). When reaching the value given by the period register (period-match T12_PM) while counting upwards (CDIR = 0), the counting direction control bit CDIR is changed to downwards (CDIR = 1) with the next counting step.

When reaching the value 0001_H (one-match T12_OM) while counting downwards, the counting direction control bit CDIR is changed to upwards with the next counting step.

As a result, in Center.Aligned mode, the timer period is given by:

$T12_{PER}$ = (<Period-Value> + 1) x2; in T12 clocks (f_{T12}) (18.2)

- With the next clock event of f_{T12} the count direction is set to counting up (CDIR = 0) when the counter reaches 0001_H while counting down.
- With the next clock event of f_{T12} the count direction is set to counting down (CDIR = 1) when the Period-Match is detected while counting up.
- With the next clock event of f_{T12} the counter counts up while CDIR = 0 and it counts down while CDIR = 1.

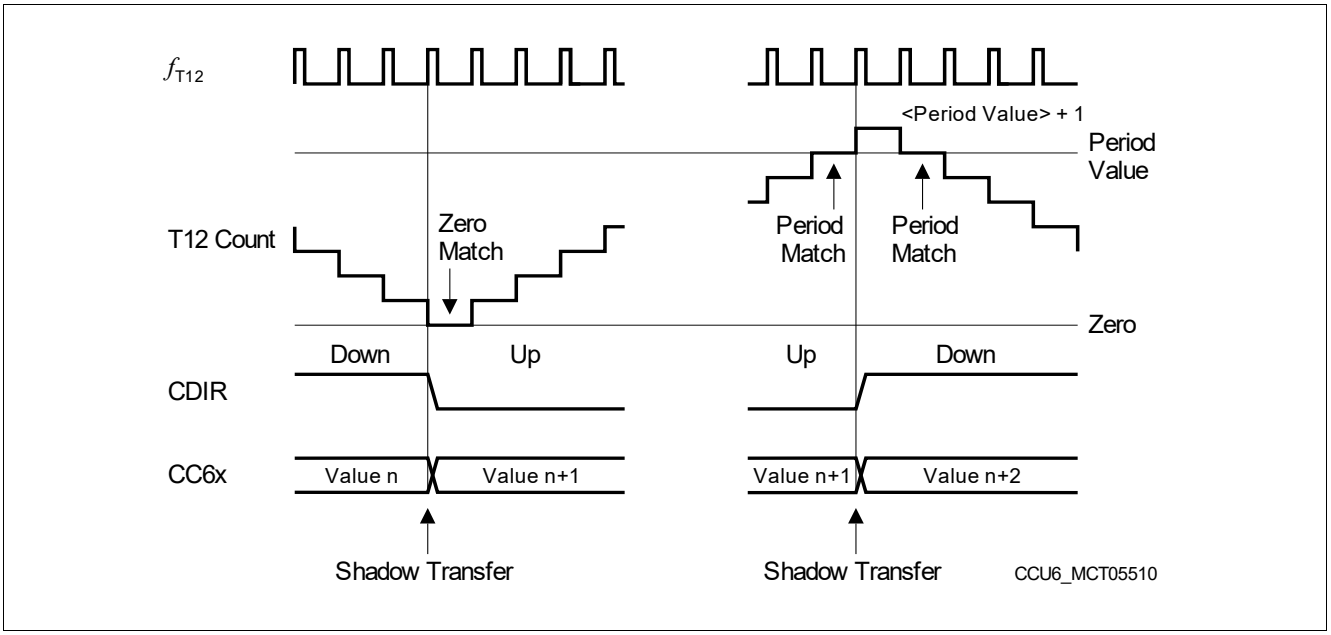


Figure 129 T12 Operation in Center-Aligned Mode

*Note: Bit CDIR changes with the next timer clock event after the one-match or the period-match. Therefore, the timer continues counting in the previous direction for one cycle before actually changing its direction (see **Figure 129**).*

18.3.2.3 Single-Shot Mode

In Single-Shot Mode, the timer run bit T12R is cleared by hardware. If bit T12SSC = 1, the timer T12 will stop when the current timer period is finished.

In Edge-Aligned mode, T12R is cleared when the timer becomes zero after having reached the period value (see **Figure 130**).

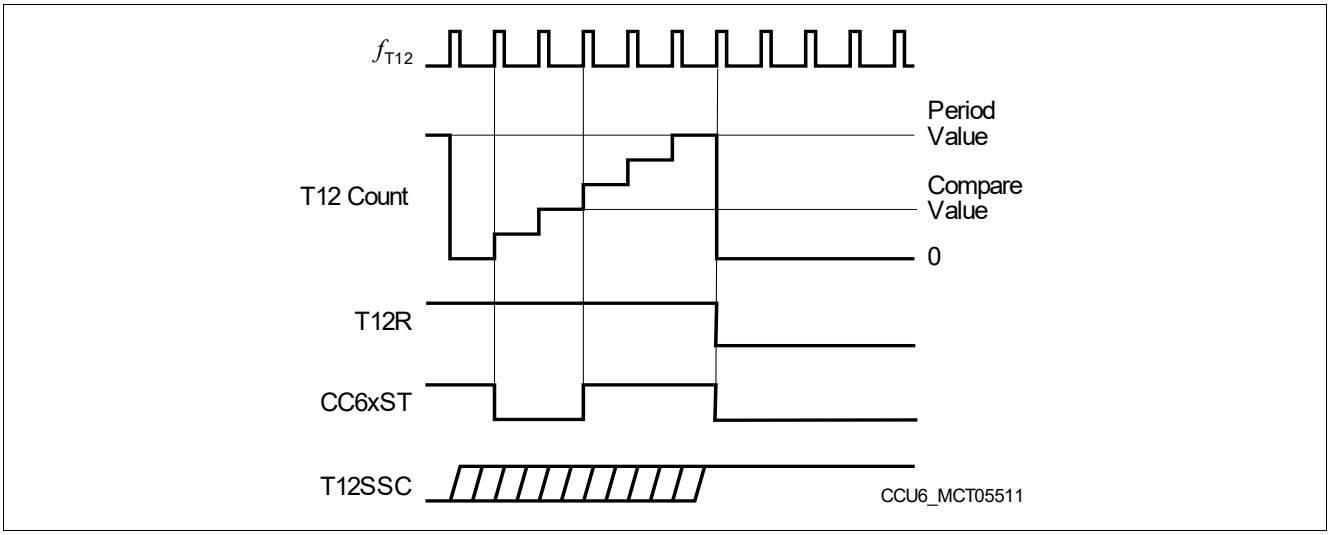


Figure 130 Single-Shot Operation in Edge-Aligned Mode

In Center-Aligned mode, the period is finished when the timer has counted down to zero (one clock cycle after the one-match while counting down, see **Figure 131**).

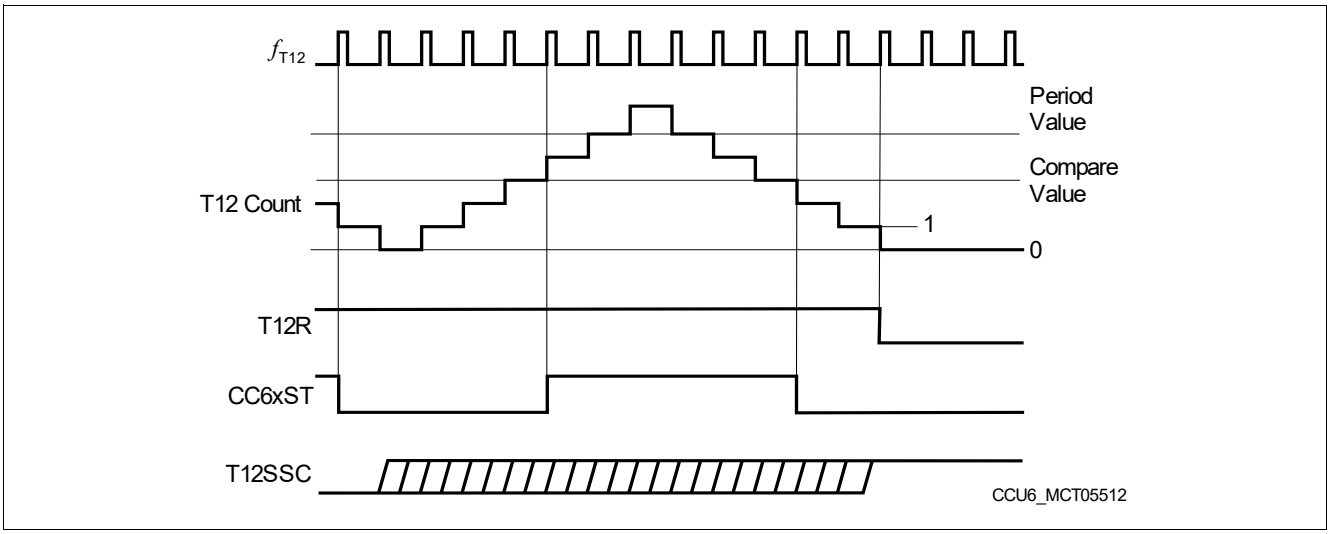


Figure 131 Single-Shot Operation in Center-Aligned Mode

18.3.3 T12 Compare Mode

Associated with Timer T12 are three individual capture/compare channels, that can perform compare or capture operations with regard to the contents of the T12 counter. The capture functions are explained in **Section 18.3.5**.

18.3.3.1 Compare Channels

In Compare Mode (see **Figure 132**), the three individual compare channels CC60 CC61, and CC62 can generate a three-phase PWM pattern.

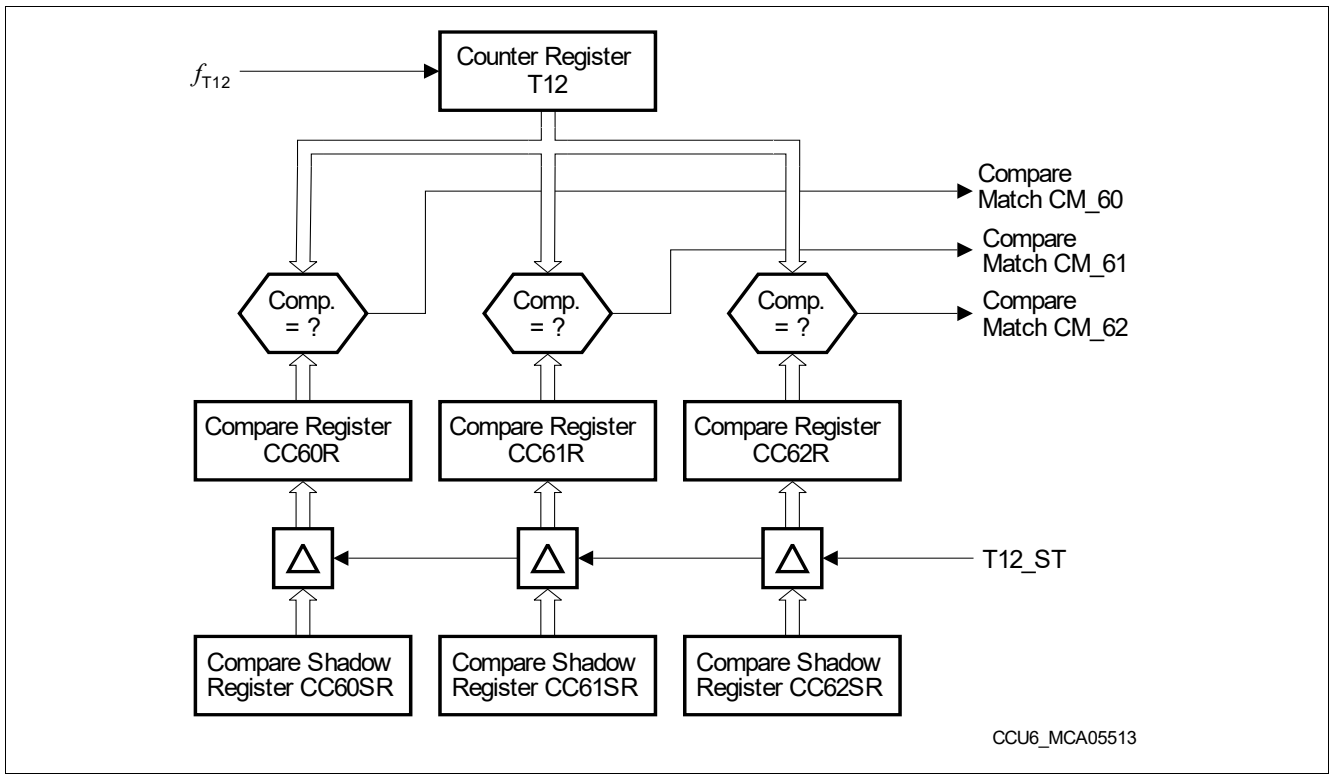


Figure 132 T12 Channel Comparators

Each compare channel is connected to the T12 counter register via its individual equal-to comparator, generating a match signal when the contents of the counter matches the contents of the associated compare register. Each channel consists of the comparator and a double register structure - the actual compare register CC6xR, feeding the comparator, and an associated shadow register CC6xSR, that is preloaded by software and transferred into the compare register when signal T12 shadow transfer, T12_ST, gets active. Providing a shadow register for the compare value as well as for other values related to the generation of the PWM signal facilitates a concurrent update by software for all relevant parameters of a three-phase PWM.

18.3.3.2 Channel State Bits

Associated with each (compare) channel is a State Bit, CMPSTAT.CC6xST, holding the status of the compare (or capture) operation (see **Figure 133**). In compare mode, the State Bits are modified according to a set of switching rules, depending on the current status of timer T12.

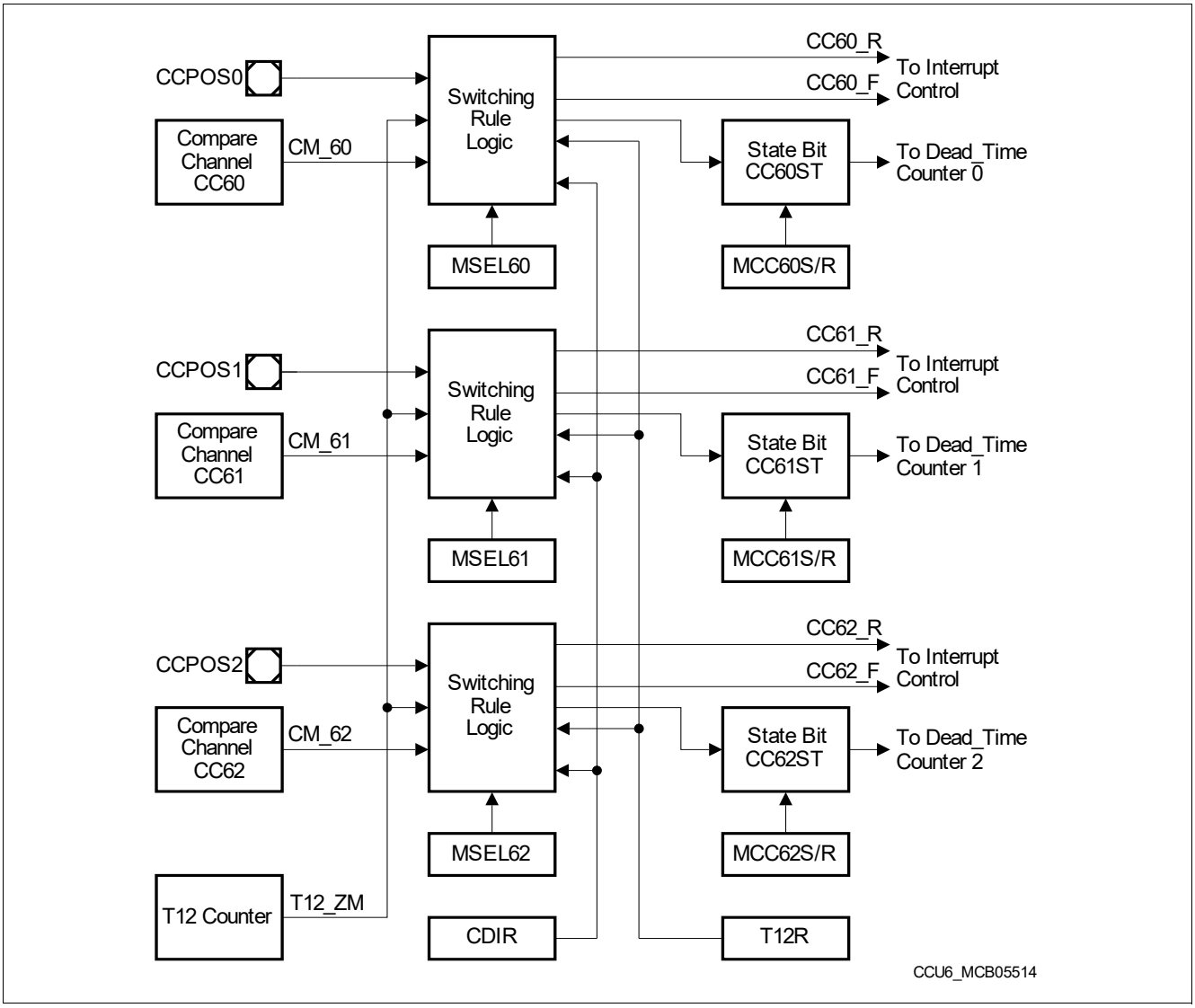


Figure 133 Compare State Bits for Compare Mode

The inputs to the switching rule logic for the CC6xST bits are the timer direction (CDIR), the timer run bit (T12R), the timer T12 zero-match signal (T12_ZM), and the actual individual compare-match signals CM_6x as well as the mode control bits, T12MSEL.MSEL6x.

In addition, each state bit can be set or cleared by software via the appropriate set and reset bits in register CMPMODIF, MCC6xS and MCC6xR. The input signals

x are used in hysteresis-like compare mode, whereas in normal compare mode, these inputs are ignored.

Note: In Hall Sensor, single shot or capture modes, additional/different rules are taken into account (see related sections).

A compare interrupt event CC6x_R is signaled when a compare match is detected while counting upwards, whereas the compare interrupt event CC6x_F is signaled when a compare match is detected while counting down. The actual setting of a State Bit has no influence on the interrupt generation in compare mode.

A modification of a State Bit CC6xST by the switching rule logic due to a compare action is only possible while Timer T12 is running (T12R = 1). If this is the case, the following switching rules apply for setting and clearing the State Bits in Compare Mode (illustrated in **Figure 134** and **Figure 135**):

A State Bit **CC6xST is set** to 1:

- with the next T12 clock (f_{T12}) after a compare-match when T12 is counting up (i.e., when the counter is incremented above the compare value);
- with the next T12 clock (f_{T12}) after a zero-match AND a parallel compare-match when T12 is counting up.

A State Bit **CC6xST is cleared** to 0:

- with the next T12 clock (f_{T12}) after a compare-match when T12 is counting down (i.e., when the counter is decremented below the compare value in center-aligned mode);
- with the next T12 clock (f_{T12}) after a zero-match AND NO parallel compare-match when T12 is counting up.

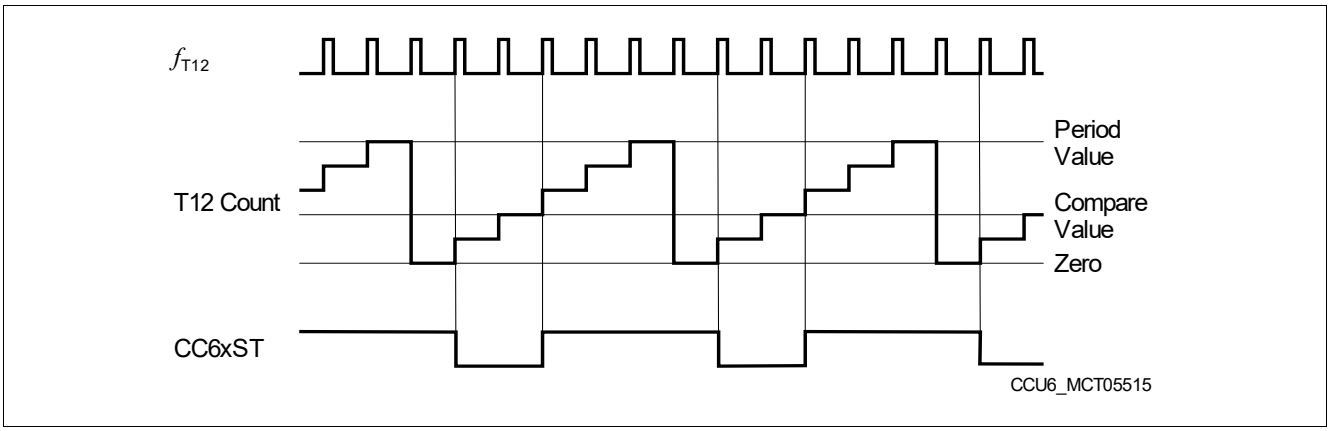


Figure 134 Compare Operation, Edge-Aligned Mode

Figure 136 illustrates some more examples for compare waveforms. It is important to note that in these examples, it is assumed that some of the compare values are changed while the timer is running. This change is performed via a software preload of the Shadow Register, CC6xSR. The value is transferred to the actual Compare Register CC6xR with the T12 Shadow Transfer signal, T12_ST, that is assumed to be enabled.

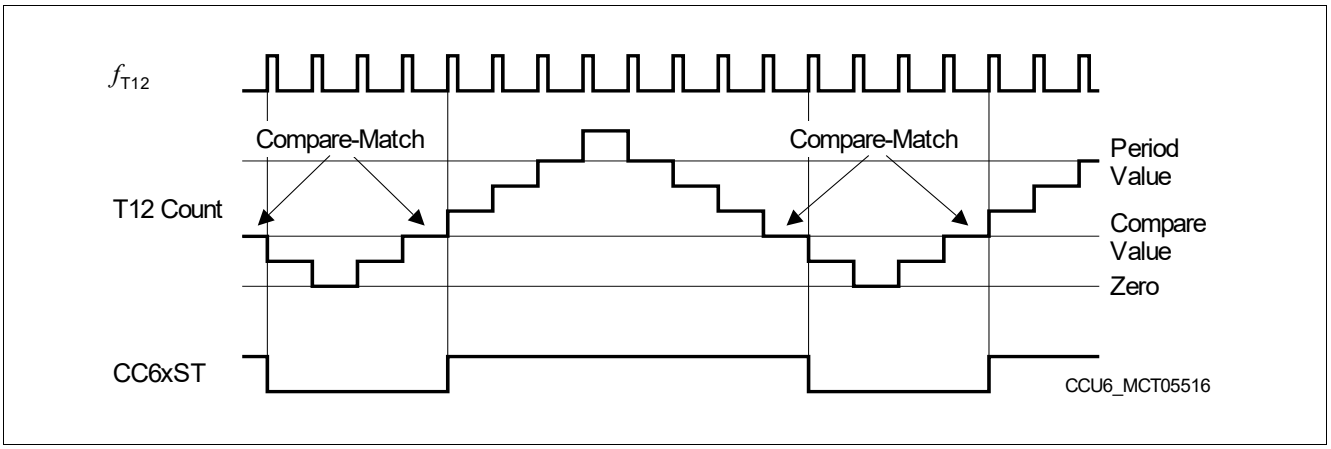


Figure 135 Compare Operation, Center-Aligned Mode

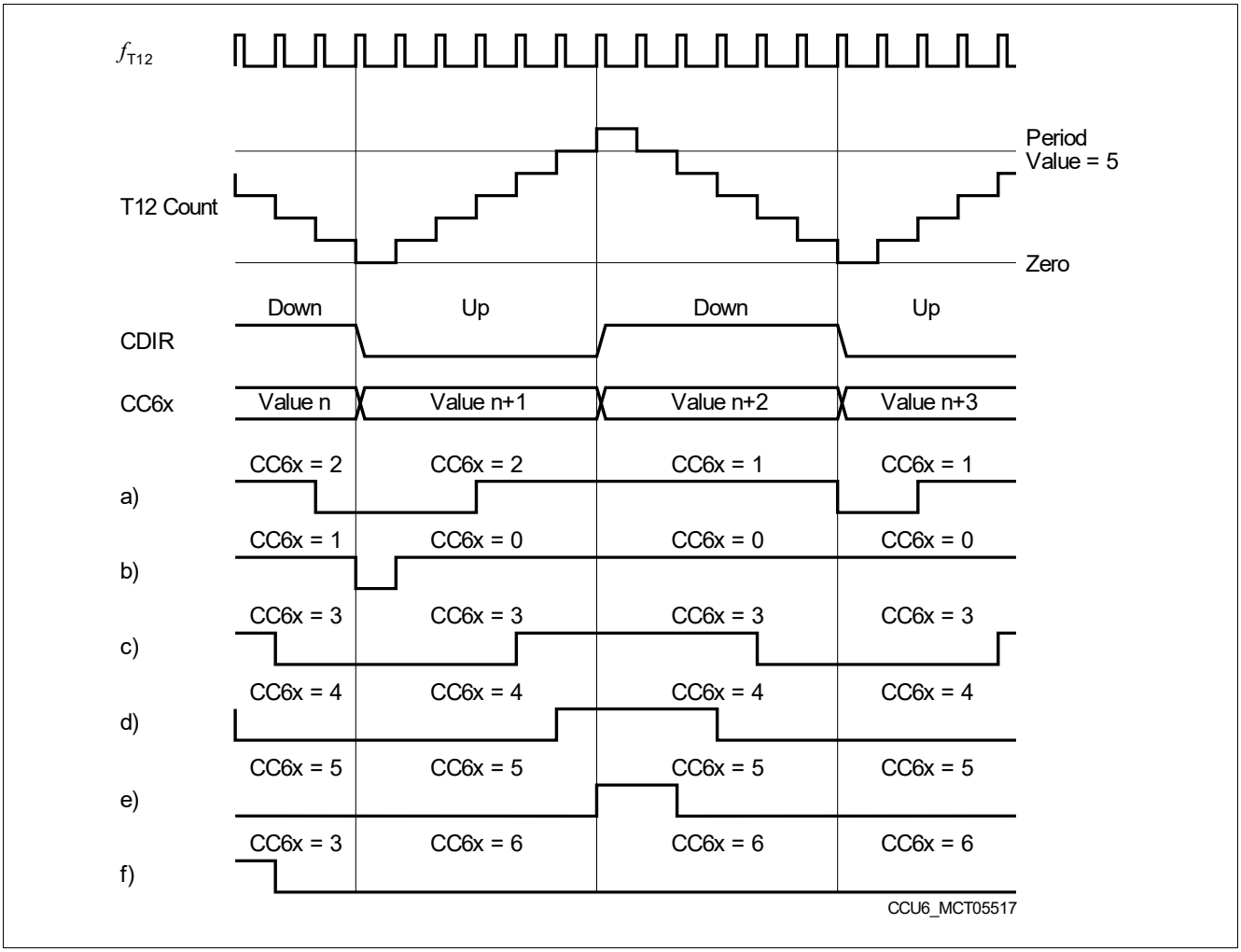


Figure 136 Compare Waveform Examples

Example b) illustrates the transition to a duty cycle of 100%. First, a compare value of 0001_H is used, then changed to 0000_H. Please note that a low pulse with the length of one T12 clock is still produced in the cycle where the new value 0000_H is in effect; this pulse originates from the previous value 0001_H. In the following timer cycles, the State Bit CC6xST remains at 1, producing a 100% duty cycle signal. In this case, the compare rule ‘zero-match AND compare-match’ is in effect.

Example f) shows the transition to a duty cycle of 0%. The new compare value is set to <Period-Value> + 1, and the State Bit CC6ST remains cleared.

Figure 137 illustrates an example for the waveforms of all three channels. With the appropriate dead-time control and output modulation, a very efficient 3-phase PWM signal can be generated.

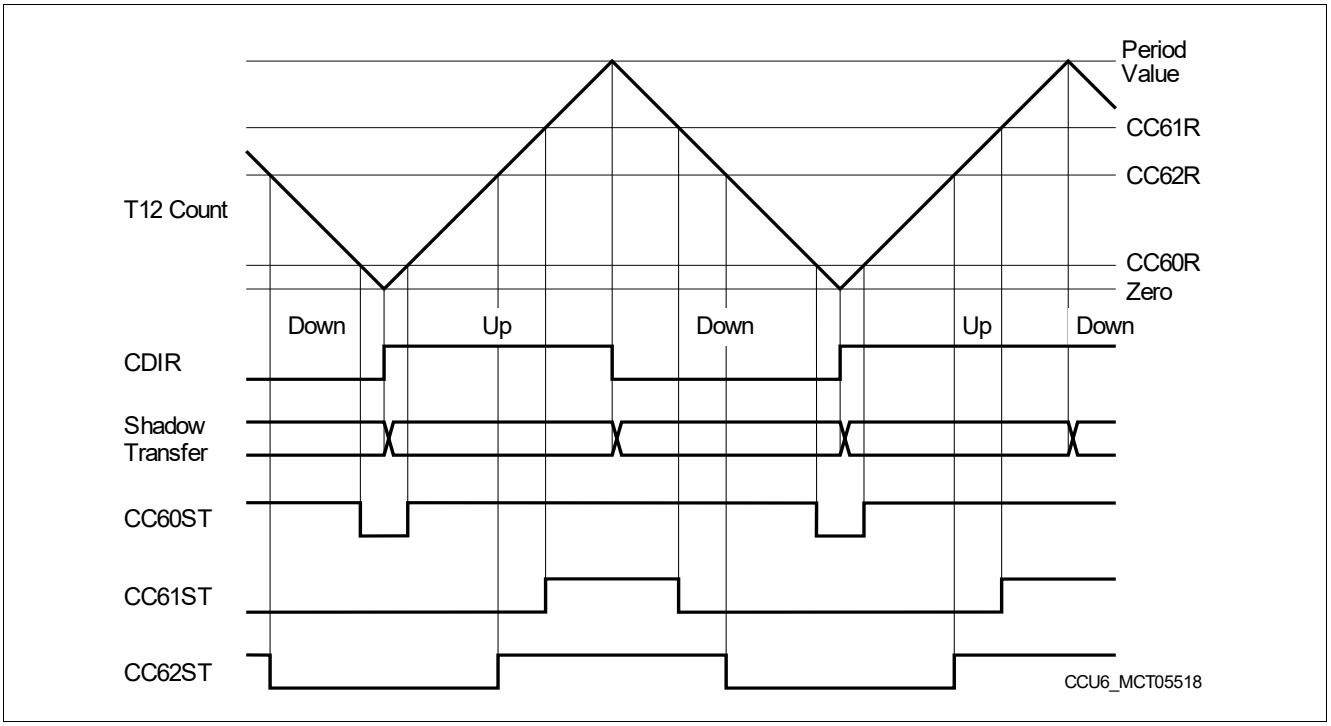


Figure 137 Three-Channel Compare Waveforms

18.3.3.3 Hysteresis-Like Control Mode

The hysteresis-like control mode (T12MSEL.MSEL6x = 1001_B) offers the possibility to switch off the PWM output if the input CCPOSx becomes 0 by clearing the State Bit CC6xST. This can be used as a simple motor control feature by using a comparator indicating, e.g., overcurrent. While CCPOSx = 0, the PWM outputs of the corresponding channel are driving their passive levels, because the setting of bit CC6xST is only possible while CCPOSx = 1.

As long as input CCPOSx is 0, the corresponding State Bit is held 0. When CCPOSx is at high level, the outputs can be in active state and are determined by bit CC6xST (see **Figure 133** for the state bit logic and **Figure 138** for the output paths). The CCPOSx inputs are evaluated with f_{CC6}.

This mode can be used to introduce a timing-related behavior to a hysteresis controller. A standard hysteresis controller detects if a value exceeds a limit and switches its output according to the compare result. Depending on the operating conditions, the switching frequency and the duty cycle are not fixed, but change permanently.

If (outer) time-related control loops based on a hysteresis controller in an inner loop should be implemented, the outer loops show a better behavior if they are synchronized to the inner loops. Therefore, the hysteresis-like mode can be used, that combines timer-related switching with a hysteresis controller behavior. For example, in this mode, an output can be switched on according to a fixed time base, but it is switched off as soon as a falling edge is detected at input CCPOSx.

This mode can also be used for standard PWM with overcurrent protection. As long as there is no low level signal at pin CCPOSx, the output signals are generated in the normal manner as described in the previous sections. Only if input CCPOSx shows a low level, e.g. due to the detection of overcurrent, the outputs are shut off to avoid harmful stress to the system.

18.3.4 Compare Mode Output Path

Figure 138 gives an overview on the signal path from a channel State Bit to its output pin in its simplest form. As illustrated, a user has a variety of controls to determine the desired output signal switching behavior in relation to the current state of the State Bit, CC6xST. Please refer to **Section 18.3.4.3** for details on the output modulation.

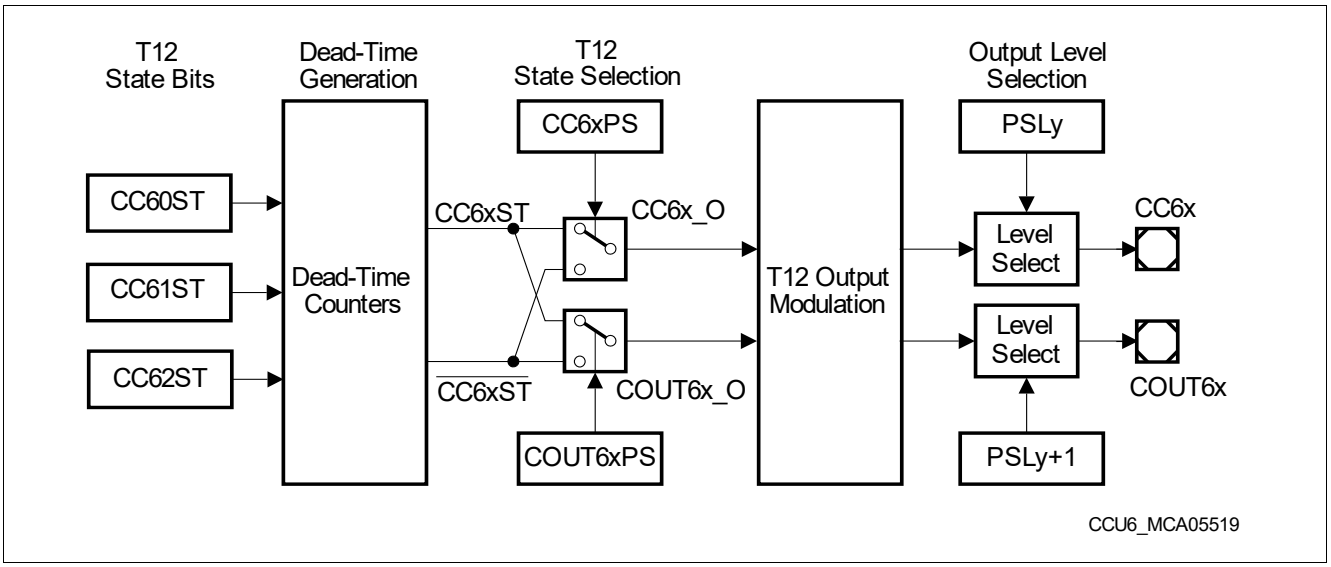


Figure 138 Compare Mode Simplified Output Path Diagram

The output path is based on signals that are defined as active or passive. The terms active and passive are not related to output levels, but to internal actions. This mainly applies for the modulation, where T12 and T13 signals are combined with the multi-channel signals and the trap function. The Output level Selection allows the user to define the output level at the output pin for the passive state (inverted level for the active state). It is recommended to configure this block in a way that an external power switch is switched off while the CCU6 delivers an output signal in the passive state.

18.3.4.1 Dead-Time Generation

The generation of (complementary) signals for the High Side and the low-side switches of one power inverter phase is based on the same compare channel. For example, if the High Side switch should be active while the T12 counter value is above the compare value (State Bit = 1), then the low-side switch should be active while the counter value is below the compare value (State Bit = 0).

In most cases, the switching behavior of the connected power switches is not symmetrical concerning the switch-on and switch-off times. A general problem arises if the time for switch-on is smaller than the time for switch-off of the power device. In this case, a short-circuit can occur in the inverter bridge leg, which may damage the complete system. In order to solve this problem by HW, this capture/compare unit contains a programmable Dead-Time Generation Block, that delays the passive to active edge of the switching signals by a programmable time (the active to passive edge is not delayed).

The Dead-Time Generation Block, illustrated in **Figure 139**, is built in a similar way for all three channels of T12. It is controlled by bits in register T12DTC. Any change of a CC6xST State Bit activates the corresponding Dead-Time Counter, that is clocked with the same input clock as T12 (f_{T12}). The length of the dead-time can be programmed by bit field DTM. This value is identical for all three channels. Writing TCTR4.DTRES = 1 sets all dead-times to passive.

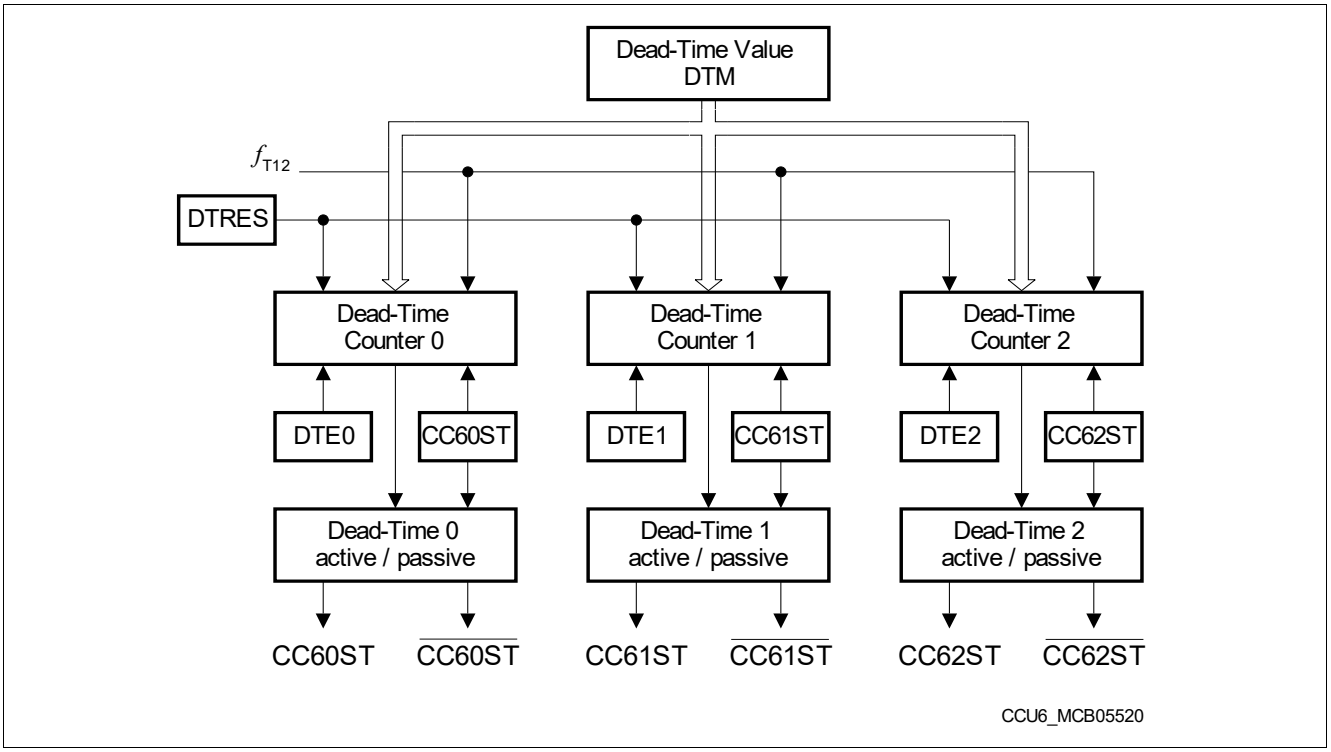


Figure 139 Dead-Time Generation Block Diagram

Each of the three dead-time counters has its individual dead-time enable bit, DTEx. An enabled dead-time counter generates a dead-time delaying the passive-to-active edge of the channel output signal. The change in a State Bit CC6xST is not taken into account while the dead-time generation of this channel is currently in progress (active). This avoids an unintentional additional dead-time if a State Bit CC6xST changes too early.

A disabled dead-time counter is always considered as passive and does not delay any edge of CC6xST.

Based on the State Bits CC6xST, the Dead-Time Generation Block outputs a direct signal CC6xST and an inverted signal $\overline{CC6xST}$ for each compare channel, each masked with the effect of the related Dead-Time Counters (waveforms illustrated in **Figure 140**).

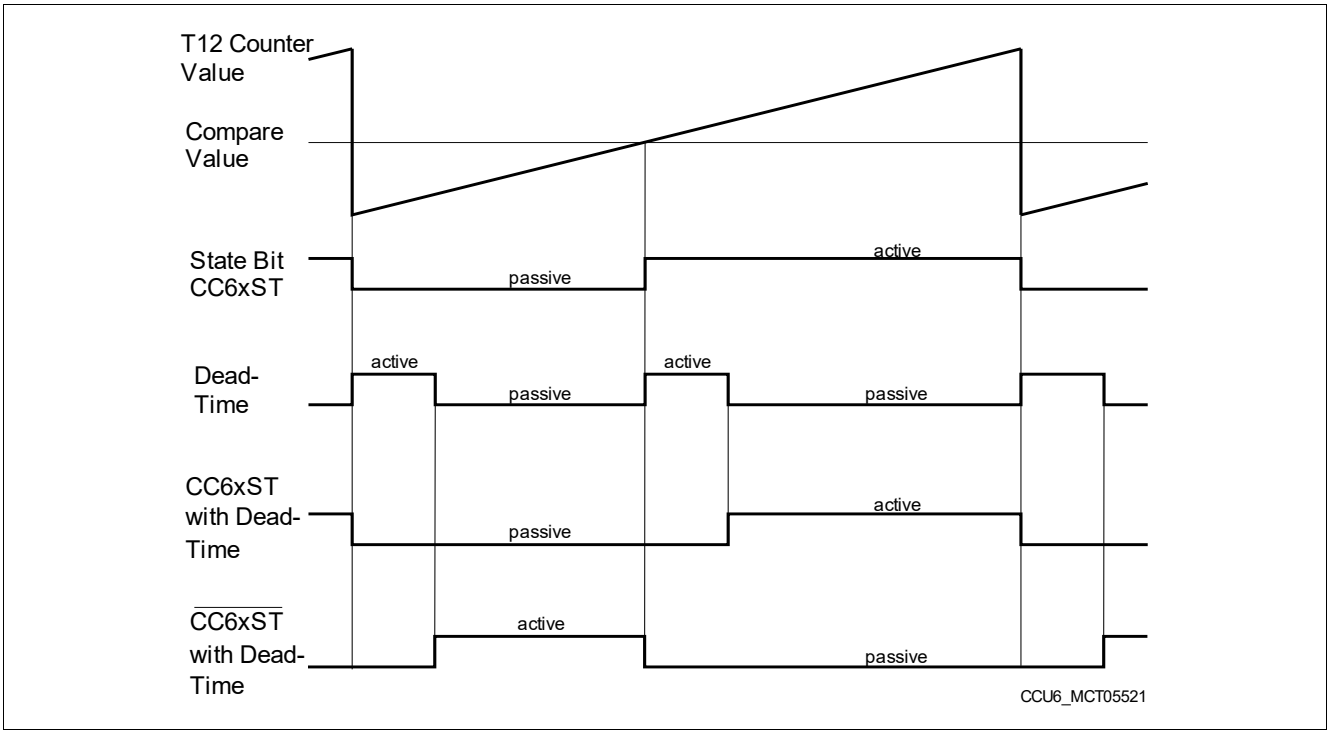


Figure 140 Dead-Time Generation Waveforms

18.3.4.2 State Selection

To support a wide range of power switches and drivers, the state selection offers the flexibility to define when an output can be active and can be modulated, especially useful for **complementary or multi-phase PWM** signals.

The state selection is based on the signals CC6xST and $\overline{\text{CC6xST}}$ delivered by the dead-time generator (see **Figure 138**). Both signals are never active at the same time, but can be passive at the same time. This happens during the dead-time of each compare channel after a change of the corresponding State Bit CC6xST.

The user can select independently for each output signal CC6xO and COUT6xO if it should be active before or after the compare value has been reached (see register CMPSTAT). With this selection, the active (conducting) phases of complementary power switches in a power inverter bridge leg can be positioned with respect to the compare value (e.g. signal CC6xO can be active before, whereas COUT6xO can be active after the compare value is reached). Like this, the output modulation, the trap logic and the output level selection can be programmed independently for each output signal, although two output signals are referring to the same compare channel.

18.3.4.3 Output Modulation and Level Selection

The last block of the data path is the Output Modulation block. Here, all the modulation sources and the trap functionality are combined and control the actual level of the output pins (controlled by the modulation enable bits T1xMODENy and MCMEN in register MODCTR). The following signal sources can be combined here **for each T12 output signal** (see **Figure 141** for compare channel CC60):

- A **T12 related compare signal** CC6x_O (for outputs CC6x) or COUT6x_O (for outputs COUT6x) delivered by the T12 block (state selection with dead-time) with an individual enable bit T12MODENy per output signal (y = 0, 2, 4 for outputs CC6x and y = 1, 3, 5 for outputs COUT6x)
- The **T13 related compare signal** CC63_O delivered by the T13 state selection with an individual enable bit T13MODENy per output signal (y = 0, 2, 4 for outputs CC6x and y = 1, 3, 5 for outputs COUT6x)
- A **multi-channel output signal** MCMPy (y = 0, 2, 4 for outputs CC6x and y = 1, 3, 5 for outputs COUT6x) with a common enable bit MCMEN
- The **trap state** TRPS with an individual enable bit TRPENy per output signal (y = 0, 2, 4 for outputs CC6x and y = 1, 3, 5 for outputs COUT6x)

If one of the modulation input signals CC6x_O/COUT6x_O, CC63_O, or MCMPy of an output modulation block is enabled and is at passive state, the modulated is also in passive state, regardless of the state of the other signals that are enabled. Only if all enabled signals are in active state the modulated output shows an active state. If no modulation input is enabled, the output is in passive state.

If the Trap State is active (TRPS = 1), then the outputs that are enabled for the trap signal (by TRPENy = 1) are set to the passive state.

The output of each of the modulation control blocks is connected to a level select block that is configured by register PSLR. It offers the option to determine the actual output level of a pin, depending on the state of the output line (decoupling of active/passive state and output polarity) as specified by the Passive State Select bit PSLy. If the modulated output signal is in the passive state, the level specified directly by PSLy is output. If it is in the active state, the inverted level of PSLy is output. This allows the user to adapt the polarity of an active output signal to the connected circuitry.

The PSLy bits have shadow registers to allow for updates without undesired pulses on the output lines. The bits related to CC6x and COUT6x (x = 0, 1, 2) are updated with the T12 shadow transfer signal (T12_ST). A read action returns the actually used values, whereas a write action targets the shadow bits. Providing a shadow register for the PSL value as well as for other values related to the generation of the PWM signal facilitates a concurrent update by software for all relevant parameters.

Figure 141 shows the output modulation structure for compare channel CC60 (output signals CC60 and COUT60). A similar structure is implemented for the other two compare channels CC61 and CC62.

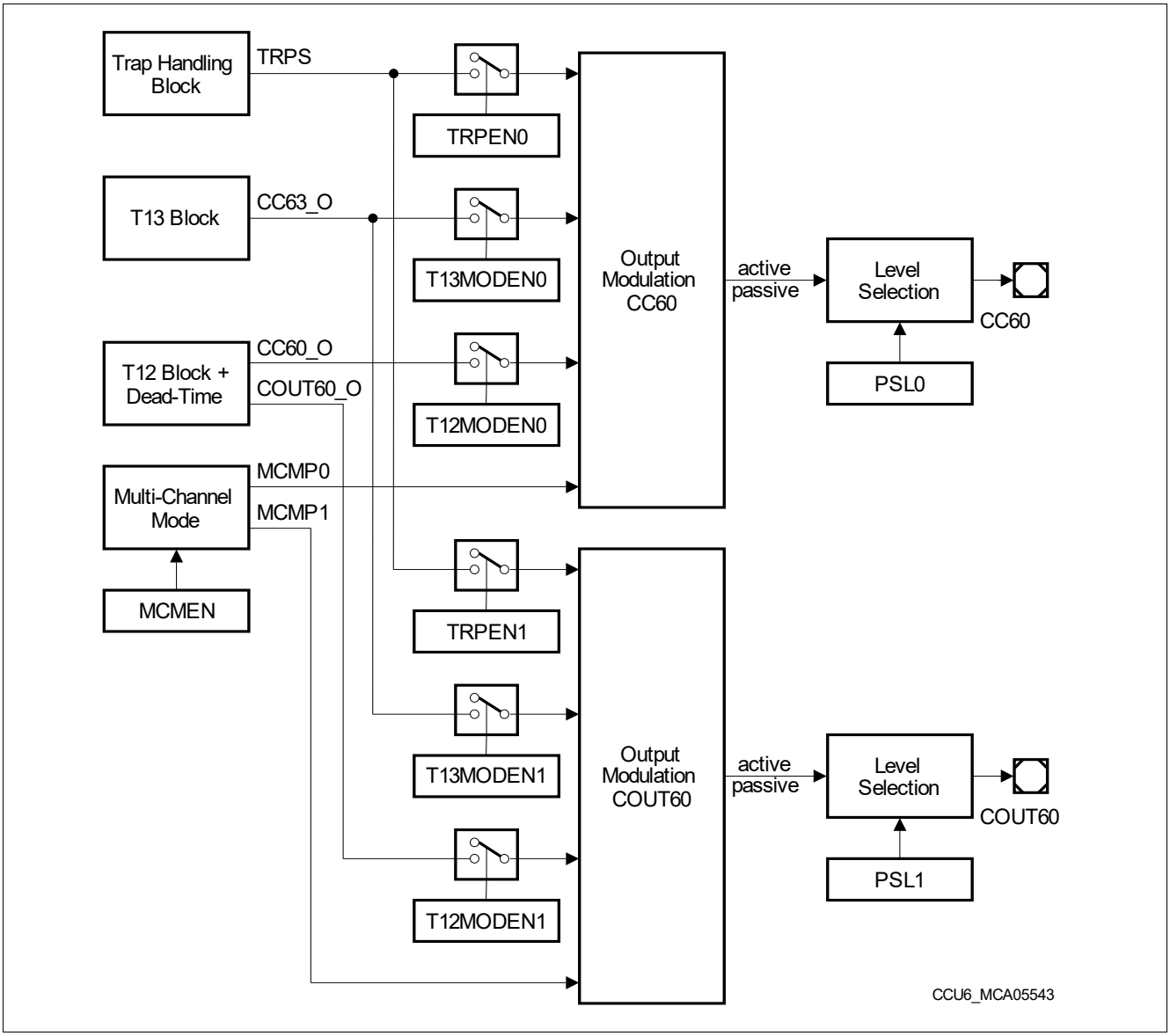


Figure 141 Output Modulation for Compare Channel CC60

18.3.5 T12 Capture Modes

Each of the three channels of the T12 Block can also be used to capture T12 time information in response to an external signal CC6xIN.

In capture mode, the interrupt event CC6x_R is detected when a rising edge is detected at the input CC6xIN, whereas the interrupt event CC6x_F is detected when a falling edge is detected.

There are a number of different modes for capture operation. In all modes, both of the registers of a channel are used. The selection of the capture modes is done via the T12MSEL.MSEL6x bit fields and can be selected individually for each of the channels.

Table 297 Capture Modes Overview

MSEL6x	Mode	Signal	Active Edge	CC6nSR Stored in	T12 Stored in
0100_B	1	CC6xIN	Rising	–	CC6xR
		CC6xIN	Falling	–	CC6xSR
0101_B	2	CC6xIN	Rising	CC6xR	CC6xSR
0110_B	3	CC6xIN	Falling	CC6xR	CC6xSR
0111_B	4	CC6xIN	Any	CC6xR	CC6xSR

Figure 142 illustrates **Capture Mode 1**. When a rising edge (0-to-1 transition) is detected at the corresponding input signal CC6xIN, the current contents of Timer T12 are captured into register CC6xR. When a falling edge (1-to-0 transition) is detected at the input signal CC6xIN, the contents of Timer T12 are captured into register CC6xSR.

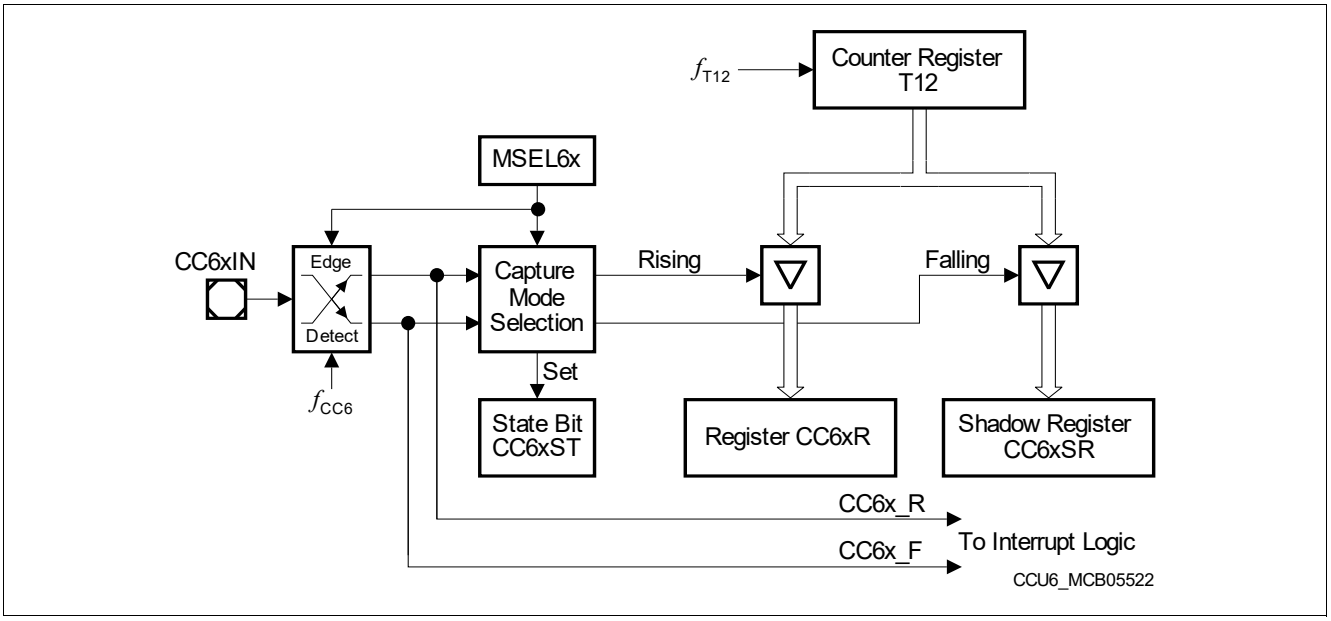


Figure 142 Capture Mode 1 Block Diagram

Capture Modes 2, 3 and 4 are shown in **Figure 143**. They differ only in the active edge causing the capture operation. In each of the three modes, when the selected edge is detected at the corresponding input signal CC6xIN, the current contents of the shadow register CC6xSR are transferred into register CC6xR, and the current Timer T12 contents are captured in register CC6xSR (simultaneous transfer). The active edge is a rising edge of CC6xIN for Capture Mode 2, a falling edge for Mode 3, and both, a rising or a falling edge for Capture Mode 4, as shown in **Table 297**. These capture modes are very useful in cases where there is little time between two consecutive edges of the input signal.

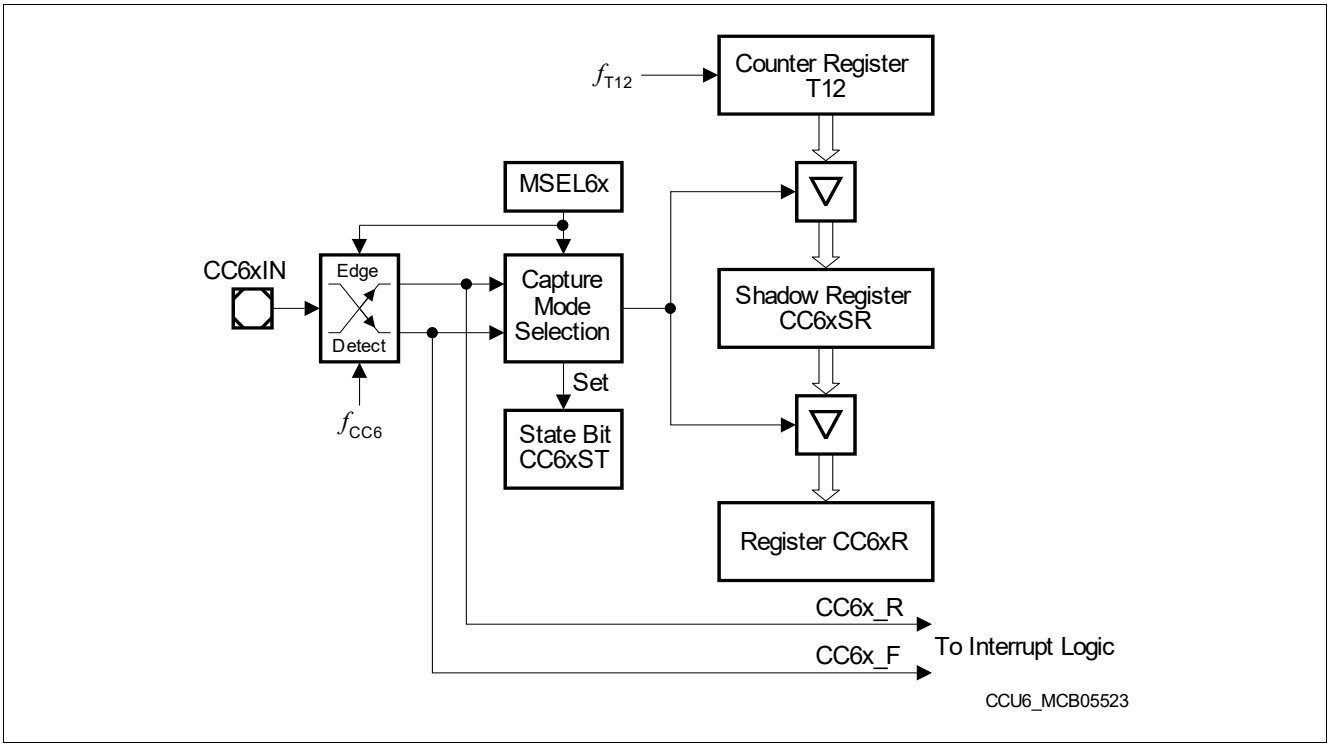


Figure 143 Capture Modes 2, 3 and 4 Block Diagram

Five further capture modes are called **Multi-Input Capture Modes**, as they use two different external inputs, signal CC6xIN and signal CCPOSx.

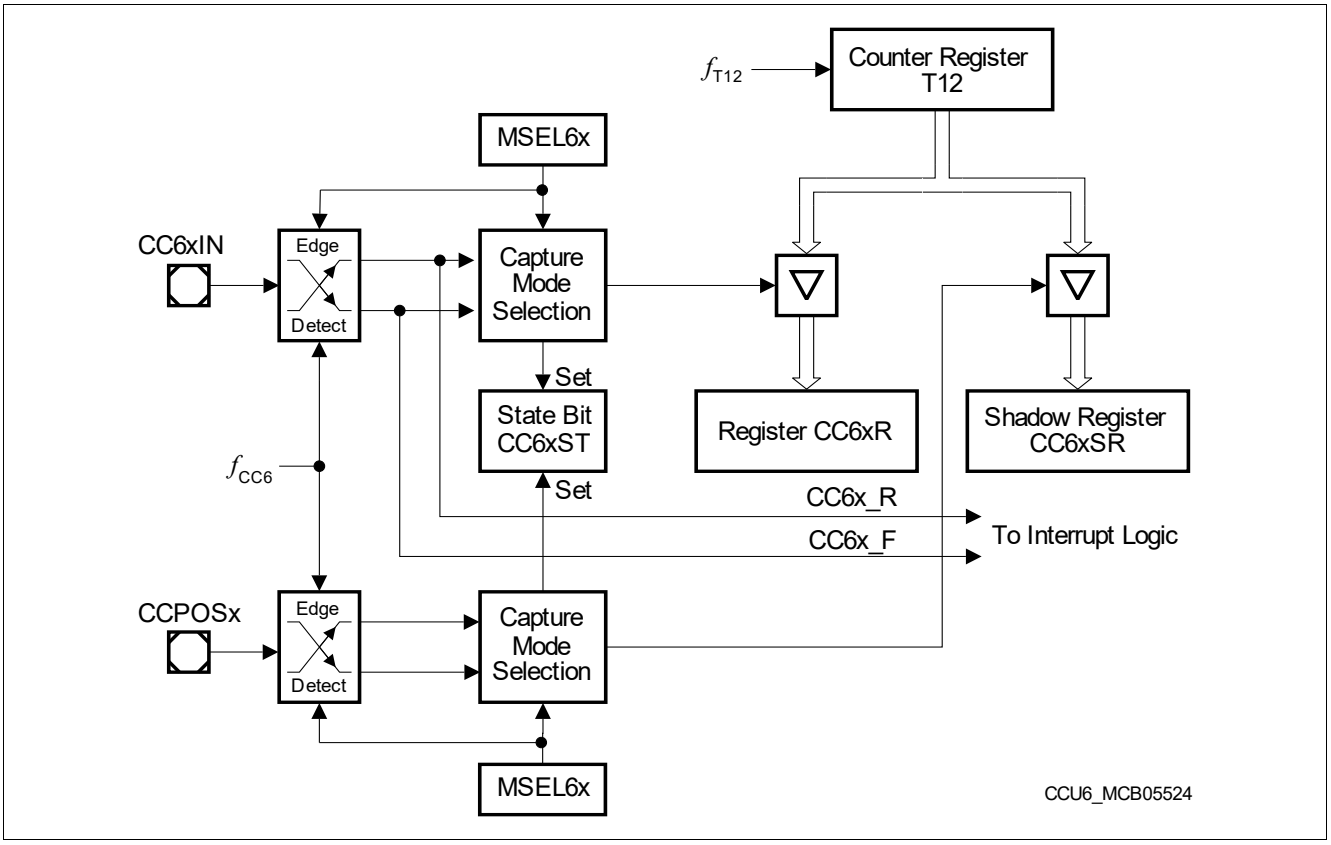


Figure 144 Multi-Input Capture Modes Block Diagram

In each of these modes, the current T12 contents are captured in register CC6xR in response to a selected event at signal CC6xIN, and in register CC6xSR in response to a selected event at signal CCPOSx. The possible events can be opposite input transitions, or the same transitions, or any transition at the two inputs. The different options are detailed in **Table 298**.

In each of the various capture modes, the Channel State Bit, CC6xST, is set to 1 when the selected capture trigger event at signal CC6xIN or CCPOSx has occurred. The State Bit is not cleared by hardware, but can be cleared by software.

In addition, appropriate signal lines to the interrupt logic are activated, that can generate an interrupt request to the CPU. Regardless of the selected active edge, all edges detected at signal CC6xIN can lead to the activation of the appropriate interrupt request line (see also **Section 18.8**).

Table 298 Multi-Input Capture Modes Overview

MSEL6x	Mode	Signal	Active Edge	T12 Stored in
1010_B	$_B$, 5	CC6xIN	Rising	CC6xR
		CCPOSx	Falling	CC6xSR
1011_B	$_B$, 6	CC6xIN	Falling	CC6xR
		CCPOSx	Rising	CC6xSR
1100_B	$_B$, 7	CC6xIN	Rising	CC6xR
		CCPOSx	Rising	CC6xSR

Table 298 Multi-Input Capture Modes Overview (cont'd)

MSEL6x	Mode	Signal	Active Edge	T12 Stored in
1101_B	$_B$, 8	CC6xIN	Falling	CC6xR
		CCPOSx	Falling	CC6xSR
1110_B	$_B$, 9	CC6xIN	Any	CC6xR
		CCPOSx	Any	CC6xSR
1111_B	–	reserved (no capture or compare action)		

18.3.6 T12 Shadow Register Transfer

A special shadow transfer signal (T12_ST) can be generated to facilitate updating the period and compare values of the compare channels CC60, CC61, and CC62 synchronously to the operation of T12. Providing a shadow register for values defining one PWM period facilitates a concurrent update by software for all relevant parameters. The next PWM period can run with a new set of parameters. The generation of this signal is requested by software via bit TCTR0.STE12 (set by writing 1 to the write-only bit TCTR4.T12STR, cleared by writing 1 to the write-only bit TCTR4.T12STD).

Figure 145 shows the shadow register structure and the shadow transfer signals, as well as on the read/write accessibility of the various registers.

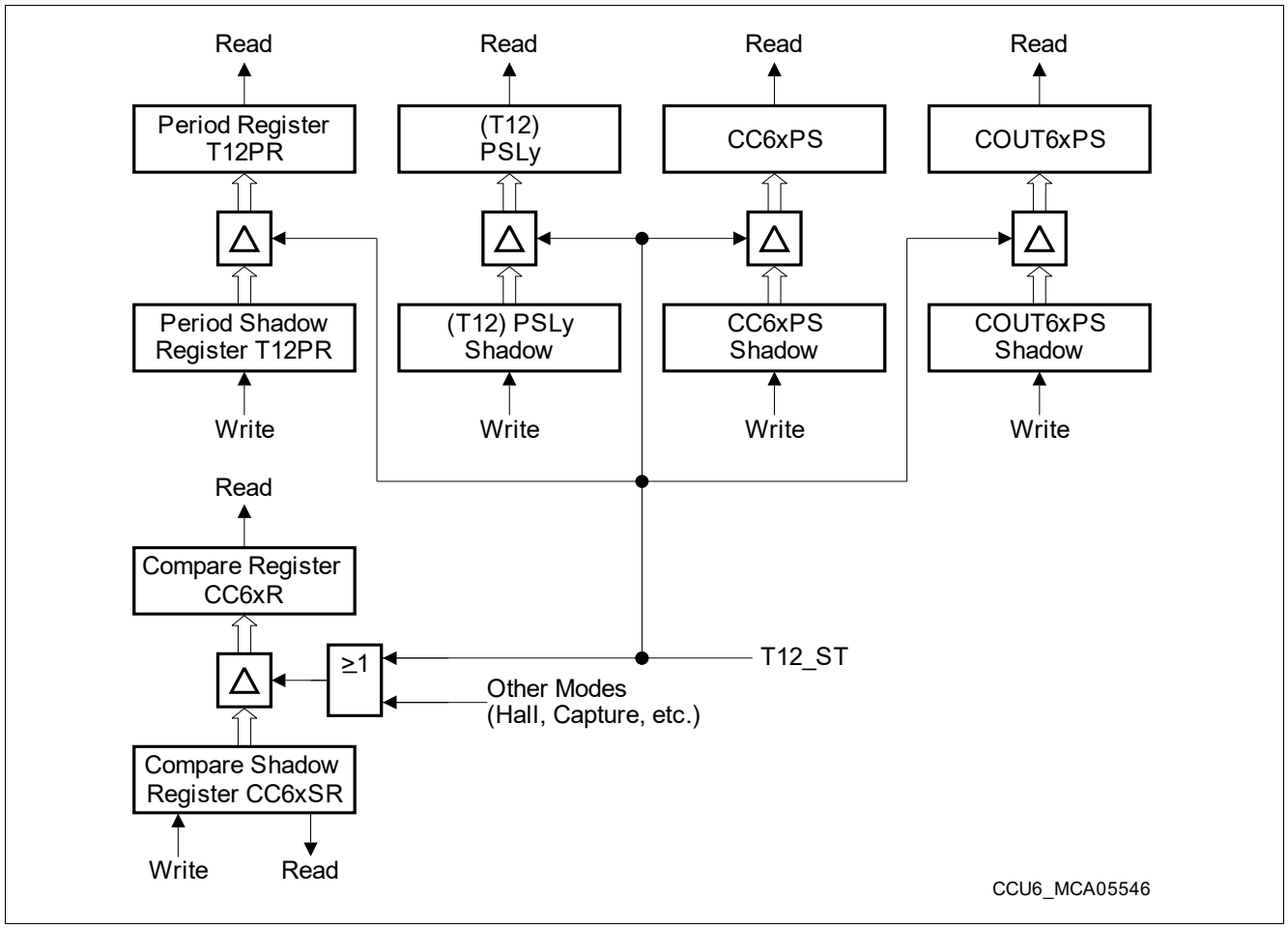


Figure 145 T12 Shadow Register Overview

A T12 shadow register transfer takes place (T12_ST active):

- while timer T12 is not running (T12R = 0), or
- STE12 = 1 and a Period-Match is detected while counting up, or
- STE12 = 1 and a One-Match is detected while counting down

When signal T12_ST is active, a shadow register transfer is triggered with the next cycle of the T12 clock. Bit STE12 is automatically cleared with the shadow register transfer.

18.3.7 Timer T12 Operating Mode Selection

The operating mode for the T12 channels are defined by the bit fields T12MSEL.MSEL6x.

Table 299 T12 Capture/Compare Modes Overview

MSEL6x	Selected operating mode
0000_B, 1111_B	Capture/Compare modes switched off
0001_B, 0010_B, 0011_B	Compare mode, see **Section 18.3.3** same behavior for all three codings
$01XX_B$	Double-Register Capture modes, see **Section 18.3.5**
1000_B	Hall Sensor Mode, see **Section 18.7** In order to properly enable this mode, all three MSEL6x fields have to be programmed to Hall Sensor mode.
1001_B	Hysteresis-like compare mode, see **Section 18.3.3.3**
1010_B, 1011_B, 1100_B, 1101_B, 1110_B	Multi-Input Capture modes, see **Section 18.3.5**

The clocking and counting scheme of the timers are controlled by the timer control registers TCTR0 and TCTR2. Specific actions are triggered by write operations to register TCTR4.

18.4 Operating Timer T13

Timer T13 is implemented similarly to Timer T12, but only with one channel in compare mode. A 16-bit up-counter is connected to a channel register via a comparator, that generates a signal when the counter contents match the contents of the channel register. A variety of control functions facilitate the adaptation of the T13 structure to different application needs. In addition, T13 can be started synchronously to timer T12 events.

This section provides information about:

- T13 overview (see **Section 18.4.1**)
- Counting scheme (see **Section 18.4.2**)
- Compare mode (see **Section 18.4.3**)
- Compare output path (see **Section 18.4.4**)
- Shadow register transfer (see **Section 18.4.5**)

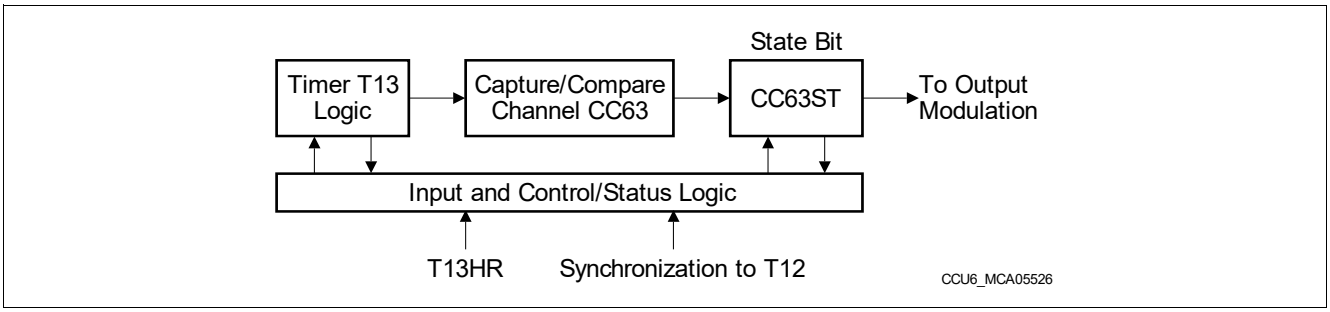


Figure 146 Overview Diagram of the Timer T13 Block

18.4.1 T13 Overview

Figure 147 shows a detailed block diagram of Timer T13. The functions of the timer T12 block are controlled by bits in registers TCTR0, TCTR2, and PISEL2.

Timer T13 receives its input clock, f_{T13}, from the module clock f_{CC6} via a programmable prescaler and an optional 1/256 divider or from an input signal T13HR. T13 can only count up (similar to the Edge-Aligned mode of T12).

Via a comparator, the timer T13 Counter Register T13 is connected to the Period Register T13PR. This register determines the maximum count value for T13. When T13 reaches the period value, signal T13_PM (T13 Period Match) is generated and T13 is cleared to 0000_H with the next T13 clock edge. The Period Register receives a new period value from its Shadow Period Register, T13PS, that is loaded via software. The transfer of a new period value from the shadow register into T13PR is controlled via the ‘T13 Shadow Transfer’ control signal, T13_ST. The generation of this signal depends on the associated control bit STE13. Providing a shadow register for the period value as well as for other values related to the generation of the PWM signal facilitates a concurrent update by software for all relevant parameters (refer to **Table 18.4.5**). Another signal indicates whether the counter contents are equal to 0000_H (T13_ZM).

A Single-Shot control bit, T13SSC, enables an automatic stop of the timer when the current counting period is finished (see **Figure 149**).

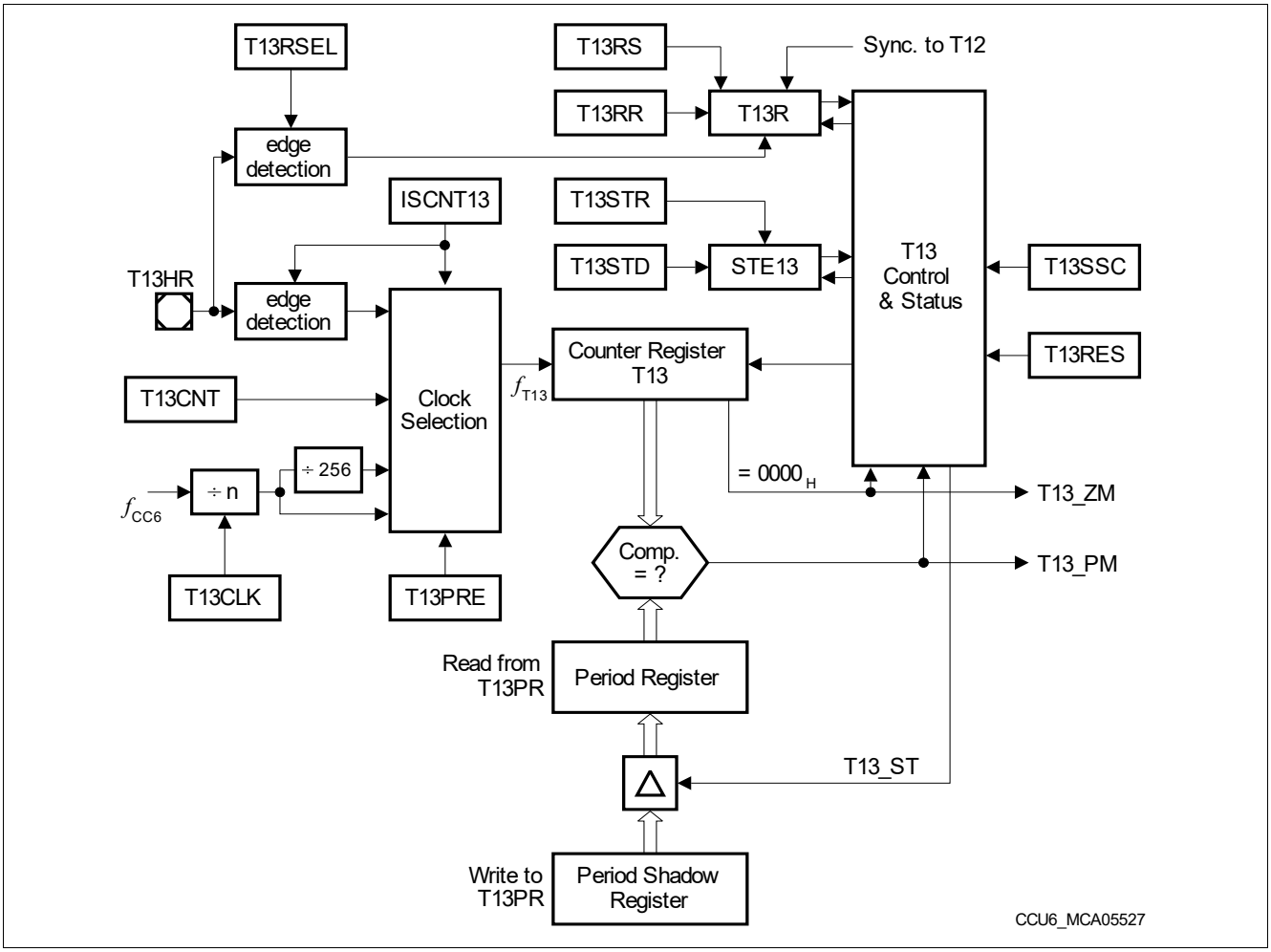


Figure 147 T13 Counter Logic and Period Comparators

The start or stop of T13 is controlled by the Run bit, T13R. This control bit can be set by software via the associated set/clear bits T13RS or T13RR in register TCTR4, or it is cleared by hardware according to preselected conditions (single-shot mode).

The timer T13 run bit T13R must not be set while the applied T13 period value is zero. Bit T13R can be set automatically if an event of T12 is detected to synchronize T13 timings to T12 events, e.g. to generate a programmable delay via T13 after an edge of a T12 compare channel before triggering an AD conversion (T13 can trigger ADC conversions).

Timer T13 can be cleared to 0000_H via control bit T13RES. Setting this write-only bit only clears the timer contents, but has no further effects, e.g., it does not stop the timer.

The generation of the T13 shadow transfer control signal, T13_ST, is enabled via bit STE13. This bit can be set or cleared by software indirectly through its associated set/reset control bits T13STR and T13STD.

Two bit fields, T13TEC and T13TED, control the synchronization of T13 to Timer T12 events. T13TEC selects the trigger event, while T13TED determines for which T12 count direction the trigger should be active.

While Timer T13 is running, write accesses to the count register T13 are not taken into account. If T13 is stopped, write actions to register T13 are immediately taken into account.

Note: The T13 Period Register and its associated shadow register are located at the same physical address. A write access to this address targets the Shadow Register, while a read access reads from the actual period register.

18.4.2 T13 Counting Scheme

This section describes the clocking and the counting capabilities of T13.

18.4.2.1 Clock Selection

In **Timer Mode** (PISEL2. ISCNT13 = 00_B), the input clock f_{T13} of Timer T13 is derived from the internal module clock f_{CC6} through a programmable prescaler and an optional 1/256 divider. The resulting prescaler factors are listed in **Table 300**. The prescaler of T13 is cleared while T13 is not running (TCTR0.T13R = 0) to ensure reproducible timings and delays.

Table 300 Timer T13 Input Clock Options

T13CLK	Resulting Input Clock f_{T13} Prescaler Off (T13PRE = 0)	Resulting Input Clock f_{T13} Prescaler On (T13PRE = 1)
000_B	f_{CC6}	f_{CC6} / 256
001_B	f_{CC6} / 2	f_{CC6} / 512
010_B	f_{CC6} / 4	f_{CC6} / 1024
011_B	f_{CC6} / 8	f_{CC6} / 2048
100_B	f_{CC6} / 16	f_{CC6} / 4096
101_B	f_{CC6} / 32	f_{CC6} / 8192
110_B	f_{CC6} / 64	f_{CC6} / 16384
111_B	f_{CC6} / 128	f_{CC6} / 32768

In **Counter Mode**, timer T13 counts one step:

- If a 1 is written to TCTR4.T13CNT and PISEL2.ISCNT13 = 01_B
- If a rising edge of input signal T13HR is detected and PISEL2.ISCNT13 = 10_B
- If a falling edge of input signal T13HR is detected and PISEL2.ISCNT13 = 11_B

18.4.2.2 T13 Counting

The period of the timer is determined by the value in the period Register T13PR according to the following formula:

$T13_{PER}$ = <Period-Value> + 1; in T13 clocks (f_{T13}) (18.3)

Timer T13 can only count up, comparable to the Edge-Aligned mode of T12. This leads to very simple 'counting rule' for the T13 counter:

- The counter is cleared with the next T13 clock edge if a Period-Match is detected. The counting direction is always upwards.

The behavior of T13 is illustrated in **Figure 148**.

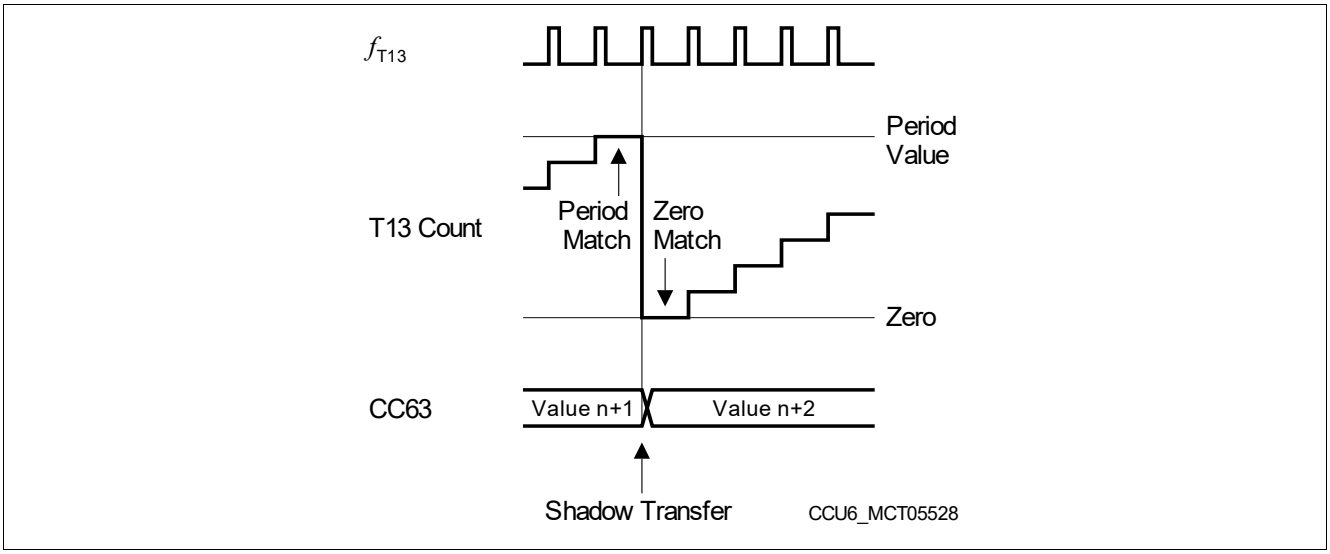


Figure 148 T13 Counting Sequence

18.4.2.3 Single-Shot Mode

In Single-Shot Mode, the timer run bit T13R is cleared by hardware. If bit T13SSC = 1, the timer T13 will stop when the current timer period is finished.

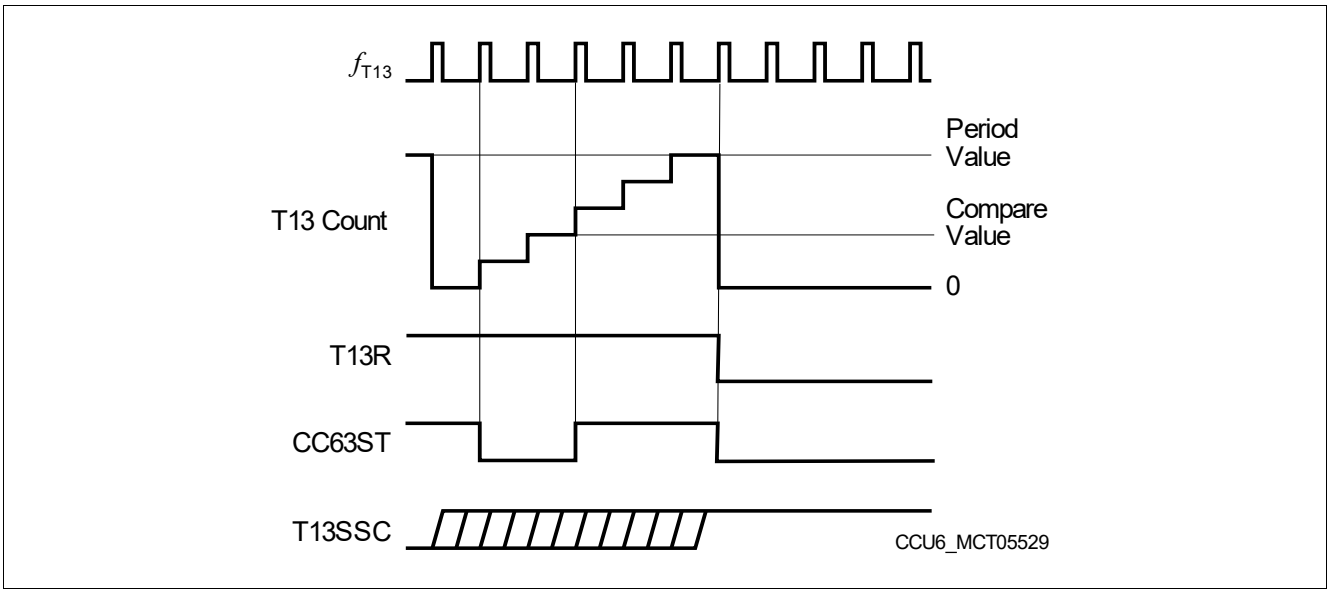


Figure 149 Single-Shot Operation of Timer T13

18.4.2.4 Synchronization to T12

Timer T13 can be synchronized to a T12 event. Bit fields T13TEC and T13TED select the event that is used to start Timer T13. The selected event sets bit T13R via HW, and T13 starts counting. Combined with the Single-Shot mode, this feature can be used to generate a programmable delay after a T12 event.

Figure 150 shows an example for the synchronization of T13 to a T12 event. Here, the selected event is a compare-match (compare value = 2) while counting up. The clocks of T12 and T13 can be different (other prescaler factor); the figure shows an example in which T13 is clocked with half the frequency of T12.

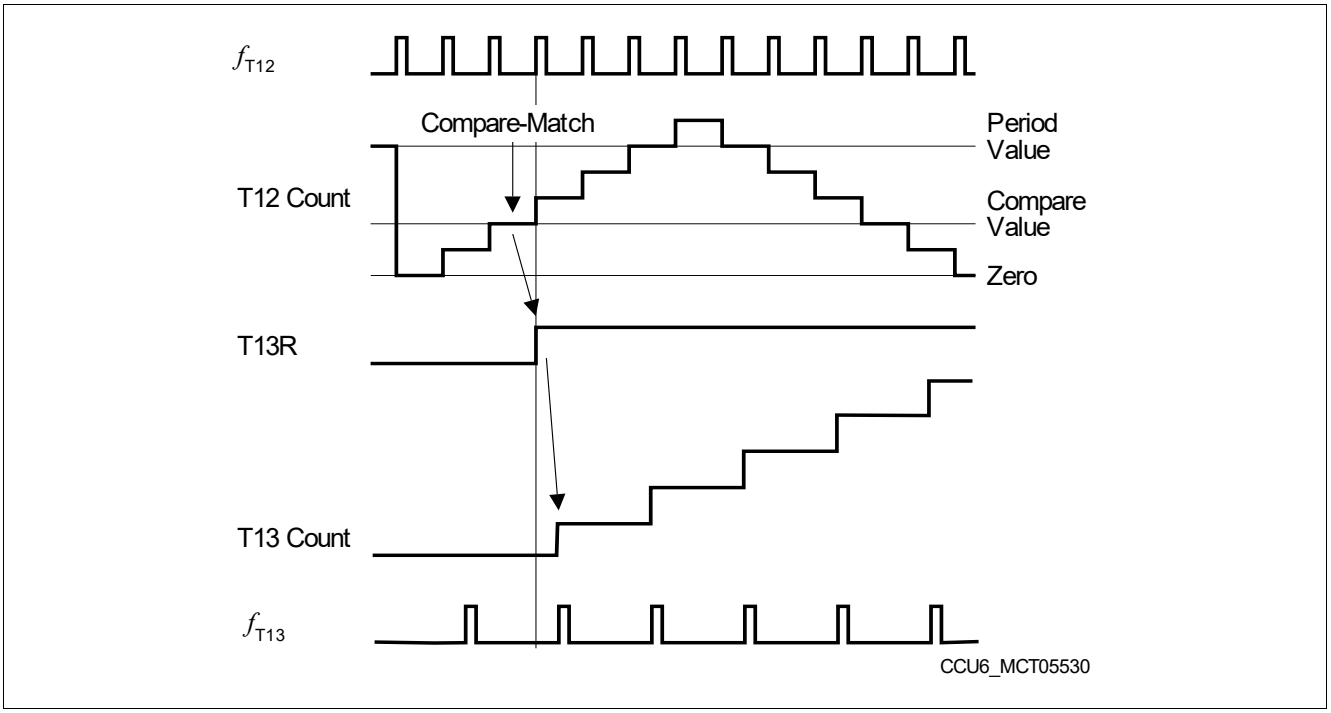


Figure 150 Synchronization of T13 to T12 Compare Match

Bit field T13TEC selects the trigger event to start T13 (automatic set of T13R for synchronization to T12 compare signals) according to the combinations shown in **Table 301**. Bit field T13TED additionally specifies for which count direction of T12 the selected trigger event should be regarded (see **Table 302**).

Table 301 T12 Trigger Event Selection

T13TEC	Selected Event
000_B	None
001_B	T12 Compare Event on Channel 0 (CM_CC60)
010_B	T12 Compare Event on Channel 1 (CM_CC61)
011_B	T12 Compare Event on Channel 2 (CM_CC62)
100_B	T12 Compare Event on any Channel (0, 1, 2)
101_B	T12 Period-Match (T12_PM)
110_B	T12 Zero-Match while counting up (T12_ZM and CDIR = 0)
111_B	Any Hall State Change

Table 302 T12 Trigger Event Additional Specifier

T13TED	Selected Event Specifier
00_B	Reserved, no action
01_B	Selected event is active while T12 is counting up (CDIR = 0)
10_B	Selected event is active while T12 is counting down (CDIR = 1)
11_B	Selected event is active independently of the count direction of T12

18.4.3 T13 Compare Mode

Associated with Timer T13 is one compare channel, that can perform compare operations with regard to the contents of the T13 counter.

Figure 146 gives an overview on the T13 channel in Compare Mode. The channel is connected to the T13 counter register via an equal-to comparator, generating a compare match signal when the contents of the counter matches the contents of the compare register.

The channel consists of the comparator and a double register structure - the actual compare register, CC63R, feeding the comparator, and an associated shadow register, CC63SR, that is preloaded by software and transferred into the compare register when signal T13 shadow transfer, T13_ST, gets active. Providing a shadow register for the compare value as well as for other values related to the generation of the PWM signal facilitates a concurrent update by software for all relevant parameters.

Associated with the channel is a State Bit, CMPSTAT.CC63ST, holding the status of the compare operation. **Figure 151** gives an overview on the logic for the State Bit.

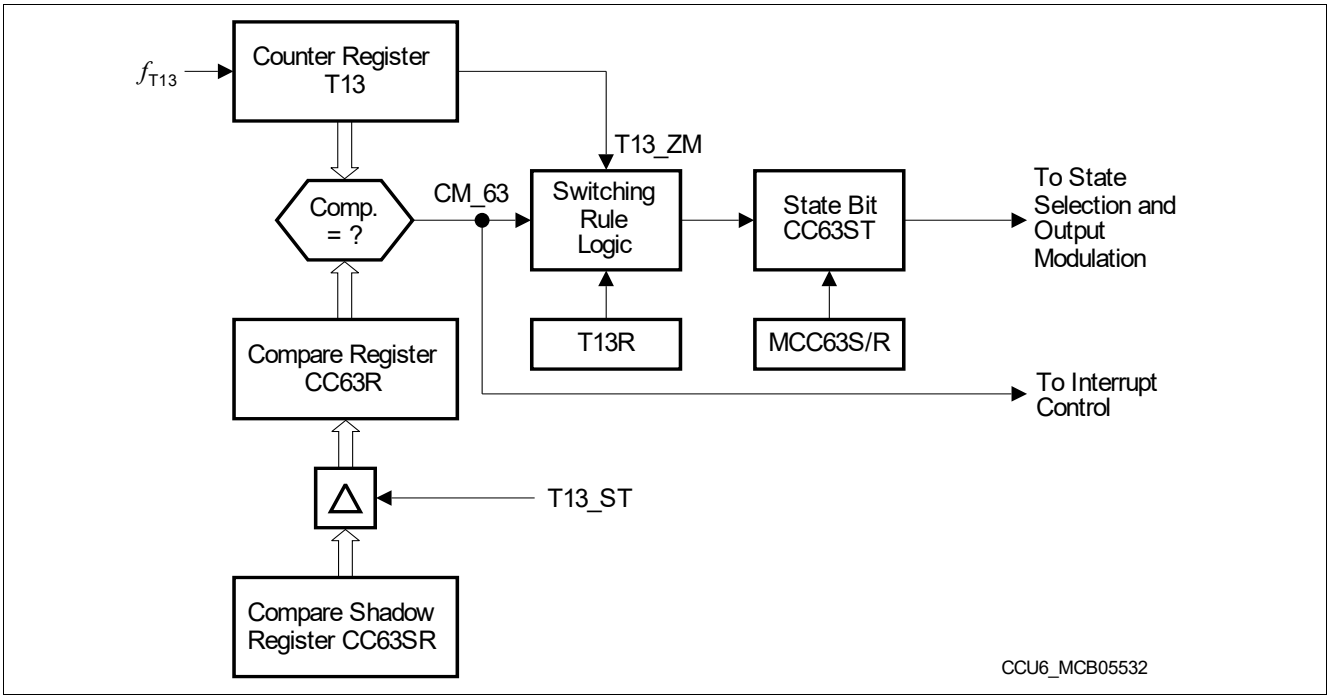


Figure 151 T13 State Bit Block Diagram

A compare interrupt event CM_63 is signaled when a compare match is detected. The actual setting of a State Bit has no influence on the interrupt generation.

The inputs to the switching rule logic for the CC63ST bit are the timer run bit (T13R), the timer zero-match signal (T13_ZM), and the actual individual compare-match signal CM_63. In addition, the state bit can be set or cleared by software via bits MCC63S and MCC63R in register CMPMODIF.

A modification of the State Bit CC63ST by hardware is only possible while Timer T13 is running (T13R = 1). If this is the case, the following switching rules apply for setting and resetting the State Bit in Compare Mode:

State Bit **CC63ST is set** to 1

- with the next T13 clock (f_{T13}) after a compare-match (T13 is always counting up) (i.e., when the counter is incremented above the compare value);
- with the next T13 clock (f_{T13}) after a zero-match AND a parallel compare-match.

State Bit **CC63ST is cleared** to 0

- with the next T13 clock (f_{T13}) after a zero-match AND NO parallel compare-match.

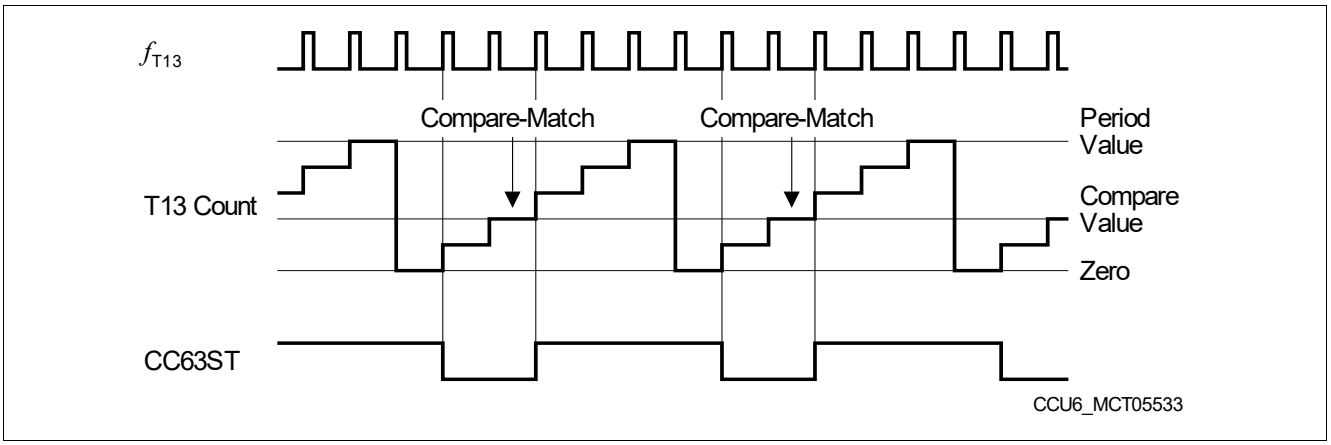


Figure 152 T13 Compare Operation

18.4.4 Compare Mode Output Path

Figure 153 gives an overview on the signal path from the channel State Bit CC63ST to its output pin COUT63. As illustrated, a user can determine the desired output behavior in relation to the current state of CC63ST. Please refer to **Section 18.3.4.3** for detailed information on the output modulation for T12 signals.

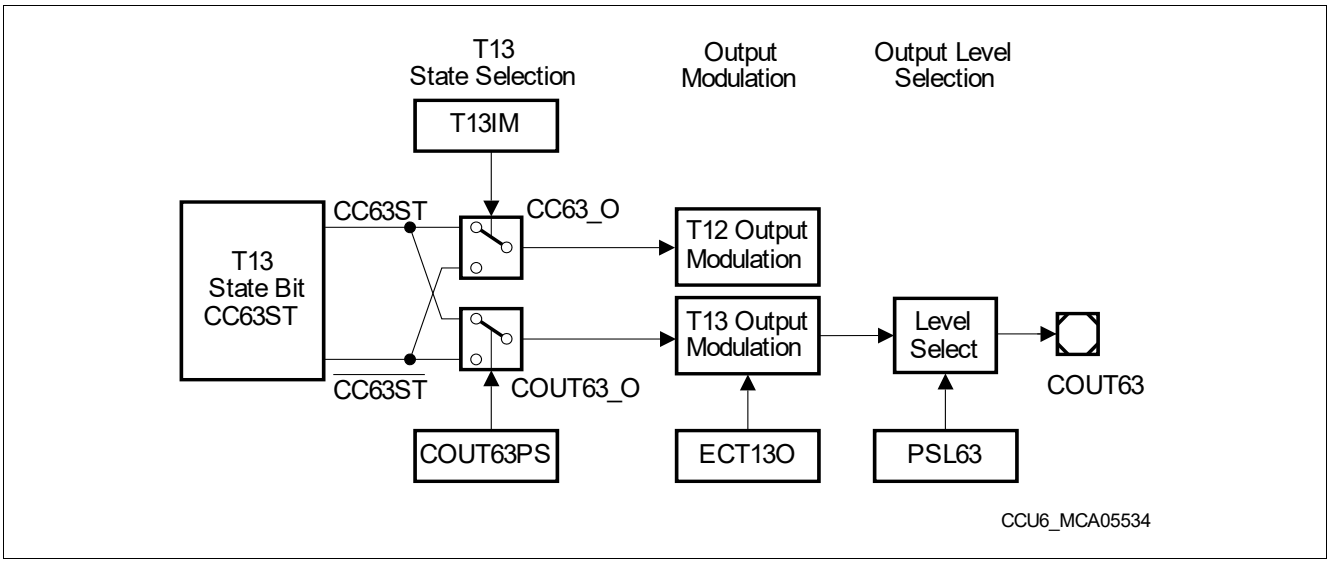


Figure 153 Channel 63 Output Path

The output line COUT63_O can generate a T13 PWM at the output pin COUT63. The signal CC63_O can be used to modulate the T12-related output signals with a T13 PWM. In order to decouple COUT63 from the internal modulation, the compare state leading to an active signal can be selected independently by bits T13IM and COUT63PS.

The last block of the data path is the Output Modulation block. Here, the modulation source T13 and the trap functionality are combined and control the actual level of the output pin COUT63 (see **Figure 154**):

- The **T13 related compare signal** COUT63_O delivered by the T13 state selection with the enable bit MODCTR.ECT13O
- The **trap state** TRPS with an individual enable bit TRPCTR.TRPEN13

If the modulation input signal COUT63_O is enabled (ECT13O = 1) and is at passive state, the modulated is also in passive state. If the modulation input is not enabled, the output is in passive state.

If the Trap State is active (TRPS = 1), then the output enabled for the trap signal (by TRPEN13 = 1) is set to the passive state.

The output of the modulation control block is connected to a level select block. It offers the option to determine the actual output level of a pin, depending on the state of the output line (decoupling of active/passive state and output polarity) as specified by the Passive State Select bit PSLR.PSL63. If the modulated output signal is in the passive state, the level specified directly by PSL63 is output. If it is in the active state, the inverted level of PSL63 is output. This allows the user to adapt the polarity of an active output signal to the connected circuitry.

The PSL63 bit has a shadow register to allow for updates with the T13 shadow transfer signal (T13_ST) without undesired pulses on the output lines. A read action returns the actually used value, whereas a write action targets the shadow bit. Providing a shadow register for the PSL value as well as for other values related to the generation of the PWM signal facilitates a concurrent update by software for all relevant parameters.

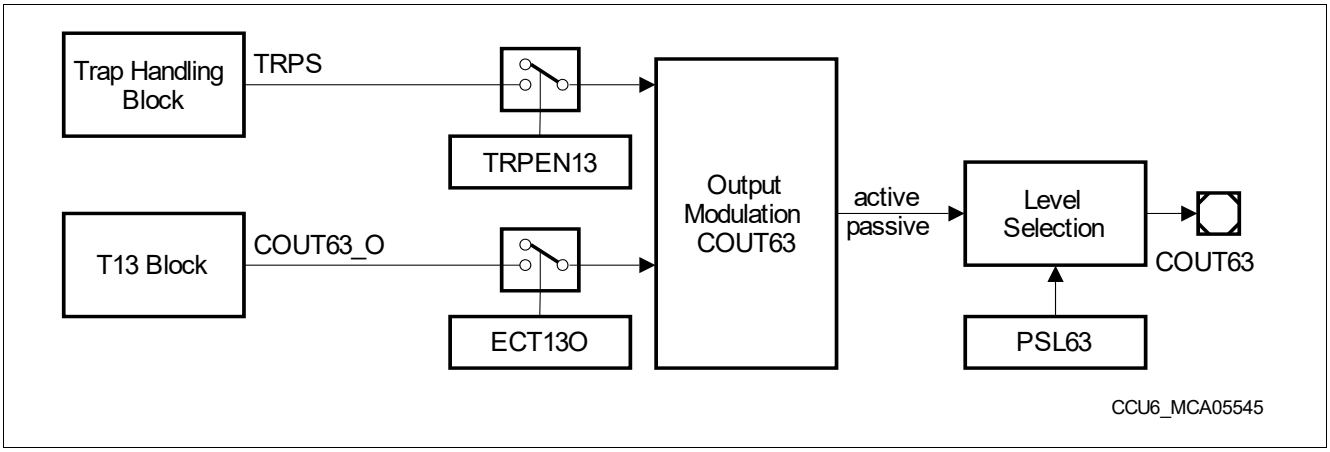


Figure 154 T13 Output Modulation

18.4.5 T13 Shadow Register Transfer

A special shadow transfer signal (T13_ST) can be generated to facilitate updating the period and compare values of the compare channel CC63 synchronously to the operation of T13. Providing a shadow register for values defining one PWM period facilitates a concurrent update by software for all relevant parameters. The next PWM period can run with a new set of parameters. The generation of this signal is requested by software via bit TCTR0.STE13 (set by writing 1 to the write-only bit TCTR4.T13STR, cleared by writing 1 to the write-only bit TCTR4.T13STD).

When signal T13_ST is active, a shadow register transfer is triggered with the next cycle of the T13 clock. Bit STE13 is automatically cleared with the shadow register transfer. A T13 shadow register transfer takes place (T13_ST active):

- while timer T13 is not running (T13R = 0), or
- STE13 = 1 and a Period-Match is detected while T13R = 1

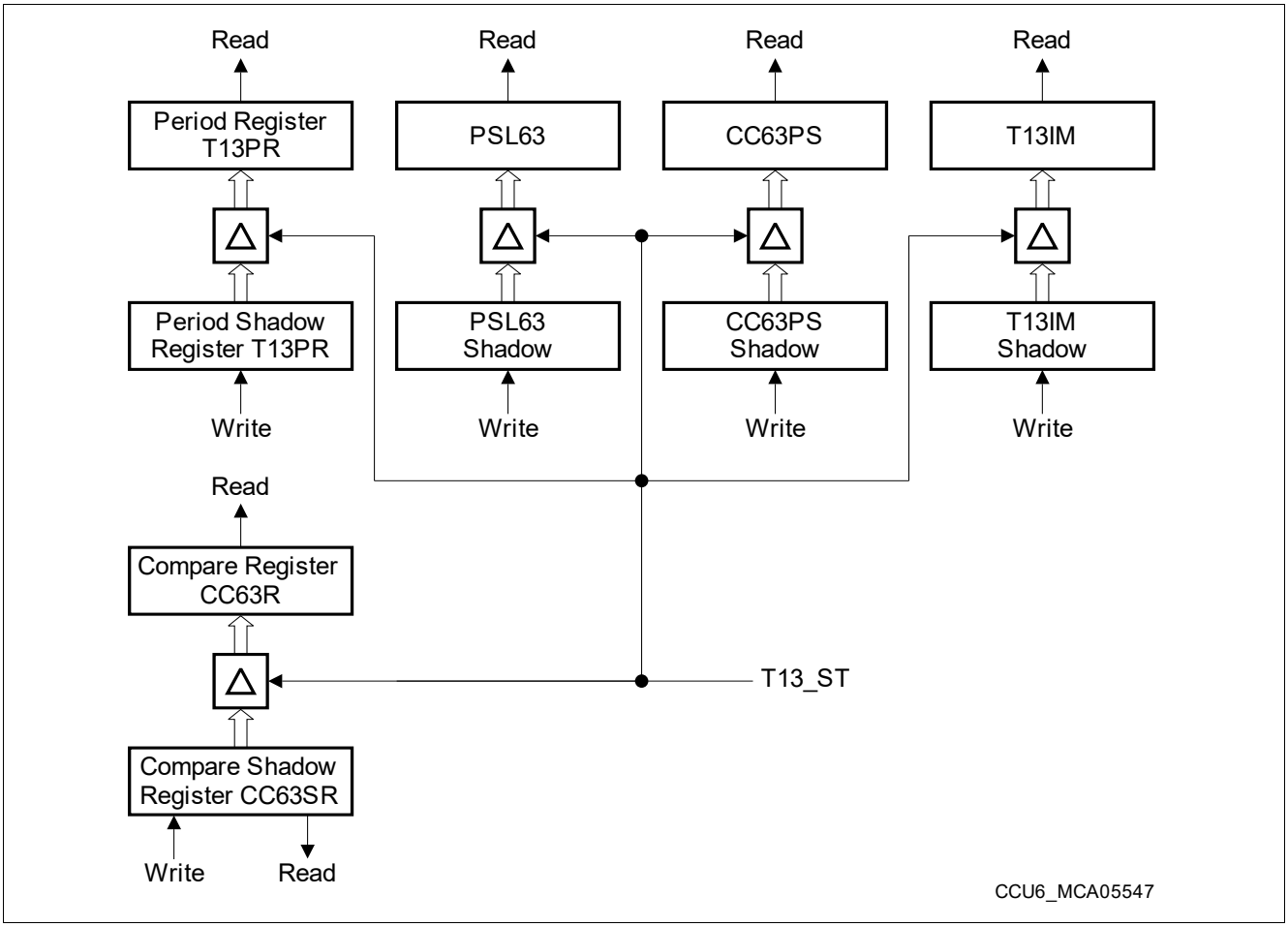


Figure 155 T13 Shadow Register Overview

18.5 Trap Handling

The trap functionality permits the PWM outputs to react on the state of the input signal $\overline{\text{CTRAP}}$. This functionality can be used to switch off the power devices if the trap input becomes active (e.g. to perform an emergency stop). The trap handling and the effect on the output modulation are controlled by the bits in the trap control register TRPCTR. The trap flags TRPF and TRPS are located in register IS and can be set/cleared by SW by writing to registers ISS and ISR.

Figure 156 gives an overview on the trap function.

The Trap Flag TRPF monitors the trap input and initiates the entry into the Trap State. The Trap State Bit TRPS determines the effect on the outputs and controls the exit of the Trap State.

When a trap condition is detected ($\overline{\text{CTRAP}}$ = 0) and the input is enabled (TRPPEN = 1), both, the Trap Flag TRPF and the Trap State Bit TRPS, are set to 1 (trap state active). The output of the Trap State Bit TRPS leads to the Output Modulation Blocks (for T12 and for T13) and can there deactivate the outputs (set them to the passive state). Individual enable control bits for each of the six T12-related outputs and the T13-related output facilitate a flexible adaptation to the application needs.

There are a number of different ways to exit the Trap State. This offers SW the option to select the best operation for the application. Exiting the Trap State can be done either immediately when the trap condition is removed ($\overline{\text{CTRAP}}$ = 1 or TRPPEN = 0), or under software control, or synchronously to the PWM generated by either Timer T12 or Timer T13.

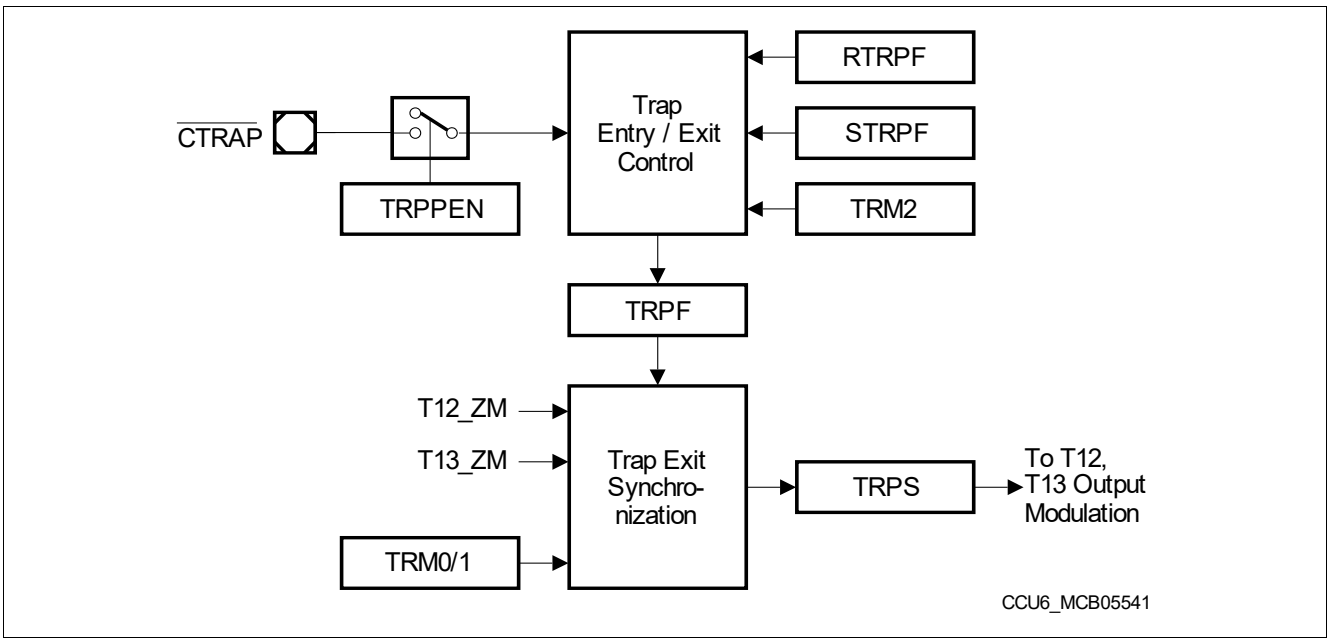


Figure 156 Trap Logic Block Diagram

Clearing of TRPF is controlled by the mode control bit TRPM2. If TRPM2 = 0, TRPF is automatically cleared by HW when $\overline{\text{CTRAP}}$ returns to the inactive level ($\overline{\text{CTRAP}}$ = 1) or if the trap input is disabled (TRPPEN = 0). When TRPM2 = 1, TRPF must be reset by SW after $\overline{\text{CTRAP}}$ has become inactive.

Clearing of TRPS is controlled by the mode control bits TRPM1 and TRPM0 (located in the Trap Control Register TRPCTR). A reset of TRPS terminates the Trap State and returns to normal operation. There are three options selected by TRPM1 and TRPM0. One is that the Trap State is left immediately when the Trap Flag TRPF is cleared, without any synchronization to timers T12 or T13. The other two options facilitate the synchronization of the termination of the Trap State to the count periods of either Timer T12 or Timer T13. **Figure 157** gives an overview on the associated operation.

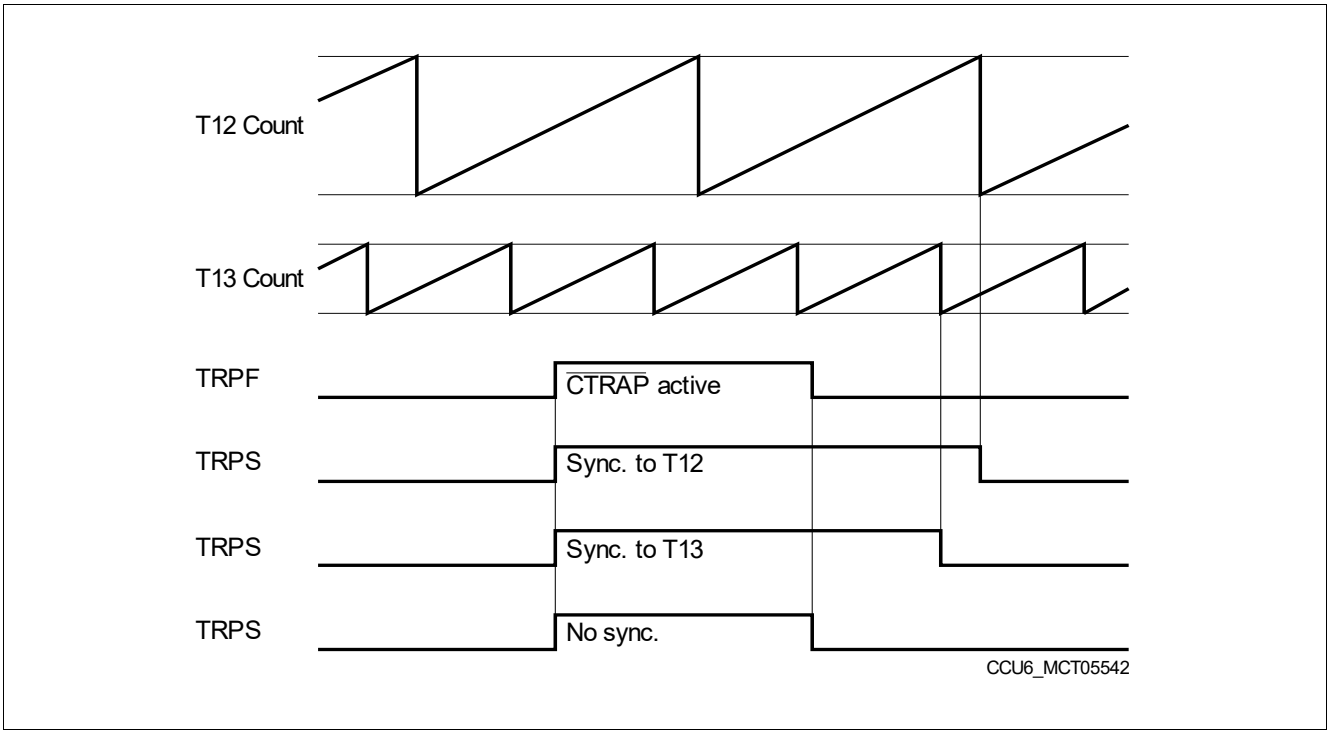


Figure 157 Trap State Synchronization (with TRM2 = 0)

18.6 Multi-Channel Mode

The Multi-Channel mode offers the possibility to modulate all six T12-related output signals with one instruction. The bits in bit field MCMOUT.MCMP are used to specify the outputs that may become active. If Multi-Channel mode is enabled (bit MODCTR.MCMEN = 1), only those outputs may become active, that have a 1 at the corresponding bit position in bit field MCMP.

This bit field has its own shadow bit field MCMOUTS.MCMPS, that can be written by software. The transfer of the new value in MCMPS to the bit field MCMP can be triggered by, and synchronized to, T12 or T13 events. This structure permits the software to write the new value, that is then taken into account by the hardware at a well-defined moment and synchronized to a PWM signal. This avoids unintended pulses due to unsynchronized modulation sources.

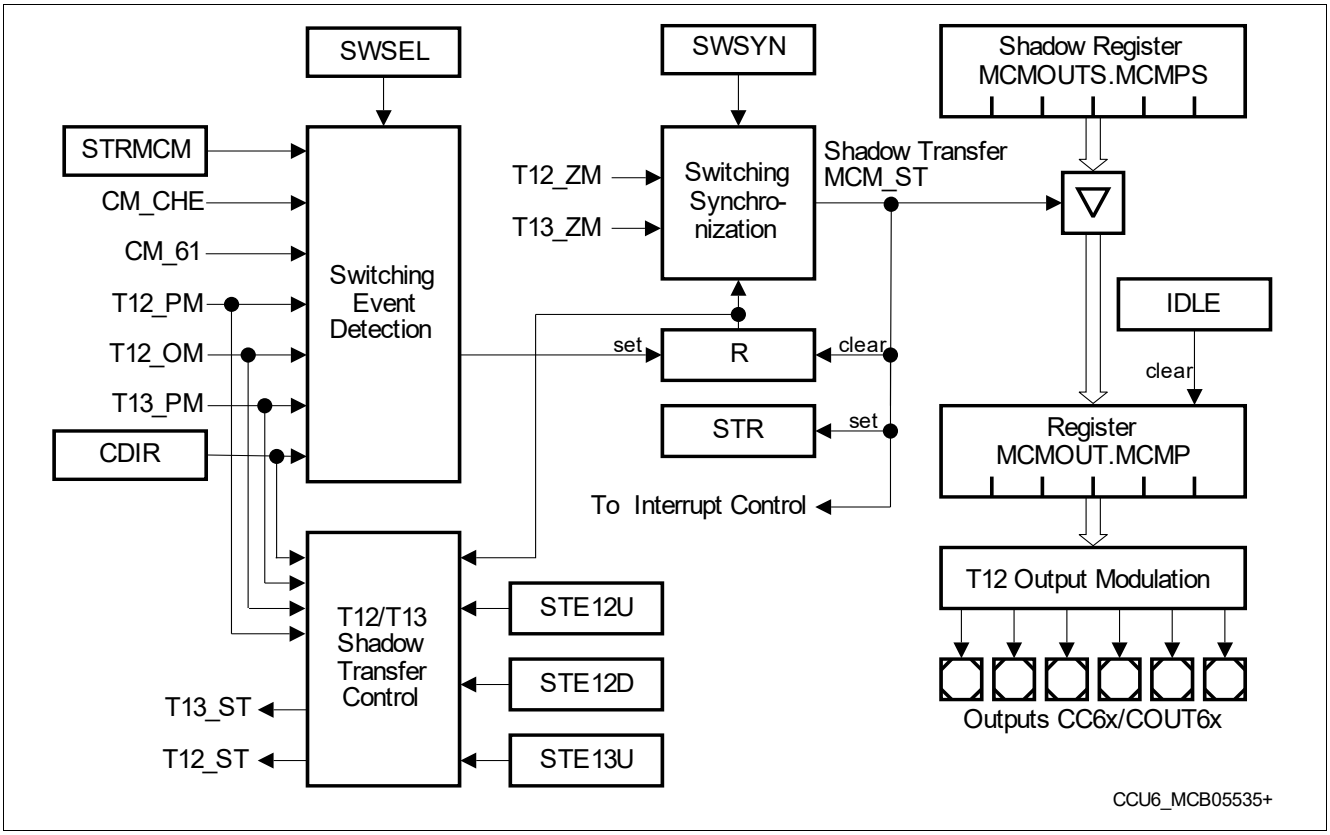


Figure 158 Multi-Channel Mode Block Diagram

Figure 158 shows the functional blocks for the Multi-Channel operation, controlled by bit fields in register MCMCTR. The event that triggers the update of bit field MCMP is chosen by SWSEL. In order to synchronize the update of MCMP to a PWM generated by T12 or T13, bit field SWSYN allows the selection of the synchronization event leading to the transfer from MCMPS to MCMP. Due to this structure, an update takes place with a new PWM period. A reminder flag R is set when the selected switching event occurs (the event is not necessarily synchronous to the modulating PWM), and is cleared when the transfer takes place. This flag can be monitored by software to check for the status of this logic block. If the shadow transfer from MCMPS to MCMP takes place, bit IS.STR becomes set and an interrupt can be generated.

In addition to the Multi-Channel shadow transfer event MCM_ST, the shadow transfers for T12 (T12_ST) and T13 (T13_ST) can be generated to allow concurrent updates of applied duty cycles for T12 and/or T13 modulation and Multi-Channel patterns.

If it is explicitly desired, the update takes place immediately with the occurrence of the selected event when the direct synchronization mode is selected. The update can also be requested by software by writing to bit

field MCMPS with the shadow transfer request bit STRMCM = 1. The option to trigger an update by SW is possible for all settings of SWSEL.

By using the direct mode and bit STRMCM = 1, the update takes place completely under software control.

Table 303 Multi-Channel Mode Switching Event Selection

SWSEL	Selected Event (see register MCMCTR)
000_B	No automatic event detection
001_B	Correct Hall Event (CM_CHE) detected at input signals CCPOSx without additional delay
010_B	T13 Period-Match (T13_PM)
011_B	T12 One-Match while counting down (T12_OM and CDIR = 1)
100_B	T12 Compare Channel 1 Event while counting up (CM_61 and CDIR = 0) to support the phase delay function by CC61 for block commutation mode.
101_B	T12 Period-Match while counting up (T12_PM and CDIR = 0)
110_B, 111_B	Reserved, no action

Table 304 Multi-Channel Mode Switching Synchronization

SWSYN	Synchronization Event (see register MCMCTR)
00_B	Direct Mode: the trigger event directly causes the shadow transfer
01_B	T13 Zero-Match (T13_ZM), the MCM shadow transfer is synchronized to a T13 PWM
10_B	T12 Zero-Match (T12_ZM), the MCM shadow transfer is synchronized to a T12 PWM
11_B	Reserved, no action

18.7 Hall Sensor Mode

For Brushless DC-Motors in block commutation mode, the Multi-Channel Mode has been introduced to provide efficient means for switching pattern generation. These patterns need to be output in relation to the angular position of the motor. For this, usually Hall sensors or Back-EMF sensing are used to determine the angular rotor position. The CCU6 provides three inputs, CCPOS0, CCPOS1, and CCPOS2, that can be used as inputs for the Hall sensors or the Back-EMF detection signals.

There is a strong correlation between the motor position and the output modulation pattern. When a certain position of the motor has been reached, indicated by the sampled Hall sensor inputs (the Hall pattern), the next, pre-determined Multi-Channel Modulation pattern has to be output. Because of different machine types, the modulation pattern for driving the motor can vary. Therefore, it is wishful to have a wide flexibility in defining the correlation between the Hall pattern and the corresponding Modulation pattern. Furthermore, a hardware mechanism significantly reduces the CPU for block-commutation.

The CCU6 offers the flexibility by having a register containing the currently assumed Hall pattern (CURH), the next expected Hall pattern (EXPH) and the corresponding output pattern (MCMP). A new Modulation pattern is output when the sampled Hall inputs match the expected ones (EXPH). To detect the next rotation phase (segment for block commutation), the CCU6 monitors the Hall inputs for changes. When the next expected Hall pattern is detected, the next corresponding Modulation pattern is output.

To increase for noise immunity (to a certain extend), the CCU6 offers the possibility to introduce a sampling delay for the Hall inputs. Some changes of the Hall inputs are not leading to the expected Hall pattern, because they are only short spikes due to noise. The Hall pattern compare logic compares the Hall inputs to the next expected pattern and also to the currently assumed pattern to filter out spikes.

For the Hall and Modulation output patterns, a double-register structure is implemented. While register MCMOUT holds the actually used values, its shadow register MCMOUTS can be loaded by software from a pre-defined table, holding the appropriate Hall and Modulation patterns for the given motor control.

A transfer from the shadow register into register MCMOUT can take place when a correct Hall pattern change is detected. Software can then load the next values into register MCMOUTS. It is also possible by software to force a transfer from MCMOUTS into MCMOUT.

Note: The Hall input signals CCPOSx and the CURH and EXPH bit fields are arranged in the following order:
CCPOS0 corresponds to CURH.0 (LSB) and EXPH.0 (LSB)
CCPOS1 corresponds to CURH.1 and EXPH.1
CCPOS2 corresponds to CURH.2 (MSB) and EXPH.2 (MSB)

18.7.1 Hall Pattern Evaluation

The Hall sensor inputs CCPOSx can be permanently monitored via an edge detection block (with the module clock f_{CC6}). In order to suppress spikes on the Hall inputs due to noise in rugged inverter environment, two optional noise filtering methods are supported by the Hall logic (both methods can be combined).

- Noise filtering with delay:
  For this function, the mode control bit fields MSEL6x for all T12 compare channels must be programmed to 1000_B and DBYP = 0. The selected event triggers Dead-Time Counter 0 to generate a programmable delay (defined by bit field DTM). When the delay has elapsed, the evaluation signal HCRDY becomes activated. Output modulation with T12 PWM signals is not possible in this mode.
- Noise filtering by synchronization to PWM:
  The Hall inputs are not permanently monitored by the edge detection block, but samples are taken only at defined points in time during a PWM period. This can be used to sample the Hall inputs when the switching noise (due to PWM) does not disturb the Hall input signals.

If neither the delay function of Dead-Time Counter 0 is not used for the Hall pattern evaluation nor the Hall mode for Brushless DC-Drive control is enabled, the timer T12 block is available for PWM generation and output modulation.

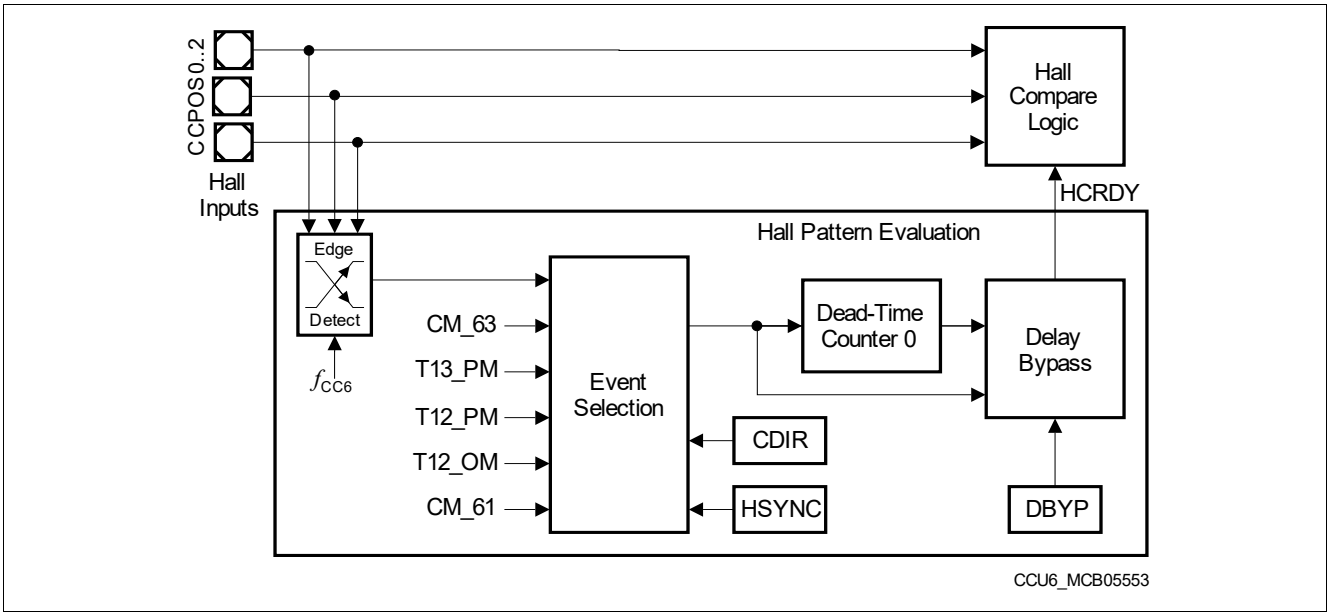


Figure 159 Hall Pattern Evaluation

If the evaluation signal HCRDY (Hall Compare Ready, see **Figure 160**) becomes activated, the Hall inputs are sampled and the Hall compare logic starts the evaluation of the Hall inputs.

Figure 159 illustrates the events for Hall pattern evaluation and the noise filter logic, **Table 305** summarizes the selectable trigger input signals.

Table 305 Hall Sensor Mode Trigger Event Selection

HSYNC	Selected Event (see register T12MSEL)
000_B	Any edge at any of the inputs CCPOSx, independent from any PWM signal (permanent check).
001_B	A T13 Compare-Match (CM_63).
010_B	A T13 Period-Match (T13_PM).
011_B	Hall sampling triggered by HW sources is switched off.

Table 305 Hall Sensor Mode Trigger Event Selection (cont'd)

HSYNC	Selected Event (see register T12MSEL)
100_B	A T12 Period-Match while counting up (T12_PM and CDIR = 0).
101_B	A T12 One-Match while counting down (T12_OM and CDIR = 1).
110_B	A T12 Compare-Match of compare channel CC61 while counting up (CM_61 and CDIR = 0).
111_B	A T12 Compare-Match of compare channel CC61 while counting down (CM_61 and CDIR = 1).

18.7.2 Hall Pattern Compare Logic

Figure 160 gives an overview on the double-register structure and the pattern compare logic. Software writes the next modulation pattern (MCMPS) and the corresponding current (CURHS) and expected (EXPHS) Hall patterns into the shadow register MCMOUTS. Register MCMOUT holds the actually used values CURH and EXPH. The modulation pattern MCMP is provided to the T12 Output Modulation block. The current (CURH) and expected (EXPH) Hall patterns are compared to the sampled Hall sensor inputs (visible in register CMPSTAT). Sampling of the inputs and the evaluation of the comparator outputs is triggered by the evaluation signal HCRDY (Hall Compare Ready), that is detailed in the next section.

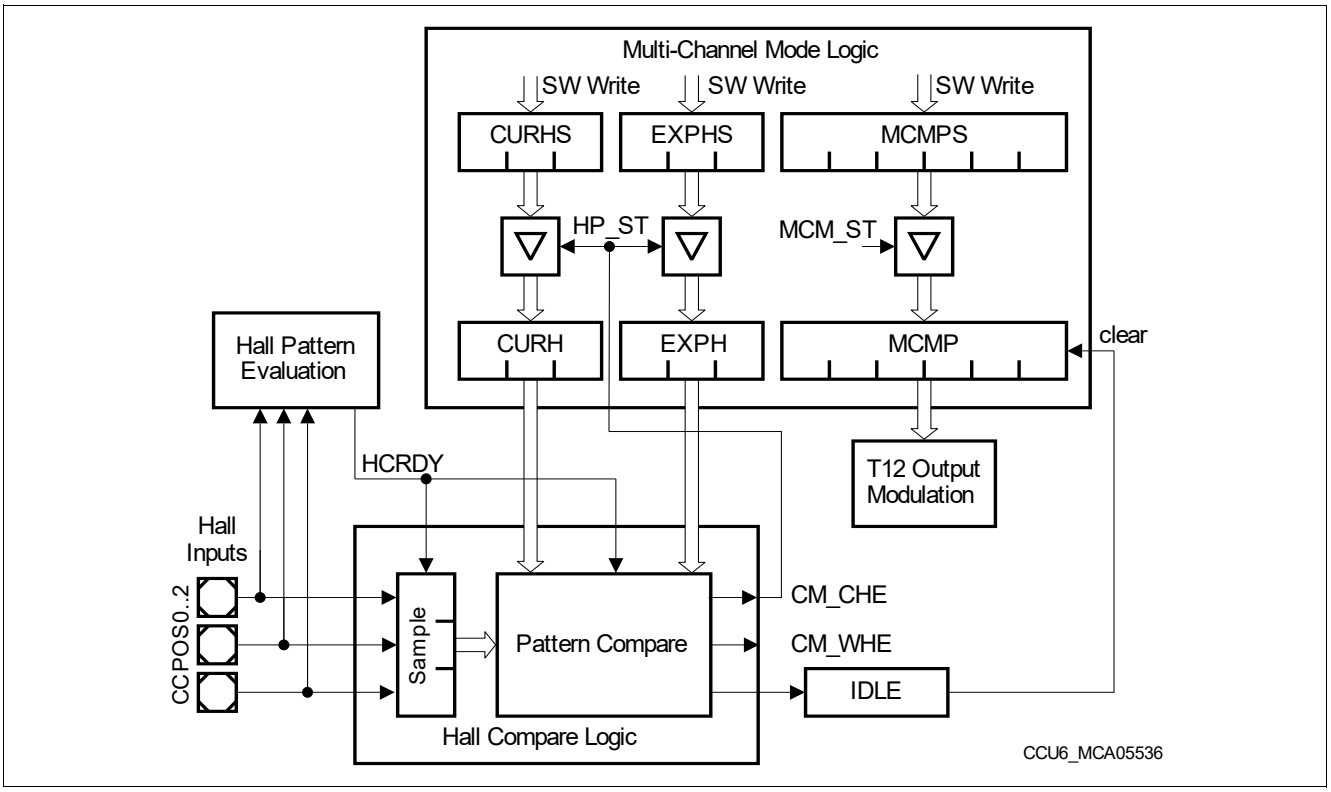


Figure 160 Hall Pattern Compare Logic

- If the sampled Hall pattern matches the value programmed in CURH, the detected transition was a spike (no Hall event) and no further actions are necessary.
- If the sampled Hall pattern matches the value programmed in EXPH, the detected transition was the expected event (correct Hall event CM_CHE) and the MCMP value has to change.
- If the sampled Hall pattern matches neither CURH nor EXPH, the transition was due to a major error (wrong Hall event CM_CWE) and can lead to an emergency shut down (IDLE).

At every correct Hall event (CM_CHE), the next Hall patterns are transferred from the shadow register MCMOUTS into MCMOUT (Hall pattern shadow transfer HP_ST), and a new Hall pattern with its corresponding output pattern can be loaded (e.g. from a predefined table in memory) by software into MCMOUTS. For the Modulation patterns, signal MCM_ST is used to trigger the transfer.

Loading this shadow register can also be done by writing MCMOUTS.STRHP = 1 (for EXPH and CURH) or MCMOUTS.STRMCMP = 1 (for MCMP).

18.7.3 Hall Mode Flags

Depending on the Hall pattern compare operation, a number of flags are set in order to indicate the status of the module and to trigger further actions and interrupt requests.

Flag IS.CHE (Correct Hall Event) is set by signal CM_CHE when the sampled Hall pattern matches the expected one (EXPH). This flag can also be set by SW by setting bit ISS.SCHE = 1. If enabled by bit IEN.ENCHE = 1, the set signal for CHE can also generate an interrupt request to the CPU. Bit field INP.INPCHE defines which service request output becomes activated in case of an interrupt request.To clear flag CHE, SW needs to write ISR.RCHE = 1.

Flag IS.WHE indicates a Wrong Hall Event. Its handling for flag setting and resetting as well as interrupt request generation are similar to the mechanism for flag CHE.

The implementation of flag STR is done in the same way as for CHE and WHE. This flag is set by HW by the shadow transfer signal MCM_ST (see also **Figure 158**).

Please note that for flags CHE, WHE, and STR, the interrupt request generation is triggered by the set signal for the flag. That means, a request can be generated even if the flag is already set. There is no need to clear the flag in order to enable further interrupt requests.

The implementation for the IDLE flag is different. It is set by HW through signal CM_WHE if enabled by bit ENIDLE. Software can also set the flag via bit SIDLE. As long as bit IDLE is set, the modulation pattern field MCMP is cleared to force the outputs to the passive state. Flag IDLE must be cleared by software by writing RIDLE = 1 in order to return to normal operation. To fully restart from IDLE mode, the transfer requests for the bit fields in register MCMOUTS to register MCMOUT have to be initiated by software via bits STRMCM and STRHP in register MCMOUTS. In this way, the release from IDLE mode is under software control, but can be performed synchronously to the PWM signal.

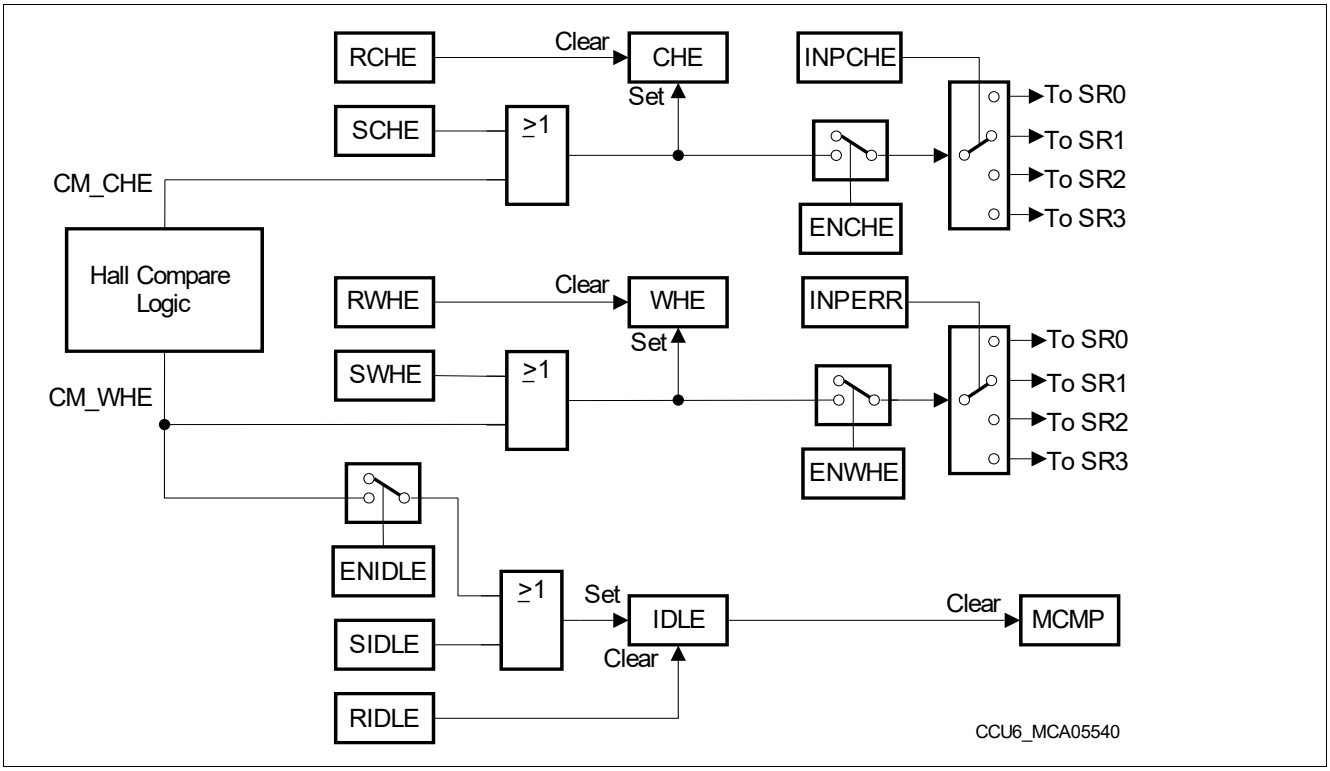


Figure 161 Hall Mode Flags

18.7.4 Hall Mode for Brushless DC-Motor Control

The CCU6 provides a mode for the Timer T12 Block especially targeted for convenient control of block commutation patterns for Brushless DC-Motors. This mode is selected by setting all T12MSEL.MSEL6x bit fields of the three T12 Channels to 1000_B.

In this mode, illustrated in **Figure 162**, channel CC60 is placed in capture mode to measure the time elapsed between the last two correct Hall events, channel CC61 in compare mode to provide a programmable phase delay between the Hall event and the application of a new PWM output pattern, and channel CC62 also in compare mode as first time-out criterion. A second time-out criterion can be built by the T12 period match event.

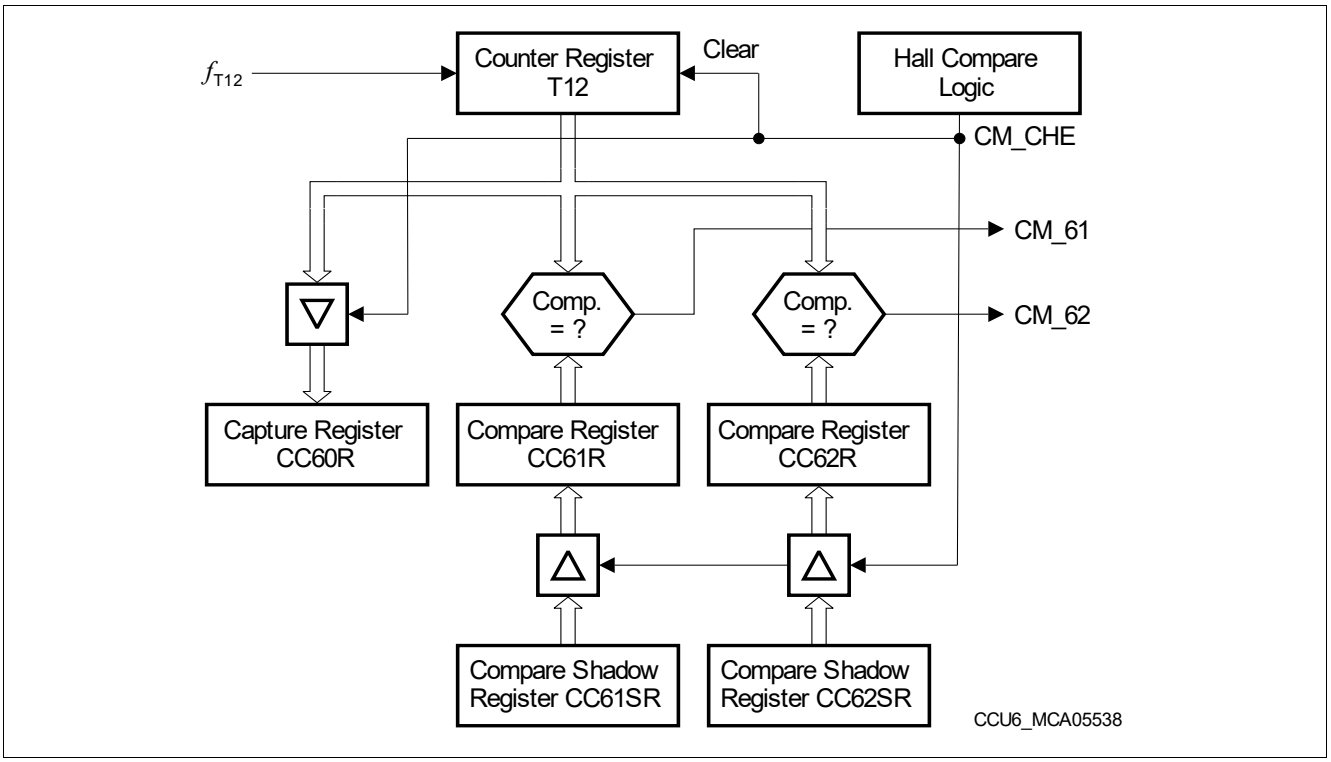


Figure 162 T12 Block in Hall Sensor Mode

The signal CM_CHE from the Hall compare logic is used to transfer the new compare values from the shadow registers CC6xSR into the actual compare registers CC6xR, performs the shadow transfer for the T12 period register, to capture the current T12 contents into register CC60R, and to clear T12.

Note: In this mode, the shadow transfer signal T12_ST is not generated. Not all shadow bits, such as the PSLy bits, will be transferred to their main registers. To program the main registers, SW needs to write to these registers while Timer T12 is stopped. In this case, a SW write actualizes both registers.

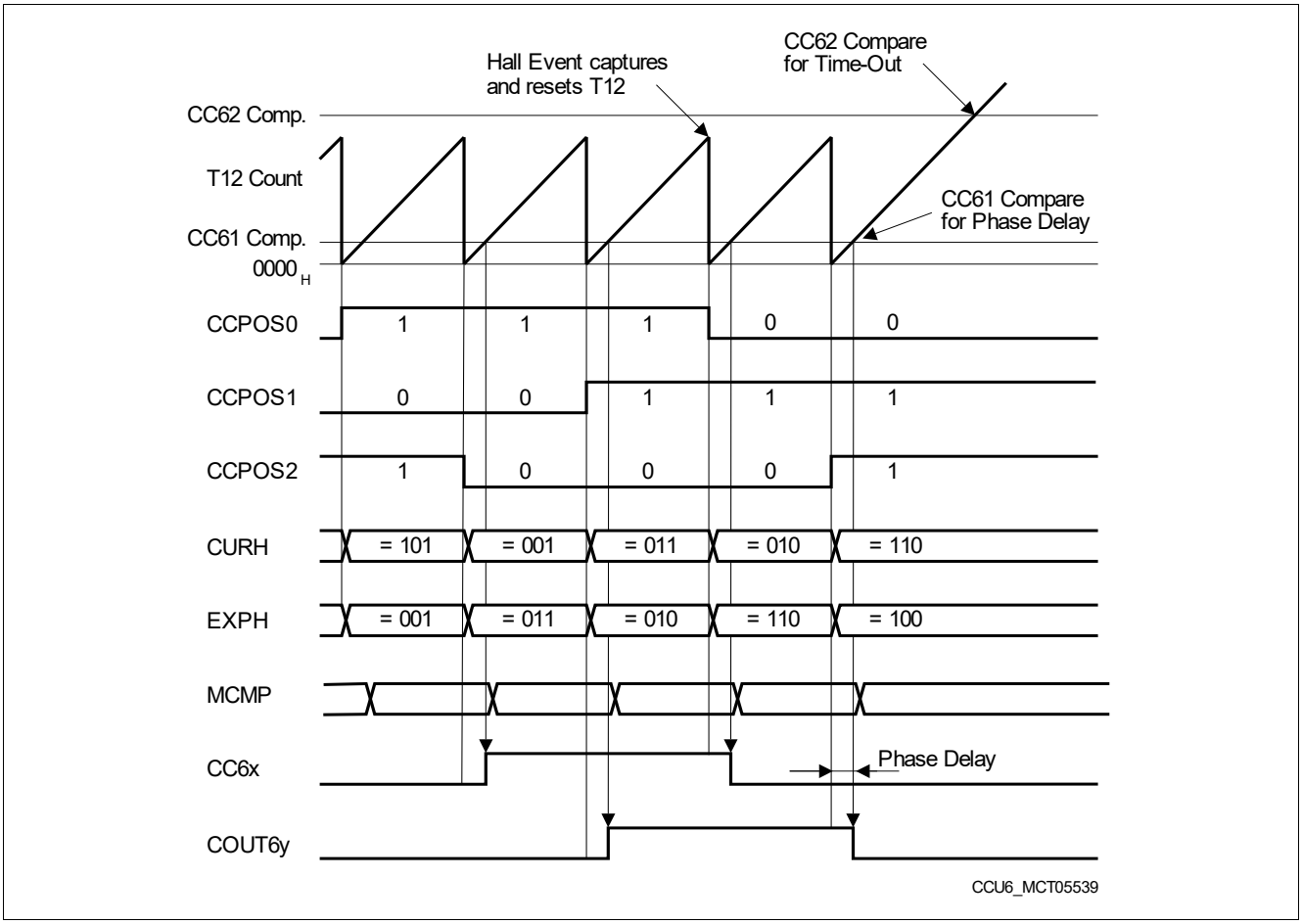


Figure 163 Brushless DC-Motor Control Example (all MSEL6x = 1000_B)

After the detection of an expected Hall pattern (CM_CHE active), the T12 count value is captured into channel CC60 (representing the actual rotor speed by measuring the elapsed time between the last two correct Hall events), and T12 is reset. When the timer reaches the compare value in channel CC61, the next multi-channel state is switched by triggering the shadow transfer of bit field MCMP (if enabled in bit field **SWEN**). This trigger event can be combined with the synchronization of the next multi-channel state to the PWM source (to avoid spikes on the output lines, see **Section 18.6**). This compare function of channel CC61 can be used as a phase delay from the position sensor input signals to the switching of the output signals, that is necessary if a sensorless back-EMF technique or Hall sensors are used. The compare value in channel CC62 can be used as a time-out trigger (interrupt), indicating that the actual motor speed is far below the desired destination value. An abnormal load change can be detected with this feature and PWM generation can be disabled.

18.8 Interrupt Handling

This section describes the interrupt handling of the CCU6 module.

18.8.1 Interrupt Structure

The HW interrupt event or the SW setting of the corresponding interrupt set bit (in register ISS) sets the event indication flags (in register IS) and can trigger the interrupt generation. The interrupt pulse is generated independently from the interrupt status flag in register IS (it is not necessary to clear the related status bit to be able to generate another interrupt). The interrupt flag can be cleared by SW by writing to the corresponding bit in register ISR.

If enabled by the related interrupt enable bit in register IEN, an interrupt pulse can be generated on one of the four service request outputs (SR0 to SR3) of the module. If more than one interrupt source is connected to the same interrupt node pointer (in register INP), the requests are logically OR-combined to one common service request output (see **Figure 164**).

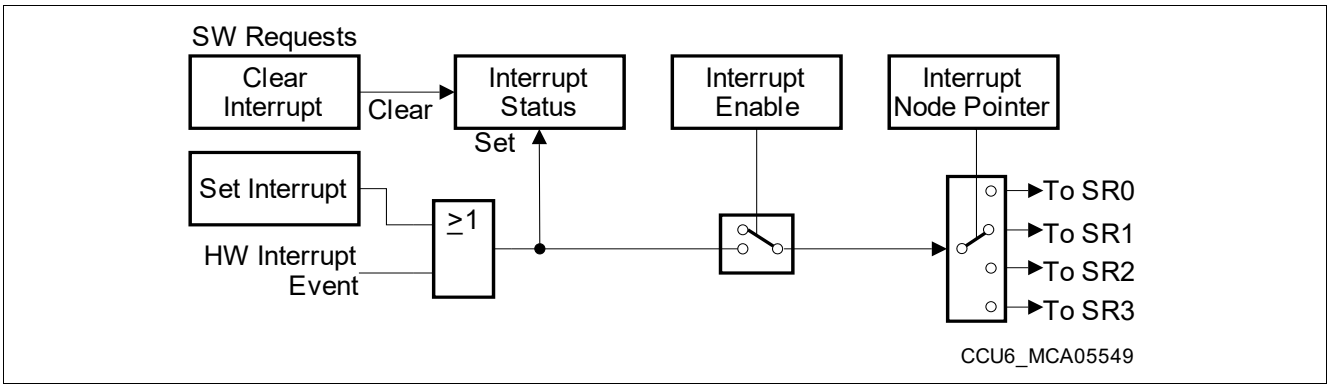


Figure 164 General Interrupt Structure

The available interrupt events in the CCU6 are shown in **Figure 165**.

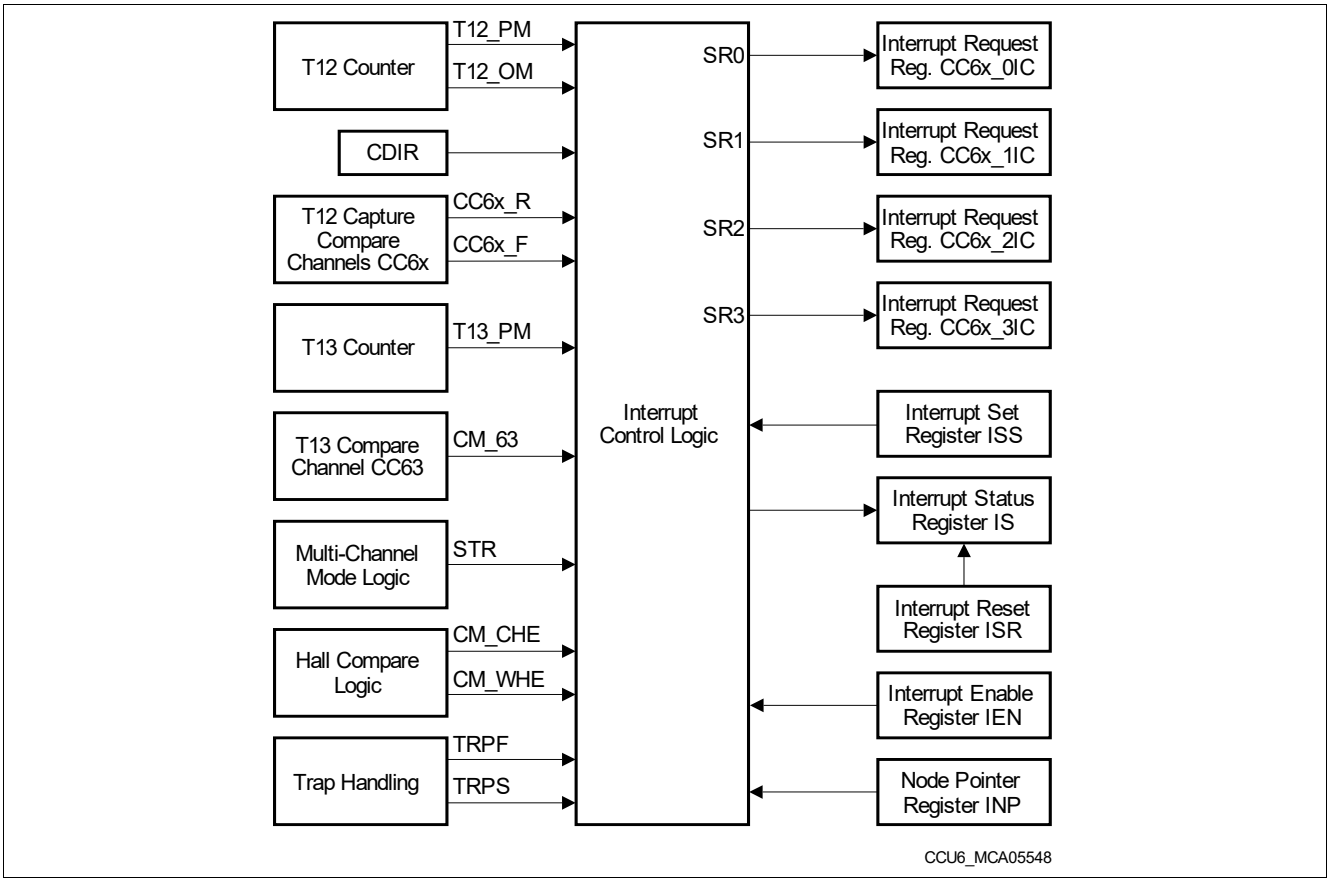


Figure 165 Interrupt Sources and Events

18.9 General Module Operation

This section provides information about the:

- Input selection (see **Section 18.9.1**)

18.9.1 Input Selection

Each CCU6 input signal can be selected from a vector of four or eight possible inputs by programming the port input select registers PISEL0 and PISEL2. This permits to adapt the pin functionality of the device to the application requirements.

The output pins for the module output signals are chosen in the ports.

Note: All functional inputs of the CCU6 are synchronized to f_{CC6} before they affect the module internal logic. The resulting delay of $2/f_{CC6}$ and for asynchronous signals an additional uncertainty of $1/f_{CC6}$ have to be taken into account for precise timing calculation. An edge of an input signal can only be correctly detected if the high phase and the low phase of the input signal are both longer than $1/f_{CC6}$.

18.10 CCU6 Register Description

All CCU6 kernel register names described in this section will be referenced in other parts of this specification with the module name prefix “CCU6_”.

Table 307 lists the CCU6 registers.

Note: If a hardware and a software request to modify a bit occur simultaneously, the software wins.

Table 306 Register Address Space

Module	Base Address	End Address	Note
CCU6	$4000C000_H$	$4000FFFF_H$	CCU6

Table 307 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
CCU6 Register Description, System Registers			
CCU6_PISEL0	Port Input Select Register 0	$6C_H$	0000_H
CCU6_PISEL2	Port Input Select Register 2	74_H	0000_H
CCU6 Register Description, Timer 12 – Related Registers			
CCU6_CC60SR	Capture/Compare Shadow Register for Channel CC60	14_H	0000_H
CCU6_CC61SR	Capture/Compare Shadow Register for Channel CC61	18_H	0000_H
CCU6_CC62SR	Capture/Compare Shadow Register for Channel CC62	$1C_H$	0000_H
CCU6_T12PR	Timer T12 Period Register	24_H	0000_H
CCU6_T12DTC	Dead-Time Control Register for Timer T12 Low	$2C_H$	0000_H
CCU6_CC60R	Capture/Compare Register for Channel CC60	34_H	0000_H
CCU6_CC61R	Capture/Compare Register for Channel CC61	38_H	0000_H
CCU6_CC62R	Capture/Compare Register for Channel CC62	$3C_H$	0000_H
CCU6_T12MSEL	T12 Capture/Compare Mode Select Register	40_H	0000_H
CCU6_T12	Timer T12 Counter Register	78_H	0000_H
CCU6 Register Description, Timer 13 – Related Registers			
CCU6_CC63R	Capture/Compare Register for Channel CC63	00_H	0000_H
CCU6_CC63SR	Capture/Compare Shadow Register for Channel CC63	20_H	0000_H
CCU6_T13PR	Timer T13 Period Register	28_H	0000_H
CCU6_T13	Timer T13 Counter Register	$7C_H$	0000_H
CCU6 Register Description, Capture/Compare Control Registers			
CCU6_TCTR4	Timer Control Register 4	04_H	0000_H
CCU6_CMPMODIF	Compare State Modification Register	10_H	0000_H
CCU6_TCTR0	Timer Control Register 0	30_H	0000_H
CCU6_TCTR2	Timer Control Register 2	58_H	0000_H

Table 307 Register Overview (cont'd)

Register Short Name	Register Long Name	Offset Address	Reset Value
CCU6_CMPSTAT	Compare State Register	80_H	0000_H
CCU6 Register Description, Global Modulation Control Registers			
CCU6_PSLR	Passive State Level Register	50_H	0000_H
CCU6_MODCTR	Modulation Control Register	$5C_H$	0000_H
CCU6_TRPCTR	Trap Control Register	60_H	0000_H
CCU6 Register Description, Multi-Channel Modulation Control Registers			
CCU6_MCMOUTS	Multi-Channel Mode Output Shadow Register	08_H	0000_H
CCU6_MCMCTR	Multi-Channel Mode Control Register	54_H	0000_H
CCU6_MCMOUT	Multi-Channel Mode Output Register	64_H	0000_H
CCU6 Register Description, Interrupt Control Registers			
CCU6_ISR	Capture/Compare Interrupt Status Reset Register	$0C_H$	0000_H
CCU6_IEN	Capture/Compare Interrupt Enable Register	44_H	0000_H
CCU6_INP	Capture/Compare Interrupt Node Pointer Register	48_H	3940_H
CCU6_ISS	Capture/Compare Interrupt Status Set Register	$4C_H$	0000_H
CCU6_IS	Capture/Compare Interrupt Status Register	68_H	0000_H

The registers are addressed wordwise.

18.10.1 System Registers

Registers PISEL0 and PISEL2 contain bit fields that select the actual input port/signal for the module inputs. This permits the adaptation of the pin functionality of the device to the application's requirements. The output pins are chosen according to the registers in the ports.

Port Input Select Register 0

CCU6_PISEL0	Offset	Reset Value
Port Input Select Register 0	**$6C_H$**	**see Table 308**

15	14	13	12	11	10	9	8
IST12HR		ISPOS2		ISPOS1		ISPOS0	
rw		rw		rw		rw	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
ISTRP		ISCC62		ISCC61		ISCC60	
rw		rw		rw		rw	

Field	Bits	Type	Description
IST12HR	15:14	rw	**Input Select for T12HR** This bit field defines the input signal used as T12HR input. 00_B **T12HRA**, Either signal T12HRA (if T12EXT = 0) or T12HRE (if T12EXT = 1) is selected. 01_B **T12HRB**, Either signal T12HRB (if T12EXT = 0) or T12HRF (if T12EXT = 1) is selected. 10_B **T12HRC**, Either signal T12HRC (if T12EXT = 0) or T12HRG (if T12EXT = 1) is selected. 11_B **T12HRD**, Either signal T12HRD (if T12EXT = 0) or T12HRH (if T12EXT = 1) is selected.
ISPOS2	13:12	rw	**Input Select for CCPOS2** This bit field defines the port pin that is used for the CCPOS2 input signal. 00_B **CCPOS2_0**, The input pin for CCPOS2_0. 01_B **CCPOS2_1**, The input pin for CCPOS2_1. 10_B **CCPOS2_2**, The input pin for CCPOS2_2. 11_B **Reserved**, Reserved
ISPOS1	11:10	rw	**Input Select for CCPOS1** This bit field defines the port pin that is used for the CCPOS1 input signal. 00_B **CCPOS1_0**, The input pin for CCPOS1_0. 01_B **CCPOS1_1**, The input pin for CCPOS1_1. 10_B **CCPOS1_2**, The input pin for CCPOS1_2. 11_B **Reserved**, Reserved
ISPOS0	9:8	rw	**Input Select for CCPOS0** This bit field defines the port pin that is used for the CCPOS0 input signal. 00_B **CCPOS0_0**, The input pin for CCPOS0_0. 01_B **CCPOS0_1**, The input pin for CCPOS0_1. 10_B **CCPOS0_2**, The input pin for CCPOS0_2. 11_B **Reserved**, Reserved
ISTRP	7:6	rw	**Input Select for CTRAP** This bit field defines the port pin that is used for the $\overline{\text{CTRAP}}$ input signal. 00_B **CTRAP_0**, The input pin for $\overline{\text{CTRAP\_0}}$. 01_B **CTRAP_1**, The input pin for $\overline{\text{CTRAP\_1}}$. 10_B **CTRAP_2**, The input pin for $\overline{\text{CTRAP\_2}}$. 11_B **DU1_UP_STS**, The output DU1_UP_STS of the Differential Measurement Unit is selected.
ISCC62	5:4	rw	**Input Select for CC62** This bit field defines the port pin that is used for the CC62 capture input signal. 00_B **CC62_0**, The input pin for CC62_0. 01_B **CC62_1**, The input pin for CC62_1. 10_B **Reserved**, Reserved 11_B **Reserved**, Reserved

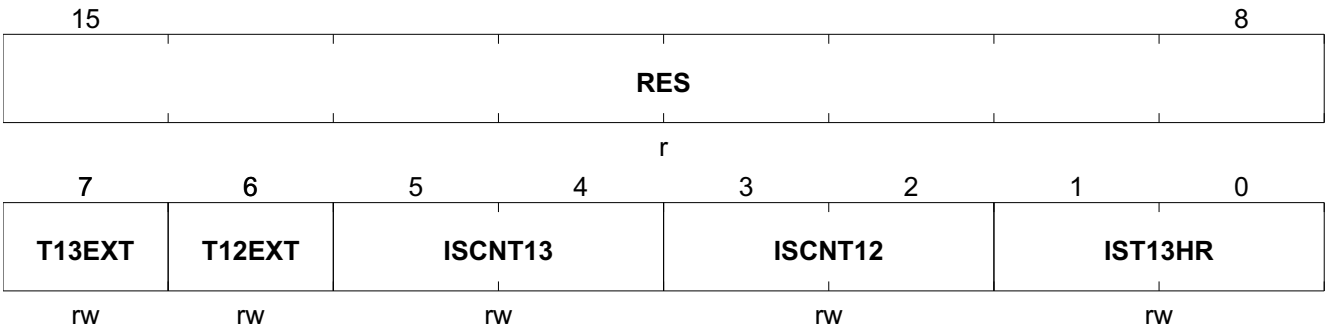
Field	Bits	Type	Description
ISCC61	3:2	rw	**Input Select for CC61** This bit field defines the port pin that is used for the CC61 capture input signal. 00_B **CC61_0**, The input pin for CC61_0. 01_B **CC61_1**, The input pin for CC61_1. 10_B **Reserved**, Reserved 11_B **Reserved**, Reserved
ISCC60	1:0	rw	**Input Select for CC60** This bit field defines the port pin that is used for the CC60 capture input signal. 00_B **CC60_0**, The input pin for CC60_0. 01_B **CC60_1**, The input pin for CC60_1. 10_B **Reserved**, Reserved 11_B **Reserved**, Reserved

Table 308 RESET of CCU6_PISEL0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Port Input Select Register 2

CCU6_PISEL2	Offset	Reset Value
Port Input Select Register 2	74_H	see Table 309



Field	Bits	Type	Description
RES	15:8	r	**Reserved**
T13EXT	7	rw	**Extension for T13HR Inputs** This bit extends the 2-bit field IST13HR. 0_B **T13HR_D_A**, One of the signals T13HR[D:A] is selected. 1_B **T13HR_H_E**, One of the signals T13HR[H:E] is selected.
T12EXT	6	rw	**Extension for T12HR Inputs** This bit extends the 2-bit field IST12HR. 0_B **T12HR_D_A**, One of the signals T12HR[D:A] is selected. 1_B **T12HR_H_E**, One of the signals T12HR[H:E] is selected.

Field	Bits	Type	Description
ISCNT13	5:4	rw	**Input Select for T13 Counting Input** This bit field defines the input event leading to a counting action of T13. 00_B **T13 prescaler**, The T13 prescaler generates the counting events. Bit TCTR4.T13CNT is not taken into account. 01_B **TCTR4.T13CNT**, Bit TCTR4.T13CNT written with 1 is a counting event. The T13 prescaler is not taken into account. 10_B **Rising edge**, The timer T13 is counting each rising edge detected in the selected T13HR signal. 11_B **Falling Edge**, The timer T13 is counting each falling edge detected in the selected T13HR signal.
ISCNT12	3:2	rw	**Input Select for T12 Counting Input** This bit field defines the input event leading to a counting action of T12. 00_B **T12 prescaler**, The T12 prescaler generates the counting events. Bit TCTR4.T12CNT is not taken into account. 01_B **TCTR4.T12CNT**, Bit TCTR4.T12CNT written with 1 is a counting event. The T12 prescaler is not taken into account. 10_B **Rising edge**, The timer T12 is counting each rising edge detected in the selected T12HR signal. 11_B **Falling edge**, The timer T12 is counting each falling edge detected in the selected T12HR signal.
IST13HR	1:0	rw	**Input Select for T13HR** This bit field defines the input signal used as T13HR input. 00_B **T13HRA**, Either signal T13HRA (if T13EXT = 0) or T13HRE (if T13EXT = 1) is selected. 01_B **T13HRB**, Either signal T13HRB (if T13EXT = 0) or T13HRF (if T13EXT = 1) is selected. 10_B **T13HRC**, Either signal T13HRC (if T13EXT = 0) or T13HRG (if T13EXT = 1) is selected. 11_B **T13HRD**, Either signal T13HRD (if T13EXT = 0) or T13HRH (if T13EXT = 1) is selected.

Table 309 RESET of CCU6_PISEL2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

18.10.2 Timer 12 – Related Registers

The generation of the patterns for a 3-channel PWM is based on timer T12. The registers related to timer T12 can be concurrently updated (with well-defined conditions) in order to ensure consistency of the three PWM channels.

Timer T12 supports capture and compare modes, which can be independently selected for the three channels CC60, CC61, and CC62.

Register T12MSEL contains control bits to select the capture/compare functionality of the three channels of timer T12. **Table 310**, **Table 311** and **Table 312** define and elaborate some of the capture/compare modes selectable. Refer to the following register description for the selection.

Table 310 Double-Register Capture Modes

	Description
0100_B	The contents of T12 are stored in CC6nR after a rising edge and in CC6nSR after a falling edge on the input pin CC6n.
0101_B	The value stored in CC6nSR is copied to CC6nR after a rising edge on the input pin CC6n. The actual timer value of T12 is simultaneously stored in the shadow register CC6nSR. This feature is useful for time measurements between consecutive rising edges on pins CC6n. COUT6n is I/O.
0110_B	The value stored in CC6nSR is copied to CC6nR after a falling edge on the input pin CC6n. The actual timer value of T12 is simultaneously stored in the shadow register CC6nSR. This feature is useful for time measurements between consecutive falling edges on pins CC6n. COUT6n is I/O.
0111_B	The value stored in CC6nSR is copied to CC6nR after any edge on the input pin CC6n. The actual timer value of T12 is simultaneously stored in the shadow register CC6nSR. This feature is useful for time measurements between consecutive edges on pins CC6n. COUT6n is I/O.

Table 311 Combined T12 Modes

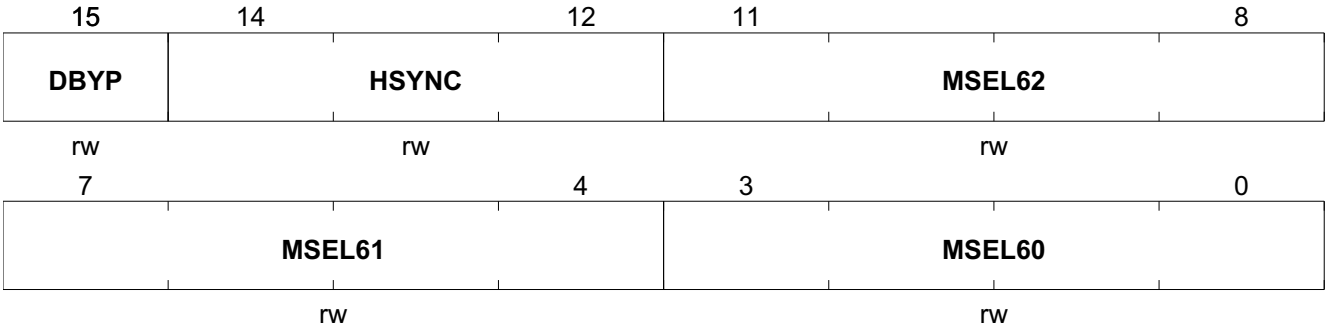
	Description
1000_B	Hall Sensor mode: Capture mode for channel 0, compare mode for channels 1 and 2. The contents of T12 are captured into CC60 at a valid hall event (which is a reference to the actual speed). CC61 can be used for a phase delay function between hall event and output switching. CC62 can act as a time-out trigger if the expected hall event comes too late. The value 1000_B must be programmed to MSEL0, MSEL1 and MSEL2 if the hall signals are used. In this mode, the contents of timer T12 are captured in CC60 and T12 is reset after the detection of a valid hall event. In order to avoid noise effects, the dead-time counter channel 0 is started after an edge has been detected at the hall inputs. On reaching the value of 000001_B, the hall inputs are sampled and the pattern comparison is done.
1001_B	Hysteresis-like control mode with dead-time generation: The negative edge of the CCPOSx input signal is used to reset bit CC6nST. As a result, the output signals can be switched to passive state immediately and switch back to active state (with dead-time) if the CCPOSx is high and the bit CC6nST is set by a compare event.

Table 312 Multi-Input Capture Modes

	Description
1010_B	The timer value of T12 is stored in CC6nR after a rising edge at the input pin CC6n. The timer value of T12 is stored in CC6nSR after a falling edge at the input pin CCPOSx.
1011_B	The timer value of T12 is stored in CC6nR after a falling edge at the input pin CC6n. The timer value of T12 is stored in CC6nSR after a rising edge at the input pin CCPOSx.
1100_B	The timer value of T12 is stored in CC6nR after a rising edge at the input pin CC6n. The timer value of T12 is stored in CC6nSR after a rising edge at the input pin CCPOSx.
1101_B	The timer value of T12 is stored in CC6nR after a falling edge at the input pin CC6n. The timer value of T12 is stored in CC6nSR after a falling edge at the input pin CCPOSx.
1110_B	The timer value of T12 is stored in CC6nR after any edge at the input pin CC6n. The timer value of T12 is stored in CC6nSR after any edge at the input pin CCPOSx.
1111_B	reserved (no capture or compare action)

T12 Capture/Compare Mode Select Register

CCU6_T12MSEL	Offset	Reset Value
T12 Capture/Compare Mode Select Register	40_H	see Table 313



Field	Bits	Type	Description
DBYP	15	rw	**Delay Bypass** Bit DBYP defines if the source signal for the sampling of the Hall input pattern (selected by HSYNC) uses the dead-time counter DTC0 of timer T12 as additional delay or if the delay is bypassed. 0_B **Not active**, The delay bypass is not active. The dead-time counter DTC0 is generating a delay after the source signal becomes active. 1_B **Active**, The delay bypass is active. The dead-time counter DTC0 is not used by the sampling of the Hall pattern.
HSYNC	14:12	rw	**Hall Synchronization** Bit field HSYNC defines the source for the sampling of the Hall input pattern and the comparison to the current and the expected Hall pattern bit fields. In all modes, a trigger by software by writing a 1 to bit SWHC is possible. 000_B **Any**, Any edge at one of the inputs CCPOSx (x = 0, 1, 2) triggers the sampling. 001_B **T13 compare-match**, A T13 compare-match triggers the sampling. 010_B **T13 period-match**, A T13 period-match triggers the sampling. 011_B **Hall**, The Hall sampling triggered by hardware sources is switched off. 100_B **T12 period-match**, A T12 period-match (while counting up) triggers the sampling. 101_B **T12 one-match**, A T12 one-match (while counting down) triggers the sampling. 110_B **T12 compare-match UP**, A T12 compare-match of channel 0 (while counting up) triggers the sampling. 111_B **T12 compare-match DOWN**, A T12 compare-match of channel 0 (while counting down) triggers the sampling.

Field	Bits	Type	Description
MSEL62	11:8	rw	**Capture/Compare Mode Selection** These bit fields select the operating mode of the three timer T12 capture/compare channels. Each channel (n = 0, 1, 2) can be programmed individually either for compare or capture operation according to: 0000_B**Compare outputs disabled**, Compare outputs disabled, pins CC6n and COUT6n can be used for I/O. No capture action. 0001_B**Pin CC6n, pin COUT6n**, Compare output on pin CC6n, pin COUT6n can be used for I/O. No capture action. 0010_B**Pin COUT6n, Pin CC6n**, Compare output on pin COUT6n, pin CC6n can be used for I/O. No capture action. 0011_B**Pins COUT6n and CC6n**, Compare output on pins COUT6n and CC6n. $01XX_B$**Double-Register Capture modes**, see **Table 310**. 1000_B**Hall Sensor mode**, see **Table 311**. In order to enable the hall edge detection, all three MSEL6x must be programmed to Hall Sensor mode. 1001_B**Hysteresis-like mode**, see **Table 311**. $101X_B$**Multi-Input Capture modes**, see **Table 312**. $11XX_B$**Multi-Input Capture modes**, see **Table 312**.
MSEL61	7:4	rw	**Capture/Compare Mode Selection** These bit fields select the operating mode of the three timer T12 capture/compare channels. Each channel (n = 0, 1, 2) can be programmed individually either for compare or capture operation according to: 0000_B**Compare outputs disabled**, Compare outputs disabled, pins CC6n and COUT6n can be used for I/O. No capture action. 0001_B**Pin CC6n, pin COUT6n**, Compare output on pin CC6n, pin COUT6n can be used for I/O. No capture action. 0010_B**Pin COUT6n, Pin CC6n**, Compare output on pin COUT6n, pin CC6n can be used for I/O. No capture action. 0011_B**Pins COUT6n and CC6n**, Compare output on pins COUT6n and CC6n. $01XX_B$**Double-Register Capture modes**, see **Table 310**. 1000_B**Hysteresis-like mode**, see **Table 311**. In order to enable the hall edge detection, all three MSEL6x must be programmed to Hall Sensor mode. 1001_B**Hysteresis-like mode**, see **Table 311**. $101X_B$**Multi-Input Capture modes**, see **Table 312**. $11XX_B$**Multi-Input Capture modes**, see **Table 312**.

Field	Bits	Type	Description
MSEL60	3:0	rw	**Capture/Compare Mode Selection** These bit fields select the operating mode of the three timer T12 capture/compare channels. Each channel (n = 0, 1, 2) can be programmed individually either for compare or capture operation according to: 0000_B **Compare outputs disabled**, Compare outputs disabled, pins CC6n and COUT6n can be used for I/O. No capture action. 0001_B **Pin CC6n, pin COUT6n**, Compare output on pin CC6n, pin COUT6n can be used for I/O. No capture action. 0010_B **Pin COUT6n, Pin CC6n**, Compare output on pin COUT6n, pin CC6n can be used for I/O. No capture action. 0011_B **Pins COUT6n and CC6n**, Compare output on pins COUT6n and CC6n. $01XX_B$ **Double-Register Capture modes**, see **Table 310**. 1000_B **Hysteresis-like mode**, see **Table 311**. In order to enable the hall edge detection, all three MSEL6x must be programmed to Hall Sensor mode. 1001_B **Hysteresis-like mode**, see **Table 311**. $101X_B$ **Multi-Input Capture modes**, see **Table 312**. $11XX_B$ **Multi-Input Capture modes**, see **Table 312**.

Table 313 RESET of CCU6_T12MSEL

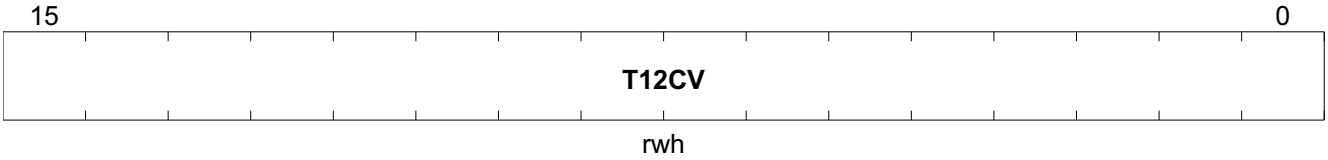
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Timer T12 Counter Register

Register T12 represents the counting value of timer T12. It can only be written while the timer T12 is stopped. Write actions while T12 is running are not taken into account. Register T12 can always be read by software.

In edge-aligned mode, T12 only counts up, whereas in center-aligned mode, T12 can count up and down.

CCU6_T12	Offset	Reset Value
Timer T12 Counter Register	**78_H**	**see Table 314**



Field	Bits	Type	Description
T12CV	15:0	rwh	**Timer T12 Counter Value** This register represents the lower 8-bit counter value of timer T12.

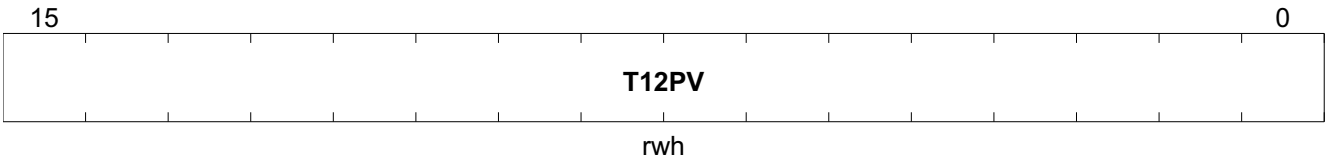
Table 314 RESET of CCU6_T12

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Timer T12 Period Register

Register T12PR contains the period value for timer T12. The period value is compared to the actual counter value of T12 and the resulting counter actions depend on the defined counting rules. This register has a shadow register and the shadow transfer is controlled by bit STE12. A read action by software delivers the value which is currently used for the compare action, whereas the write action targets a shadow register. The shadow register structure allows a concurrent update of all T12-related values.

CCU6_T12PR	Offset	Reset Value
Timer T12 Period Register	24_H	see Table 315



Field	Bits	Type	Description
T12PV	15:0	rwh	**T12 Period Value** The value T12PV defines the counter value for T12, which leads to a period-match. On reaching this value, the timer T12 is set to zero (edge-aligned mode) or changes its count direction to down counting (center-aligned mode).

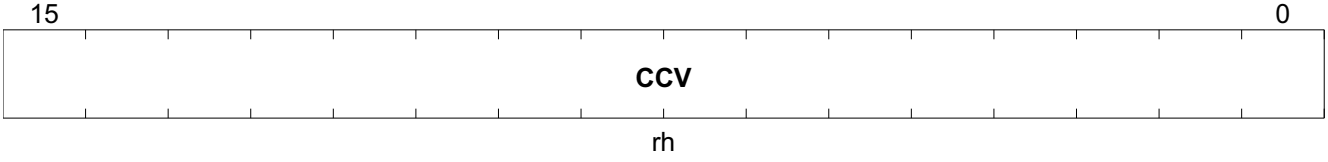
Table 315 RESET of CCU6_T12PR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Capture/Compare Register for Channel CC60

In compare mode, the registers CC60R is the actual compare registers for T12. The values stored in CC60R are compared (all three channels in parallel) to the counter value of T12. In capture mode, the current value of the T12 counter register is captured by registers CC60R if the corresponding capture event is detected.

CCU6_CC60R	Offset	Reset Value
Capture/Compare Register for Channel CC60	34_H	see Table 318



Field	Bits	Type	Description
CCV	15:0	rh	**Channel 0 Capture/Compare Value** In compare mode, the bit fields CCV contain the values that are compared to the T12 counter value. In capture mode, the captured value of T12 can be read from these registers.

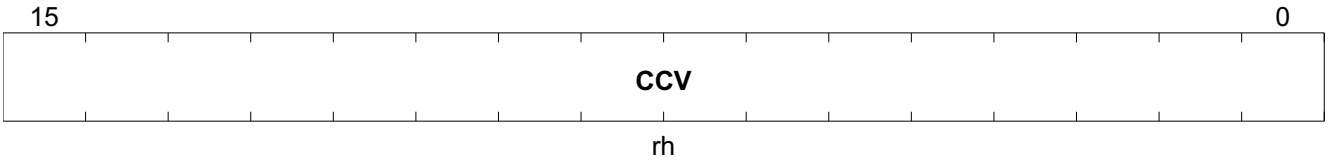
Table 316 RESET of CCU6_CC60R

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Capture/Compare Register for Channel CC61

In compare mode, the registers CC61R is the actual compare registers for T12. The values stored in CC61R are compared (all three channels in parallel) to the counter value of T12. In capture mode, the current value of the T12 counter register is captured by registers CC61R if the corresponding capture event is detected.

CCU6_CC61R	Offset	Reset Value
Capture/Compare Register for Channel CC61	**38_H**	**see Table 317**



Field	Bits	Type	Description
CCV	15:0	rh	**Channel 1 Capture/Compare Value** In compare mode, the bit fields CCV contain the values that are compared to the T12 counter value. In capture mode, the captured value of T12 can be read from these registers.

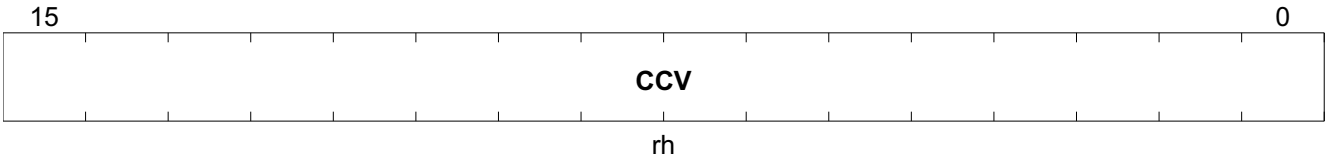
Table 317 RESET of CCU6_CC61R

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Capture/Compare Register for Channel CC62

In compare mode, the registers CC62R is the actual compare registers for T12. The values stored in CC62R are compared (all three channels in parallel) to the counter value of T12. In capture mode, the current value of the T12 counter register is captured by registers CC62R if the corresponding capture event is detected.

CCU6_CC62R	Offset	Reset Value
Capture/Compare Register for Channel CC62	**$3C_H$**	**see Table 318**



Field	Bits	Type	Description
CCV	15:0	rh	**Channel 2 Capture/Compare Value** In compare mode, the bit fields CCV contain the values that are compared to the T12 counter value. In capture mode, the captured value of T12 can be read from these registers.

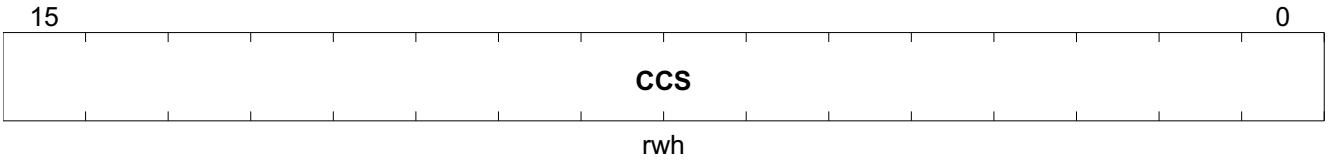
Table 318 RESET of CCU6_CC62R

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Capture/Compare Shadow Register for Channel CC60

The registers CC60R can only be read by software, the modification of the value is done by a shadow register transfer from register CC60SR. The corresponding shadow registers CC60SR can be read and written by software. In capture mode, the value of the T12 counter register can also be captured by registers CC60SR if the selected capture event is detected (depending on the selected mode).

CCU6_CC60SR	Offset	Reset Value
Capture/Compare Shadow Register for Channel CC60	14_H	see Table 319



Field	Bits	Type	Description
CCS	15:0	rwh	**Shadow Register for Channel 0 Capture/Compare Value** In compare mode, the contents of bit field CCS are transferred to the bit field CCV for the corresponding channel during a shadow transfer. In capture mode, the captured value of T12 can be read from these registers.

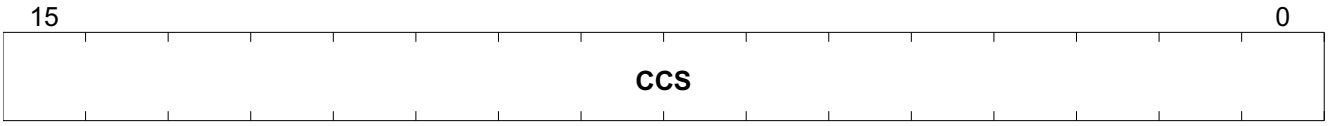
Table 319 RESET of CCU6_CC60SR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Capture/Compare Shadow Register for Channel CC61

The registers CC61R can only be read by software, the modification of the value is done by a shadow register transfer from register CC61SR. The corresponding shadow registers CC61SR can be read and written by software. In capture mode, the value of the T12 counter register can also be captured by registers CC61SR if the selected capture event is detected (depending on the selected mode).

CCU6_CC61SR	Offset	Reset Value
Capture/Compare Shadow Register for Channel CC61	18_H	see Table 320



Field	Bits	Type	Description
CCS	15:0	rwh	**Shadow Register for Channel 1 Capture/Compare Value** In compare mode, the contents of bit field CCS are transferred to the bit field CCV for the corresponding channel during a shadow transfer. In capture mode, the captured value of T12 can be read from these registers.

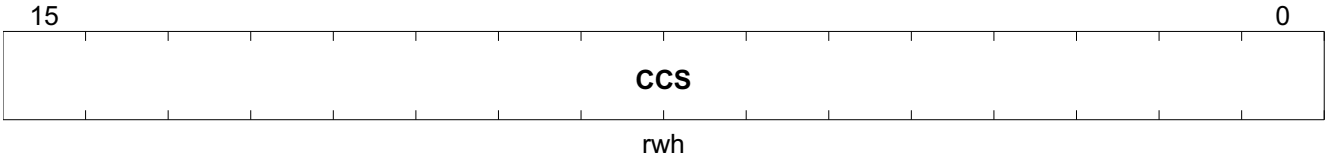
Table 320 RESET of CCU6_CC61SR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Capture/Compare Shadow Register for Channel CC62

The registers CC62R can only be read by software, the modification of the value is done by a shadow register transfer from register CC62SR. The corresponding shadow registers CC62SR can be read and written by software. In capture mode, the value of the T12 counter register can also be captured by registers CC62SR if the selected capture event is detected (depending on the selected mode).

CCU6_CC62SR	Offset	Reset Value
Capture/Compare Shadow Register for Channel CC62	$1C_H$	see Table 321



Field	Bits	Type	Description
CCS	15:0	rwh	**Shadow Register for Channel 2 Capture/Compare Value** In compare mode, the contents of bit field CCS are transferred to the bit field CCV for the corresponding channel during a shadow transfer. In capture mode, the captured value of T12 can be read from these registers.

Table 321 RESET of CCU6_CC62SR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Dead-Time Control Register for Timer T12 Low

Register T12DTC controls the dead-time generation for the timer T12 compare channels. Each channel can be independently enabled/disabled for dead-time generation. If enabled, the transition from passive state to active state is delayed by the value defined by bit field DTM. The dead-time counter can only be reloaded while it is zero.

The dead time counters are clocked with the same frequency as T12. This structure allows symmetrical dead-time generation in center-aligned and in edge-aligned PWM mode. A duty cycle of 50% leads to CC6x, COUT6x switched on for: 0.5 * period - dead time.

Note: The dead-time counters are not reset by bit T12RES, but by bit DTRES.

CCU6_T12DTC	Offset	Reset Value
Dead-Time Control Register for Timer T12 Low	$2C_H$	see Table 322

15	14	13	12	11	10	9	8	7	0
RES	DTR2	DTR1	DTR0	RES	DTE2	DTE1	DTE0	DTM	
r	rh	rh	rh	r	rw	rw	rw	rw	

Field	Bits	Type	Description
RES	15	r	**Reserved** Returns 0 if read; should be written with 0.
DTR2	14	rh	**Dead-Time Run Indication Bit 2** Bit DTR2 indicates the status of the dead-time generation for compare channel 2 of timer T12. 0_B **Zero**, The value of the corresponding dead-time counter channel is 0. 1_B **Not Zero**, The value of the corresponding dead-time counter channel is not 0.

Field	Bits	Type	Description
DTR1	13	rh	**Dead-Time Run Indication Bit 1** Bit DTR1 indicates the status of the dead-time generation for compare channel 1 of timer T12. 0_B **Zero**, The value of the corresponding dead-time counter channel is 0. 1_B **Not Zero**, The value of the corresponding dead-time counter channel is not 0.
DTR0	12	rh	**Dead-Time Run Indication Bit 0** Bit DTR0 indicate the status of the dead-time generation for compare channel 0 of timer T12. 0_B **Zero**, The value of the corresponding dead-time counter channel is 0. 1_B **Not Zero**, The value of the corresponding dead-time counter channel is not 0.
RES	11	r	**Reserved**
DTE2	10	rw	**Dead-Time Enable Bit 2** Bit DTE2 enables and disables the dead-time generation for compare channel 2 of timer T12. 0_B **Disabled**, Dead-time generation is disabled. The corresponding outputs switch from the passive state to the active state (according to the actual compare status) without any delay. 1_B **Enabled**, Dead-time generation is enabled. The corresponding outputs switch from the passive state to the active state (according to the compare status) with the delay programmed in bit field DTM.
DTE1	9	rw	**Dead-Time Enable Bit 1** Bit DTE1 enables and disables the dead-time generation for compare channel 1 of timer T12. 0_B **Disabled**, Dead-time generation is disabled. The corresponding outputs switch from the passive state to the active state (according to the actual compare status) without any delay. 1_B **Enabled**, Dead-time generation is enabled. The corresponding outputs switch from the passive state to the active state (according to the compare status) with the delay programmed in bit field DTM.

Field	Bits	Type	Description
DTE0	8	rw	**Dead-Time Enable Bit 0** Bit DTE0 enables and disables the dead-time generation for compare channel 0 of timer T12. 0_B **Disabled**, Dead-time generation is disabled. The corresponding outputs switch from the passive state to the active state (according to the actual compare status) without any delay. 1_B **Enabled**, Dead-time generation is enabled. The corresponding outputs switch from the passive state to the active state (according to the compare status) with the delay programmed in bit field DTM.
DTM	7:0	rw	**Dead-Time** Bit field DTM determines the programmable delay between switching from the passive state to the active state of the selected outputs. The switching from the active state to the passive state is not delayed.

Table 322 RESET of CCU6_T12DTC

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

18.10.3 Timer 13 – Related Registers

The generation of the patterns for a single channel pulse width modulation (PWM) is based on timer T13. The registers related to timer T13 can be concurrently updated (with well-defined conditions) in order to ensure consistency of the PWM signal. T13 can be synchronized to several timer T12 events.

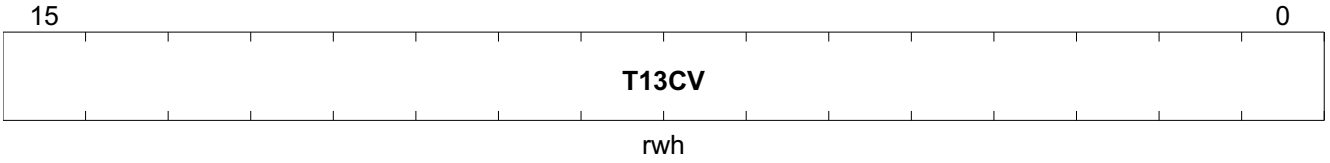
Timer T13 supports only compare mode on its compare channel CC63.

Register T13 represents the counting value of timer T13. It can only be written while the timer T13 is stopped. Write actions while T13 is running are not taken into account. Register T13 can always be read by software.

Timer T13 supports only edge-aligned mode (counting up).

Timer T13 Counter Register

CCU6_T13	Offset	Reset Value
Timer T13 Counter Register	**$7C_H$**	**see Table 323**



Field	Bits	Type	Description
T13CV	15:0	rwh	**Timer T13 Counter Value** This register represents the lower 8-bit counter value of timer T13.

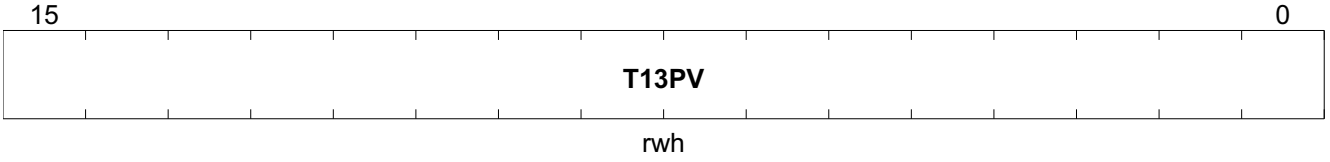
Table 323 RESET of CCU6_T13

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Timer T13 Period Register

Register T13PR contains the period value for timer T13. The period value is compared to the actual counter value of T13 and the resulting counter actions depend on the defined counting rules. This register has a shadow register and the shadow transfer is controlled by bit STE13. A read action by software delivers the value which is currently used for the compare action, whereas the write action targets a shadow register. The shadow register structure allows a concurrent update of all T13-related values.

CCU6_T13PR	Offset	Reset Value
Timer T13 Period Register	**28_H**	**see Table 324**



Field	Bits	Type	Description
T13PV	15:0	rwh	**T13 Period Value** The value T13PV defines the counter value for T13, which leads to a period-match. On reaching this value, the timer T13 is set to zero.

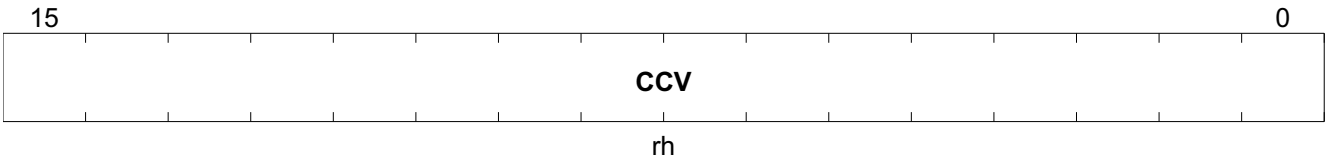
Table 324 RESET of CCU6_T13PR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Capture/Compare Register for Channel CC63

Register CC63R is the actual compare register for T13. The value stored in CC63R is compared to the counter value of T13. The State Bit CC63ST is located in register CMPSTAT.

CCU6_CC63R	Offset	Reset Value
Capture/Compare Register for Channel CC63	00_H	see Table 325



Field	Bits	Type	Description
CCV	15:0	rh	**Channel CC63 Compare Value Low Byte** The bit field CCV contains the value that is compared to the T13 counter value.

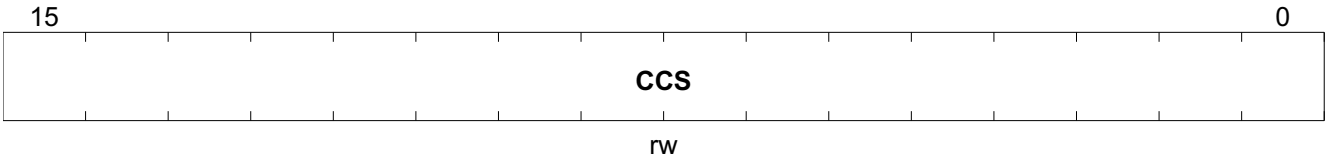
Table 325 RESET of CCU6_CC63R

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Capture/Compare Shadow Register for Channel CC63

The register CC63R can only be read by software and the modification of the value is done by a shadow register transfer from register CC63SR. The corresponding shadow register CC63SR can be read and written by software.

CCU6_CC63SR	Offset	Reset Value
Capture/Compare Shadow Register for Channel CC63	20_H	see Table 326



Field	Bits	Type	Description
CCS	15:0	rw	**Shadow Register for Channel CC63 Compare Value** The contents of bit field CCS are transferred to the bit field CCV during a shadow transfer.

Table 326 RESET of CCU6_CC63SR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

18.10.4 Capture/Compare Control Registers

Compare State Register

The Compare State Register CMPSTAT contains status bits monitoring the current capture and compare state, and control bits defining the active/passive state of the compare channels.

CCU6_CMPSTAT	Offset	Reset Value
Compare State Register	80_H	see Table 327

15	14	13	12	11	10	9	8
T13IM	COUT63PS	COUT62PS	CC62PS	COUT61PS	CC61PS	COUT60PS	CC60PS
rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh
7	6	5	4	3	2	1	0
RES	CC63ST	CCPOS2	CCPOS1	CCPOS0	CC62ST	CC61ST	CC60ST
r	rh	rh	rh	rh	rh	rh	rh

Field	Bits	Type	Description
T13IM	15	rwh	**T13 Inverted Modulation** Bit T13IM inverts the T13 signal for the modulation of the CC6x and COUT6x (x = 0, 1, 2) signals. This bit has a shadow bit and is updated in parallel to the compare and period registers of T13. A read action targets the actually used values, whereas a write action targets the shadow bit. 0_B **Not inverted**, T13 output is not inverted. 1_B **Inverted**, T13 output is inverted for further modulation.
COUT63PS	14	rwh	**Passive State Select for Compare Outputs** Bits COUT6xPS select the state of the corresponding compare channel, which is considered to be the passive state. During the passive state, the passive level (defined in register PSLR) is driven by the output pin. Bits COUT6xPS (x = 0, 1, 2) are related to T12, bit COUT63PS is related to T13. These bits have shadow bits and are updated in parallel to the capture/compare registers of T12 and T13, respectively. A read action targets the actually used values, whereas a write action targets the shadow bits. In capture mode, these bits are not used. 0_B **Zero**, The corresponding compare output drives passive level while CC6xST is 0. 1_B **One**, The corresponding compare output drives passive level while CC6xST is 1.

Field	Bits	Type	Description
COUT62PS	13	rwh	**Passive State Select for Compare Outputs** COUT6xPS select the state of the corresponding compare channel, which is considered to be the passive state. During the passive state, the passive level (defined in register PSLR) is driven by the output pin. Bits COUT6xPS (x = 0, 1, 2) are related to T12, bit COUT63PS is related to T13. These bits have shadow bits and are updated in parallel to the capture/compare registers of T12 and T13, respectively. A read action targets the actually used values, whereas a write action targets the shadow bits. In capture mode, these bits are not used. 0_B **Zero**, The corresponding compare output drives passive level while CC6xST is 0. 1_B **One**, The corresponding compare output drives passive level while CC6xST is 1.
CC62PS	12	rwh	**Passive State Select for Compare Outputs** Bits CC6xPS select the state of the corresponding compare channel, which is considered to be the passive state. During the passive state, the passive level (defined in register PSLR) is driven by the output pin. Bits CC6xPS are related to T12, bit COUT63PS is related to T13. These bits have shadow bits and are updated in parallel to the capture/compare registers of T12 and T13, respectively. A read action targets the actually used values, whereas a write action targets the shadow bits. In capture mode, these bits are not used. 0_B **Zero**, The corresponding compare output drives passive level while CC6xST is 0. 1_B **One**, The corresponding compare output drives passive level while CC6xST is 1.
COUT61PS	11	rwh	**Passive State Select for Compare Outputs** Bits COUT6xPS select the state of the corresponding compare channel, which is considered to be the passive state. During the passive state, the passive level (defined in register PSLR) is driven by the output pin. Bits COUT6xPS (x = 0, 1, 2) are related to T12, bit COUT63PS is related to T13. These bits have shadow bits and are updated in parallel to the capture/compare registers of T12 and T13, respectively. A read action targets the actually used values, whereas a write action targets the shadow bits. In capture mode, these bits are not used. 0_B **Zero**, The corresponding compare output drives passive level while CC6xST is 0. 1_B **One**, The corresponding compare output drives passive level while CC6xST is 1.

Field	Bits	Type	Description
CC61PS	10	rwh	**Passive State Select for Compare Outputs** Bits CC6xPS select the state of the corresponding compare channel, which is considered to be the passive state. During the passive state, the passive level (defined in register PSLR) is driven by the output pin. Bits CC6xPS are related to T12, bit COUT63PS is related to T13. These bits have shadow bits and are updated in parallel to the capture/compare registers of T12 and T13, respectively. A read action targets the actually used values, whereas a write action targets the shadow bits. In capture mode, these bits are not used. 0_B **Zero**, The corresponding compare output drives passive level while CC6xST is 0. 1_B **One**, The corresponding compare output drives passive level while CC6xST is 1.
COUT60PS	9	rwh	**Passive State Select for Compare Outputs** Bits COUT6xPS select the state of the corresponding compare channel, which is considered to be the passive state. During the passive state, the passive level (defined in register PSLR) is driven by the output pin. Bits COUT6xPS (x = 0, 1, 2) are related to T12, bit COUT63PS is related to T13. These bits have shadow bits and are updated in parallel to the capture/compare registers of T12 and T13, respectively. A read action targets the actually used values, whereas a write action targets the shadow bits. In capture mode, these bits are not used. 0_B **Zero**, The corresponding compare output drives passive level while CC6xST is 0. 1_B **One**, The corresponding compare output drives passive level while CC6xST is 1.
CC60PS	8	rwh	**Passive State Select for Compare Outputs** Bits CC6xPS select the state of the corresponding compare channel, which is considered to be the passive state. During the passive state, the passive level (defined in register PSLR) is driven by the output pin. Bits CC6xPS are related to T12, bit COUT63PS is related to T13. These bits have shadow bits and are updated in parallel to the capture/compare registers of T12 and T13, respectively. A read action targets the actually used values, whereas a write action targets the shadow bits. In capture mode, these bits are not used. 0_B **Zero**, The corresponding compare output drives passive level while CC6xST is 0. 1_B **One**, The corresponding compare output drives passive level while CC6xST is 1.
RES	7	r	**Reserved** Returns 0 if read.

Field	Bits	Type	Description
CC63ST	6	rh	**Capture/Compare State Bits** Bit CC63ST is related to T13. These bits are set and reset according to the T12 and T13 switching rules. 0_B **Less**, In compare mode, the timer count is less than the compare value. In capture mode, the selected edge has not yet been detected since the bit has been reset by software the last time. 1_B **Greater**, In compare mode, the counter value is greater than or equal to the compare value. In capture mode, the selected edge has been detected.
CCPOS2	5	rh	**Sampled Hall Pattern Bit 2** Bit CCPOS2 indicate the value of the input Hall pattern that has been compared to the current and expected value. The value is sampled when the event hcrdy (Hall compare ready) occurs. 0_B **Zero**, The input CCPOS2 has been sampled as 0. 1_B **One**, The input CCPOS2 has been sampled as 1.
CCPOS1	4	rh	**Sampled Hall Pattern Bit 1** Bit CCPOS1 indicate the value of the input Hall pattern that has been compared to the current and expected value. The value is sampled when the event hcrdy (Hall compare ready) occurs. 0_B **Zero**, The input CCPOS1 has been sampled as 0. 1_B **One**, The input CCPOS1 has been sampled as 1.
CCPOS0	3	rh	**Sampled Hall Pattern Bit 0** Bit CCPOS0 indicate the value of the input Hall pattern that has been compared to the current and expected value. The value is sampled when the event hcrdy (Hall compare ready) occurs. 0_B **Zero**, The input CCPOS0 has been sampled as 0. 1_B **One**, The input CCPOS0 has been sampled as 1.
CC62ST	2	rh	**Capture/Compare State Bits** Bits CC6xST monitor the state of the capture/compare channels. Bits CC6xST are related to T12; bit CC63ST is related to T13. These bits are set and reset according to the T12 and T13 switching rules. 0_B **Less**, In compare mode, the timer count is less than the compare value. In capture mode, the selected edge has not yet been detected since the bit has been reset by software the last time. 1_B **Greater**, In compare mode, the counter value is greater than or equal to the compare value. In capture mode, the selected edge has been detected.

Field	Bits	Type	Description
CC61ST	1	rh	**Capture/Compare State Bits** Bits CC6xST monitor the state of the capture/compare channels. Bits CC6xST are related to T12; bit CC63ST is related to T13. These bits are set and reset according to the T12 and T13 switching rules. 0_B **Less**, In compare mode, the timer count is less than the compare value. In capture mode, the selected edge has not yet been detected since the bit has been reset by software the last time. 1_B **Greater**, In compare mode, the counter value is greater than or equal to the compare value. In capture mode, the selected edge has been detected.
CC60ST	0	rh	**Capture/Compare State Bits** Bits CC6xST monitor the state of the capture/compare channels. Bits CC6xST are related to T12; bit CC63ST is related to T13. These bits are set and reset according to the T12 and T13 switching rules. 0_B **Less**, In compare mode, the timer count is less than the compare value. In capture mode, the selected edge has not yet been detected since the bit has been reset by software the last time. 1_B **Greater**, In compare mode, the counter value is greater than or equal to the compare value. In capture mode, the selected edge has been detected.

Table 327 RESET of CCU6_CMPSTAT

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Compare State Modification Register

The Compare Status Modification Register CMPMODIF provides software-control (independent set and clear conditions) for the channel state bits CC6xST. This feature enables the user to individually change the status of the output lines by software, for example when the corresponding compare timer is stopped.

CCU6_CMPMODIF	Offset	Reset Value
Compare State Modification Register	10_H	see Table 328

15	14	13	11	10	9	8
RES	MCC63R	RES		MCC62R	MCC61R	MCC60R
r	w	r		w	w	w

7	6	5	3	2	1	0
RES	MCC63S	RES		MCC62S	MCC61S	MCC60S
r	w	r		w	w	w

Field	Bits	Type	Description
RES	15	r	**Reserved**
MCC63R	14	w	**Capture/Compare Status Modification Bits (Reset)** These bits are used to reset the corresponding CC63ST bits by software. This feature allows the user to individually change the status of the output lines by software, e.g. when the corresponding compare timer is stopped. This allows a bit manipulation of CC63ST-bits by a single data write action. Functionality see **Table 329**.
RES	13:11	r	**Reserved** Returns 0 if read.
MCC62R	10	w	**Capture/Compare Status Modification Bit 2(Reset)** This bit is used to reset the corresponding CC62ST bits by software. This feature allows the user to individually change the status of the output lines by software, e.g. when the corresponding compare timer is stopped. This allows a bit manipulation of CC62ST-bits by a single data write action. Functionality see **Table 329**.
MCC61R	9	w	**Capture/Compare Status Modification Bit 1(Reset)** This bit is used to reset the corresponding CC61ST bits by software. This feature allows the user to individually change the status of the output lines by software, e.g. when the corresponding compare timer is stopped. This allows a bit manipulation of CC61ST-bits by a single data write action. Functionality see **Table 329**.
MCC60R	8	w	**Capture/Compare Status Modification Bit 0(Reset)** This bit is used to reset the corresponding CC60ST bits by software. This feature allows the user to individually change the status of the output lines by software, e.g. when the corresponding compare timer is stopped. This allows a bit manipulation of CC620T-bits by a single data write action. Functionality see **Table 329**.
RES	7	r	**Reserved**
MCC63S	6	w	**Capture/Compare Status Modification Bits (Set)** This bit is used to set the corresponding CC63ST bits by software. This feature allows the user to individually change the status of the output lines by software, e.g. when the corresponding compare timer is stopped. This allows a bit manipulation of CC63ST-bits by a single data write action. Functionality see **Table 329**.
RES	5:3	r	**Reserved**

Field	Bits	Type	Description
MCC62S	2	w	**Capture/Compare Status Modification Bit 2 (Set)** This bit is used to set the corresponding CC62ST bits by software. This feature allows the user to individually change the status of the output lines by software, e.g. when the corresponding compare timer is stopped. This allows a bit manipulation of CC62ST-bits by a single data write action. Functionality see **Table 329**.
MCC61S	1	w	**Capture/Compare Status Modification Bit 1 (Set)** This bit is used to set the corresponding CC61ST bits by software. This feature allows the user to individually change the status of the output lines by software, e.g. when the corresponding compare timer is stopped. This allows a bit manipulation of CC61ST-bits by a single data write action. Functionality see **Table 329**.
MCC60S	0	w	**Capture/Compare Status Modification Bit 0 (Set)** This bit is used to set the corresponding CC60ST bits by software. This feature allows the user to individually change the status of the output lines by software, e.g. when the corresponding compare timer is stopped. This allows a bit manipulation of CC60ST-bits by a single data write action. Functionality see **Table 329**.

Table 328 RESET of CCU6_CMPMODIF

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Table 329 Capture/Compare Status Modification Bits (Set and Reset)

Field	Bits	Description
MCC60S, MCC61S, MCC62S, MCC63S	0 1 2 6	The following functionality of a write access to bits concerning the same capture/compare state bit is provided (x = 0, 1, 2, 3): MCC6xR, MCC6xS = 00_B , Bit CC6xST is not changed. 01_B , Bit CC6xST is set. 10_B , Bit CC6xST is reset. 11_B , Reserved (toggle)
MCC60R, MCC61R, MCC62R, MCC63R	8 9 10 14	

Timer Control Register 0

Register TCTR0 controls the basic functionality of both timers T12 and T13.

Note: A write action to the bit fields T12CLK or T12PRE is only taken into account while the timer T12 is not running (T12R = 0). A write action to the bit fields T13CLK or T13PRE is only taken into account while the timer T13 is not running (T13R = 0).

CCU6_TCTR0	Offset	Reset Value
Timer Control Register 0	30_H	see Table 330

15	14	13	12	11	10	9	8
RES		STE13	T13R	T13PRE	T13CLK		
r		rh	rh	rw	rw		

7	6	5	4	3	2	1	0
CTM	CDIR	STE12	T12R	T12PRE	T12CLK		
rw	rh	rh	rh	rw	rw		

Field	Bits	Type	Description
RES	15:14	r	**Reserved** Returns 0 if read.
STE13	13	rh	**Timer T13 Shadow Transfer Enable** Bit STE13 enables or disables the shadow transfer of the T13 period value, the compare value and passive state select bit and level from their shadow registers to the actual registers if a T13 shadow transfer event is detected. Bit STE13 is cleared by hardware after the shadow transfer. A T13 shadow transfer event is a period-match. 0_B **Disabled**, The shadow register transfer is disabled. 1_B **Enabled**, The shadow register transfer is enabled.
T13R	12	rh	**Timer T13 Run Bit** T13R starts and stops timer T13. It is set/reset by software by setting bits T13RS or T13RR or it is set/reset by hardware according to the function defined by bit fields T13SSC, T13TEC and T13TED. A concurrent set/reset action on T13R (from T13SSC, T13TEC, T13RR or T13RS) will have no effect. The bit T13R will remain unchanged. 0_B **Stop**, Timer T13 is stopped. 1_B **Run**, Timer T13 is running.
T13PRE	11	rw	**Timer T13 Prescaler Bit** In order to support higher clock frequencies, an additional prescaler factor of 1/256 can be enabled for the prescaler for T13. 0_B **Disabled**, The additional prescaler for T13 is disabled. 1_B **Enabled**, The additional prescaler for T13 is enabled.

Field	Bits	Type	Description
T13CLK	10:8	rw	**Timer T13 Input Clock Select** Selects the input clock for timer T13 which is derived from the peripheral clock according to the equation$f_{T13} = f_{CCU}/2^{<T13CLK>}$. 000_B **1**, $f_{T13} = f_{CCU}$ 001_B **2**, $f_{T13} = f_{CCU} / 2$ 010_B **4**, $f_{T13} = f_{CCU} / 4$ 011_B **8**, $f_{T13} = f_{CCU} / 8$ 100_B **16**, $f_{T13} = f_{CCU} / 16$ 101_B **32**, $f_{T13} = f_{CCU} / 32$ 110_B **64**, $f_{T13} = f_{CCU} / 64$ 111_B **128**, $f_{T13} = f_{CCU} / 128$
CTM	7	rw	**T12 Operating Mode** 0_B **Edge-aligned Mode**, T12 always counts up and continues counting from zero after reaching the period value. 1_B **Center-aligned Mode**, T12 counts down after detecting a period-match and counts up after detecting a one-match.
CDIR	6	rh	**Count Direction of Timer T12** This bit is set/reset according to the counting rules of T12. 0_B **UP**, T12 counts up. 1_B **DOWN**, T12 counts down.
STE12	5	rh	**Timer T12 Shadow Transfer Enable** Bit STE12 enables or disables the shadow transfer of the T12 period value, the compare values and passive state select bits and levels from their shadow registers to the actual registers if a T12 shadow transfer event is detected. Bit STE12 is cleared by hardware after the shadow transfer. A T12 shadow transfer event is a period-match while counting up or a one-match while counting down. 0_B **Disabled**, The shadow register transfer is disabled. 1_B **Enabled**, The shadow register transfer is enabled.
T12R	4	rh	**Timer T12 Run Bit** T12R starts and stops timer T12. It is set/reset by software by setting bits T12RS or T12RR, or it is reset by hardware according to the function defined by bit field T12SSC. A concurrent set/reset action on T12R (from T12SSC, T12RR or T12RS) will have no effect. The bit T12R will remain unchanged. 0_B **Stop**, Timer T12 is stopped. 1_B **Run**, Timer T12 is running.
T12PRE	3	rw	**Timer T12 Prescaler Bit** In order to support higher clock frequencies, an additional prescaler factor of 1/256 can be enabled for the prescaler for T12. 0_B **Disabled**, The additional prescaler for T12 is disabled. 1_B **Enabled**, The additional prescaler for T12 is enabled.

Field	Bits	Type	Description
T12CLK	2:0	rw	**Timer T12 Input Clock Select** Selects the input clock for timer T12 which is derived from the peripheral clock according to the equation $f_{T12} = f_{CCU} / 2^{<T12CLK>}$. 000_B **1**, $f_{T12} = f_{CCU}$ 001_B **2**, $f_{T12} = f_{CCU} / 2$ 010_B **4**, $f_{T12} = f_{CCU} / 4$ 011_B **8**, $f_{T12} = f_{CCU} / 8$ 100_B **16**, $f_{T12} = f_{CCU} / 16$ 101_B **32**, $f_{T12} = f_{CCU} / 32$ 110_B **64**, $f_{T12} = f_{CCU} / 64$ 111_B **128**, $f_{T12} = f_{CCU} / 128$

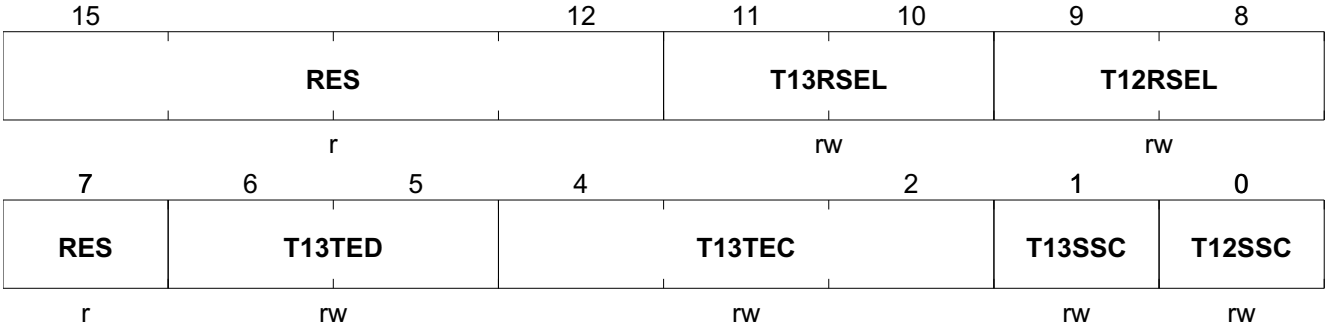
Table 330 RESET of CCU6_TCTR0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Timer Control Register 2

Register TCTR2 controls the single-shot and the synchronization functionality of both timers T12 and T13. Both timers can run in single-shot mode. In this mode, they stop their counting sequence automatically after one counting period with a count value of zero. The single-shot mode and the synchronization feature of T13 to T12 allow the generation of events with a programmable delay after well-defined PWM actions of T12. For example, this feature can be used to trigger AD conversions, after a specified delay (to avoid problems due to switching noise), synchronously to a PWM event.

CCU6_TCTR2	Offset	Reset Value
Timer Control Register 2	58_H	see Table 331



Field	Bits	Type	Description
RES	15:12	r	**Reserved** Returns 0 if read.
T13RSEL	11:10	rw	**Timer T13 External Run Selection** Bit field T13RSEL defines the event of signal T13HR that can set the run bit T13R by hardware. 00_B **Disabled**, The external setting of T13R is disabled. 01_B **Rising edge**, Bit T13R is set if a rising edge of signal T13HR is detected. 10_B **Falling edge**, Bit T13R is set if a falling edge of signal T13HR is detected. 11_B **Edge**, Bit T13R is set if an edge of signal T13HR is detected.
T12RSEL	9:8	rw	**Timer T12 External Run Selection** Bit field T12RSEL defines the event of signal T12HR that can set the run bit T12R by hardware. 00_B **Disabled**, The external setting of T12R is disabled. 01_B **Rising edge**, Bit T12R is set if a rising edge of signal T12HR is detected. 10_B **Falling edge**, Bit T12R is set if a falling edge of signal T12HR is detected. 11_B **Edge**, Bit T12R is set if an edge of signal T12HR is detected.
RES	7	r	**Reserved** Returns 0 if read.

Field	Bits	Type	Description
T13TED	6:5	rw	**Timer T13 Trigger Event Direction** Bit field T13TED delivers additional information to control the automatic set of bit T13R in the case that the trigger action defined by T13TEC is detected. 00_B **No action**, 01_B **Up**, while T12 is counting up 10_B **Down**, while T12 is counting down 11_B **Independent**, independent on the count direction of T12
T13TEC	4:2	rw	**T13 Trigger Event Control** Bit field T13TEC selects the trigger event to start T13 (automatic set of T13R for synchronization to T12 compare signals) according to following combinations: 000_B **No action**, 001_B **Channel 0**, set T13R on a T12 compare event on channel 0 010_B **Channel 1**, set T13R on a T12 compare event on channel 1 011_B **Channel 2**, set T13R on a T12 compare event on channel 2 100_B **Channel 0,1,2**, set T13R on any T12 compare event on the channels 0, 1, or 2 101_B **Period-match**, set T13R upon a period-match of T12 110_B **Zero-match**, set T13R upon a zero-match of T12 (while counting up) 111_B **CCPOSx**, set T13R on any edge of inputs CCPOSx
T13SSC	1	rw	**Timer T13 Single Shot Control** This bit controls the single shot-mode of T13. 0_B **No action**, No hardware action on T13R 1_B **Enabled**, The single-shot mode is enabled, the bit T13R is reset by hardware if T13 reaches its period value. In parallel to the reset action of bit T13R, the bit CC63ST is reset.
T12SSC	0	rw	**Timer T12 Single Shot Control** This bit controls the single shot-mode of T12. 0_B **Disabled**, The single-shot mode is disabled, no hardware action on T12R. 1_B **Enabled**, The single shot mode is enabled, the bit T12R is reset by hardware if: – T12 reaches its period value in edge-aligned mode – T12 reaches the value 1 while down counting in center-aligned mode. In parallel to the reset action of bit T12R, the bits CC6xST (x = 0, 1, 2) are reset.

Table 331 RESET of CCU6_TCTR2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Example

If the timer T13 is intended to start at any compare event on T12 (T13TEC = 100_B), the trigger event direction can be programmed to:

- counting up >> a T12 channel 0, 1, 2 compare match triggers T13R only while T12 is counting up
- counting down >> a T12 channel 0, 1, 2 compare match triggers T13R only while T12 is counting down
- independent from bit CDIR >> each T12 channel 0, 1, 2 compare match triggers T13R

The timer count direction is taken from the value of bit CDIR. As a result, if T12 is running in edge-aligned mode (counting up only), T13 can only be started automatically if bit field T13TED = 01_B or 11_B.

Timer Control Register 4

Register TCTR4 provides software-control (independent set and clear conditions) for the run bits T12R and T13R. Furthermore, the timers can be reset (while running) and bits STE12 and STE13 can be controlled by software. Reading these bits always returns 0.

CCU6_TCTR4	Offset	Reset Value
Timer Control Register 4	04_H	see Table 332

15	14	13	12	11	10	9	8
T13STD	T13STR	T13CNT	RES		T13RES	T13RS	T13RR
w	w	w	r		w	w	w
7	6	5	4	3	2	1	0
T12STD	T12STR	T12CNT	RES	DTRES	T12RES	T12RS	T12RR
w	w	w	r	w	w	w	w

Field	Bits	Type	Description
T13STD	15	w	**Timer T13 Shadow Transfer Disable** 0_B **No action**, 1_B **STE13 reset**, STE13 is reset without triggering the shadow transfer.
T13STR	14	w	**Timer T13 Shadow Transfer Request** 0_B **No action**, 1_B **STE13 set**, STE13 is set, enabling the shadow transfer.
T13CNT	13	w	**Timer T13 Count Event** 0_B **No action**, 1_B **Count**, If enabled (PISEL2), timer T13 counts one step.
RES	12:11	r	**Reserved** Returns 0 if read.
T13RES	10	w	**Timer T13 Reset** 0_B **No effect**, No effect on T13. 1_B **Zero**, The T13 counter register is reset to zero. The switching of the output signals is according to the switching rules. Setting of T13RES has no impact on bit T13R.

Field	Bits	Type	Description
T13RS	9	w	**Timer T13 Run Set** Setting this bit sets the T13R bit. 0_B **No influence**, T13R is not influenced. 1_B **T13R set**, T13R is set, T13 counts.
T13RR	8	w	**Timer T13 Run Reset** Setting this bit resets the T13R bit. 0_B **No influence**, T13R is not influenced. 1_B **T13R cleared**, T13R is cleared, T13 stops counting.
T12STD	7	w	**Timer T12 Shadow Transfer Disable** 0_B **No action**, 1_B **STE12 reset**, STE12 is reset without triggering the shadow transfer.
T12STR	6	w	**Timer T12 Shadow Transfer Request** 0_B **No action**, 1_B **STE12 set**, STE12 is set, enabling the shadow transfer.
T12CNT	5	w	**Timer T12 Count Event** 0_B **No action**, 1_B **Count**, If enabled (PISEL2), timer T12 counts one step.
RES	4	r	**Reserved** Returns 0 if read.
DTRES	3	w	**Dead-Time Counter Reset** 0_B **No effect**, No effect on the dead-time counters. 1_B **Zero**, The three dead-time counter channels are reset to zero.
T12RES	2	w	**Timer T12 Reset** 0_B **No effect**, No effect on T12. 1_B **Zero**, The T12 counter register is reset to zero. The switching of the output signals is according to the switching rules. Setting of T12RES has no impact on bit T12R.
T12RS	1	w	**Timer T12 Run Set** Setting this bit sets the T12R bit. 0_B **No influence**, T12R is not influenced. 1_B **T12R set**, T12R is set, T12 counts.
T12RR	0	w	**Timer T12 Run Reset** Setting this bit resets the T12R bit. 0_B **No influence**, T12R is not influenced. 1_B **T12R cleared**, T12R is cleared, T12 stops counting.

Table 332 RESET of CCU6_TCTR4

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Note: A simultaneous write of a 1 to bits which set and reset the same bit will trigger no action. The corresponding bit will remain unchanged.

18.10.5 Global Modulation Control Registers

Modulation Control Register

Register MODCTR contains control bits enabling the modulation of the corresponding output signal by PWM pattern generated by the timers T12 and T13. Furthermore, the multi-channel mode can be enabled as additional modulation source for the output signals.

CCU6_MODCTR	Offset	Reset Value
Modulation Control Register	$5C_H$	see Table 333

Field	Bits	Type	Description
ECT13O	15	rw	**Enable Compare Timer T13 Output** 0_B **Disabled**, The alternate output function COUT63 is disabled. 1_B **Enabled**, The alternate output function COUT63 is enabled for the PWM signal generated by T13.
RES	14	r	**Reserved** Returns 0 if read.
T13MODEN	13:8	rw	**T13 Modulation Enable** Setting these bits enables the modulation of the corresponding compare channel by a PWM pattern generated by timer T13. The bit positions are corresponding to the following output signals: Bit 0: modulation of CC60 Bit 1: modulation of COUT60 Bit 2: modulation of CC61 Bit 3: modulation of COUT61 Bit 4: modulation of CC62 Bit 5: modulation of COUT62 The enable feature of the modulation is defined as follows: 0_B **Disabled**, The modulation of the corresponding output signal by a T13 PWM pattern is disabled. 1_B **Enabled**, The modulation of the corresponding output signal by a T13 PWM pattern is enabled.

Field	Bits	Type	Description
MCMEN	7	rw	**Multi-Channel Mode Enable** 0_B **Disabled**, The modulation of the corresponding output signal by a multi-channel pattern according to bit field MCMOUT is disabled. 1_B **Enabled**, The modulation of the corresponding output signal by a multi-channel pattern according to bit field MCMOUT is enabled.
RES	6	r	**Reserved** Returns 0 if read.
T12MODEN	5:0	rw	**T12 Modulation Enable** Setting these bits enables the modulation of the corresponding compare channel by a PWM pattern generated by timer T12. The bit positions are corresponding to the following output signals: Bit 0: modulation of CC60 Bit 1: modulation of COUT60 Bit 2: modulation of CC61 Bit 3: modulation of COUT61 Bit 4: modulation of CC62 Bit 5: modulation of COUT62 The enable feature of the modulation is defined as follows: 0_B **Disabled**, The modulation of the corresponding output signal by a T12 PWM pattern is disabled. 1_B **Enabled**, The modulation of the corresponding output signal by a T12 PWM pattern is enabled.

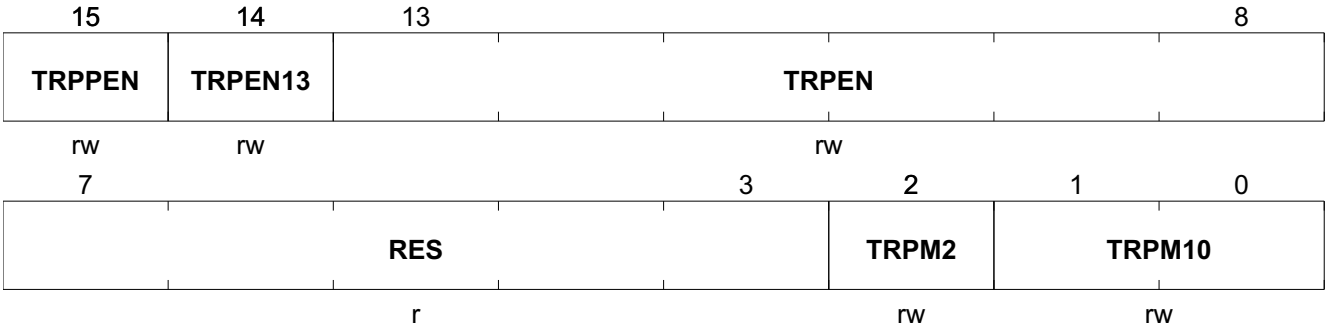
Table 333 RESET of CCU6_MODCTR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Trap Control Register

The register TRPCTR controls the trap functionality. It contains independent enable bits for each output signal and control bits to select the behavior in case of a trap condition. The trap condition is a low-level on the $\overline{\text{CTRAP}}$ input pin, which is monitored (inverted level) by bit IS.TRPF. While TRPF = 1 (trap input active), the trap state bit IS.TRPS is set to 1.

CCU6_TRPCTR	Offset	Reset Value
Trap Control Register	**60_H**	**see Table 334**



Field	Bits	Type	Description
TRPPEN	15	rw	**Trap Pin Enable** 0_B **Disabled**, The trap functionality based on the input pin $\overline{\text{CTRAP}}$ is disabled. A trap can only be generated by software by setting bit TRPF. 1_B **Enabled**, The trap functionality based on the input pin $\overline{\text{CTRAP}}$ is enabled. A trap can be generated by software by setting bit TRPF or by $\overline{\text{CTRAP}}$ = 0.
TRPEN13	14	rw	**Trap Enable Control for Timer T13** 0_B **Disabled**, The trap functionality for T13 is disabled. Timer T13 (if selected and enabled) provides PWM functionality even while TRPS = 1. 1_B **Enabled**, The trap functionality for T13 is enabled. The timer T13 PWM output signal is set to the passive state while TRPS = 1.
TRPEN	13:8	rw	**Trap Enable Control** Setting these bits enables the trap functionality for the following corresponding output signals: Bit 0: trap functionality of CC60 Bit 1: trap functionality of COUT60 Bit 2: trap functionality of CC61 Bit 3: trap functionality of COUT61 Bit 4: trap functionality of CC62 Bit 5: trap functionality of COUT62 The enable feature of the trap functionality is defined as follows: 0_B **Disabled**, The trap functionality of the corresponding output signal is disabled. The output state is independent from bit TRPS. 1_B **Enabled**, The trap functionality of the corresponding output signal is enabled. The output is set to the passive state while TRPS = 1.
RES	7:3	r	**Reserved** Returns 0 if read.

Field	Bits	Type	Description
TRPM2	2	rw	**Trap Mode Control Bit 2** 0_B **Hardware reset**, The trap state can be left (return to normal operation = bit TRPS = 0) as soon as the input $\overline{\text{CTRAP}}$ becomes inactive. Bit TRPF is automatically cleared by hardware if the input pin $\overline{\text{CTRAP}}$ becomes 1. Bit TRPS is automatically cleared by hardware if bit TRPF is 0 and if the synchronization condition (according to TRPM10) is detected. 1_B **Software reset**, The trap state can be left (return to normal operation = bit TRPS = 0) as soon as bit TRPF is reset by software after the input $\overline{\text{CTRAP}}$ becomes inactive (TRPF is not cleared by hardware). Bit TRPS is automatically cleared by hardware if bit TRPF = 0 and if the synchronization condition (according to TRPM10) is detected.
TRPM10	1:0	rw	**Trap Mode Control Bits 1, 0** These two bits define the behavior of the selected outputs when leaving the trap state after the trap condition has become inactive again. A synchronization to the timer driving the PWM pattern permits to avoid unintended short pulses when leaving the trap state. The combination (TRPM1, TRPM0) leads to: 00_B **T12 zero-match**, The trap state is left (return to normal operation according to TRPM2) when a zero-match of T12 (while counting up) is detected (synchronization to T12). 01_B **T13 zero-match**, The trap state is left (return to normal operation according to TRPM2) when a zero-match of T13 is detected (synchronization to T13). 10_B **Reserved**, 11_B **Immediately**, The trap state is left (return to normal operation according to TRPM2) immediately without any synchronization to T12 or T13.

Table 334 RESET of CCU6_TRPCTR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

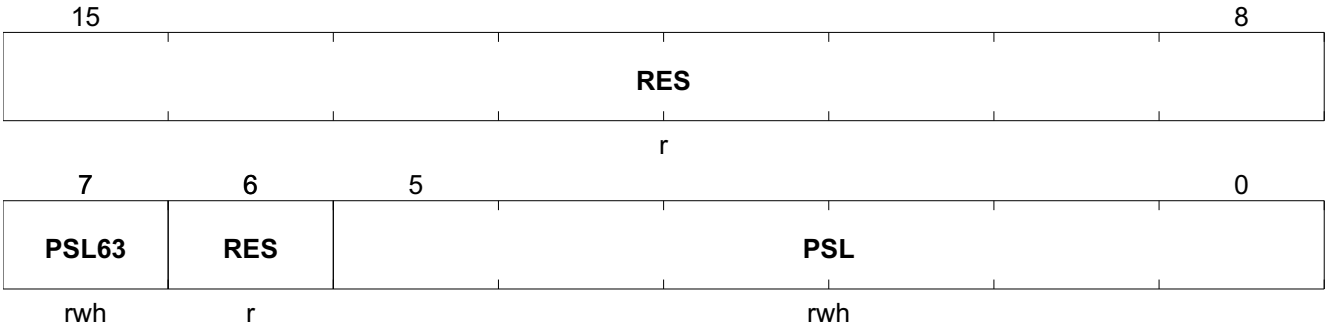
Table 335 Trap Mode Control Bits 1, 0

Field	Bits	Description
TRPM0, TRPM1	0 1	A synchronization to the timer driving the PWM pattern permits to avoid unintended short pulses when leaving the trap state. The combination (TRPM1, TRPM0) leads to: 00_B , The trap state is left (return to normal operation according to TRPM2) when a zero-match of T12 (while counting up) is detected (synchronization to T12). 01_B , The trap state is left (return to normal operation according to TRPM2) when a zero-match of T13 is detected (synchronization to T13). 10_B , reserved 11_B , The trap state is left (return to normal operation according to TRPM2) immediately without any synchronization to T12 or T13.

Passive State Level Register

Register PSLR defines the passive state level driven by the output pins of the module. The passive state level is the value that is driven by the port pin during the passive state of the output. During the active state, the corresponding output pin drives the active state level, which is the inverted passive state level. The passive state level permits the adaptation of the driven output levels to the driver polarity (inverted, not inverted) of the connected power stage. The bits in this register have shadow bit fields to permit a concurrent update of all PWM-related parameters (bit field PSL is updated with T12_ST, whereas PSL63 is updated with T13_ST). The actually used values can be read (attribute “rh”), whereas the shadow bits can only be written (attribute “w”).

CCU6_PSLR	Offset	Reset Value
Passive State Level Register	50_H	see Table 336



Field	Bits	Type	Description
RES	15:8	r	**Reserved**
PSL63	7	rwh	**Passive State Level of Output COUT63** This bit field defines the passive level of the output pin COUT63. 0_B **Level 0**, The passive level is 0. 1_B **Level 1**, The passive level is 1.
RES	6	r	**Reserved** Returns 0 if read.
PSL	5:0	rwh	**Compare Outputs Passive State Level** The bits of this bit field define the passive level driven by the module outputs during the passive state. The bit positions are: Bit 0: passive level for output CC60 Bit 1: passive level for output COUT60 Bit 2: passive level for output CC61 Bit 3: passive level for output COUT61 Bit 4: passive level for output CC62 Bit 5: passive level for output COUT62 The value of each bit position is defined as: 0_B **Level 0**, The passive level is 0. 1_B **Level 1**, The passive level is 1.

Table 336 RESET of CCU6_PSLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Notes

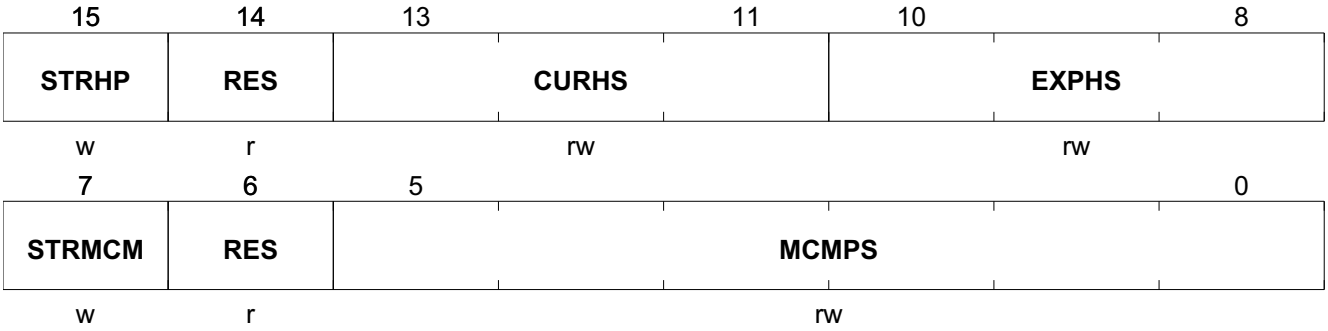
1. *Bit field PSL has a shadow register to allow for updates without undesired pulses on the output lines. The bits are updated with the T12 shadow transfer. A read action targets the actually used values, whereas a write action targets the shadow bits.*
2. *Bit field PSL63 has a shadow register to allow for updates without undesired pulses on the output line. The bit is updated with the T13 shadow transfer. A read action targets the actually used values, whereas a write action targets the shadow bits.*

18.10.6 Multi-Channel Modulation Control Registers

Multi-Channel Mode Output Shadow Register

Register MCMOUTS contains bits used as pattern input for the multi-channel mode and the Hall mode. This register is a shadow register (that can be read and written) for register MCMOUT, which indicates the currently active signals.

CCU6_MCMOUTS	Offset	Reset Value
Multi-Channel Mode Output Shadow Register	08_H	see Table 337



Field	Bits	Type	Description
STRHP	15	w	**Shadow Transfer Request for the Hall Pattern** Setting these bits during a write action leads to an immediate update of bit fields CURH and EXPH by the value written to bit fields CURHS and EXPH. This functionality permits an update triggered by software. When read, this bit always delivers 0. 0_B **by Hardware**, The bit fields CURH and EXPH are updated according to the defined hardware action. The write access to bit fields CURHS and EXPHS does not modify the bit fields CURH and EXPH. 1_B **by Software**, The bit fields CURH and EXPH are updated by the value written to the bit fields CURHS and EXPHS.

Field	Bits	Type	Description
RES	14	r	**Reserved** Returns 0 if read.
CURHS	13:11	rw	**Current Hall Pattern Shadow** Bit field CURHS is the shadow bit field for bit field CURH. The bit field is transferred to bit field CURH if an edge on the hall input pins CCPOSx (x = 0, 1, 2) is detected.
EXPHS	10:8	rw	**Expected Hall Pattern Shadow** Bit field EXPHS is the shadow bit field for bit field EXPH. The bit field is transferred to bit field EXPH if an edge on the hall input pins CCPOSx (x = 0, 1, 2) is detected.
STRMCM	7	w	**Shadow Transfer Request for MCMPS** Setting this bit during a write action leads to an immediate update of bit field MCMP by the value written to bit field MCMPS. This functionality permits an update triggered by software. When read, this bit always delivers 0. 0_B **by Hardware**, Bit field MCMP is updated according to the defined hardware action. The write access to bit field MCMPS does not modify bit field MCMP. 1_B **by Software**, Bit field MCMP is updated by the value written to bit field MCMPS.
RES	6	r	**Reserved** Returns 0 if read.
MCMPS	5:0	rw	**Multi-Channel PWM Pattern Shadow** Bit field MCMPS is the shadow bit field for bit field MCMP. The multi-channel shadow transfer is triggered according to the transfer conditions defined by register MCMCTR.

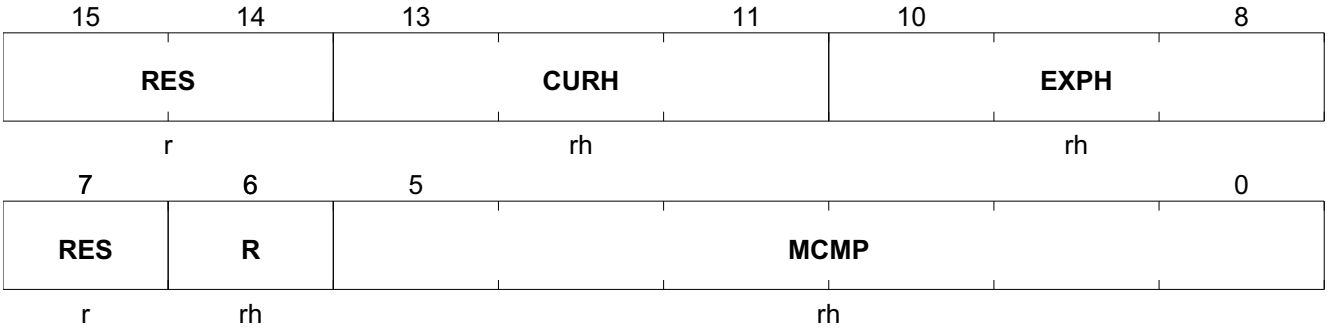
Table 337 RESET of CCU6_MCMOUTS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Multi-Channel Mode Output Register

Register MCMOUT shows the multi-channel control bits that are currently used. Register MCMOUT is defined as follows:

CCU6_MCMOUT	Offset	Reset Value
Multi-Channel Mode Output Register	**64_H**	**see Table 338**



Field	Bits	Type	Description
RES	15:14	r	**Reserved** Returns 0 if read.
CURH	13:11	rh	**Current Hall Pattern** Bit field CURH is written by a shadow transfer from bit field CURHS.The contents are compared after every detected edge at the hall input pins with the pattern at the hall input pins in order to detect the occurrence of the next desired (= expected) hall pattern or a wrong pattern. If the current hall input pattern is equal to bit field CURH, the detected edge at the hall input pins has been an invalid transition (e.g. a spike).
EXPH	10:8	rh	**Expected Hall Pattern** Bit field EXPH is written by a shadow transfer from bit field EXPHS. The contents are compared after every detected edge at the hall input pins with the pattern at the hall input pins in order to detect the occurrence of the next desired (= expected) hall pattern or a wrong pattern. If the current hall pattern at the hall input pins is equal to the bit field EXPH, bit CHE (correct hall event) is set and an interrupt request is generated (if enabled by bit ENCHE). If the current hall pattern at the hall input pins is not equal to the bit fields CURH or EXPH, bit WHE (wrong hall event) is set and an interrupt request is generated (if enabled by bit ENWHE).
RES	7	r	**Reserved** Returns 0 if read.
R	6	rh	**Reminder Flag** This reminder flag indicates that the shadow transfer from bit field MCMPS to MCMP has been requested by the selected trigger source. This bit is cleared when the shadow transfer takes place and while MCMEN = 0. 0_B **No shadow transfer**, Currently, no shadow transfer from MCMPS to MCMP is requested. 1_B **Shadow transfer**, A shadow transfer from MCMPS to MCMP has been requested by the selected trigger source, but it has not yet been executed, because the selected synchronization condition has not yet occurred.

Field	Bits	Type	Description
MCMP	5:0	rh	**Multi-Channel PWM Pattern** Bit field MCMP is written by a shadow transfer from bit field MCMPS. It contains the output pattern for the multi-channel mode. If this mode is enabled by bit MCMEN in register MODCTR, the output state of the following output signal can be modified: Bit 0: multi-channel state for output CC60 Bit 1: multi-channel state for output COUT60 Bit 2: multi-channel state for output CC61 Bit 3: multi-channel state for output COUT61 Bit 4: multi-channel state for output CC62 Bit 5: multi-channel state for output COUT62 The multi-channel patterns can set the related output to the passive state. While IDLE = 1, bit field MCMP is cleared. 0_B **Passive**, The output is set to the passive state. The PWM generated by T12 or T13 is not taken into account. 1_B **PWM**, The output can deliver the PWM generated by T12 or T13 (according to register MODCTR).

Table 338 RESET of CCU6_MCMOUT

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Note: The bits in the bit fields EXPH and CURH correspond to the hall patterns at the input pins CCPOSx (x = 0, 1, 2) in the following order (EXPH.2, EXPH.1, EXPH.0), (CURH.2, CURH.1, CURH.0), (CCPOS2, CCPOS.1, CCPOS0).

Multi-Channel Mode Control Register

Register MCMCTR contains control bits for the multi-channel functionality.

CCU6_MCMCTR	Offset	Reset Value
Multi-Channel Mode Control Register	**54_H**	**see Table 339**

15	14	13	12	11	10	9	8
RES					STE13U	STE12D	STE12U
r					rw	rw	rw

7	6	5	4	3	2	1	0
RES		SWSYN		RES	SWSEL		
r		rw		r	rw		

Field	Bits	Type	Description
RES	15:11	r	**Reserved**

Field	Bits	Type	Description
STE13U	10	rw	**Shadow Transfer Enable for T13 Upcounting** This bit enables the shadow transfer T13_ST if flag MCMOUT.R is set or becomes set while a T13 period match is detected. 0_B **No action**, 1_B **Enabled**, The T13_ST shadow transfer mechanism is enabled if MCMEN = 1.
STE12D	9	rw	**Shadow Transfer Enable for T12 Downcounting** This bit enables the shadow transfer T12_ST if flag MCMOUT.R is set or becomes set while a T12 one match is detected while counting down. 0_B **No action**, 1_B **Enabled**, The T12_ST shadow transfer mechanism is enabled if MCMEN = 1.
STE12U	8	rw	**Shadow Transfer Enable for T12 Upcounting** This bit enables the shadow transfer T12_ST if flag MCMOUT.R is set or becomes set while a T12 period match is detected while counting up. 0_B **No action**, 1_B **Enabled**, The T12_ST shadow transfer mechanism is enabled if MCMEN = 1.
RES	7:6	r	**Reserved** Returns 0 if read.
SWSYN	5:4	rw	**Switching Synchronization** Bit field SWSYN triggers the shadow transfer between MCMPS and MCMP if it has been requested before (flag R set by an event selected by SWSEL). This feature permits the synchronization of the outputs to the PWM source, that is used for modulation (T12 or T13). 00_B **Direct**, the trigger event directly causes the shadow transfer 01_B **T13 zero-match**, T13 zero-match triggers the shadow transfer 10_B **T12 zero-match**, a T12 zero-match (while counting up) triggers the shadow transfer 11_B **Reserved**, reserved; no action
RES	3	r	**Reserved**

Field	Bits	Type	Description
SWSEL	2:0	rw	**Switching Selection** Bit field SWSEL selects one of the following trigger request sources (next multi-channel event) for the shadow transfer from MCMPS to MCMP. The trigger request is stored in the reminder flag R until the shadow transfer is done and flag R is cleared automatically with the shadow transfer. The shadow transfer takes place synchronously with an event selected in bit field SWSYN. 000_B **No request**, no trigger request will be generated 001_B **Correct pattern**, correct hall pattern on CCPOSx detected 010_B **T13 period-match**, T13 period-match detected (while counting up) 011_B **T12 one-match**, T12 one-match (while counting down) 100_B **T12 channel1 compare-match**, T12 channel 1 compare-match detected (phase delay function) 101_B **T12 period-match**, T12 period match detected (while counting up) else reserved, no trigger request will be generated

Table 339 RESET of CCU6_MCMCTR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

18.10.7 Interrupt Control Registers

Capture/Compare Interrupt Status Register

Register IS contains the individual interrupt request bits. This register can only be read; write actions have no impact on the contents of this register. The software can set or reset the bits individually by writing to the registers ISS (to set the bits) or to register ISR (to reset the bits).

The interrupt generation is independent from the value of the bits in register IS, e.g. the interrupt will be generated (if enabled) even if the corresponding bit is already set. The trigger for an interrupt generation is the detection of a set condition (by HW or SW) for the corresponding bit in register IS.

In compare mode (and hall mode), the timer-related interrupts are only generated while the timer is running (T1xR = 1). In capture mode, the capture interrupts are also generated while the timer T12 is stopped.

Note: Not all bits in register IS can generate an interrupt. Other status bits have been added, that have a similar structure for their set and clear actions. It is recommended that SW checks the interrupt bits bit-wisely (instead of common OR over the bits).

CCU6_IS	Offset	Reset Value
Capture/Compare Interrupt Status Register	68_H	see Table 340

15	14	13	12	11	10	9	8
STR	IDLE	WHE	CHE	TRPS	TRPF	T13PM	T13CM
rh	rh	rh	rh	rh	rh	rh	rh
7	6	5	4	3	2	1	0
T12PM	T12OM	ICC62F	ICC62R	ICC61F	ICC61R	ICC60F	ICC60R
rh	rh	rh	rh	rh	rh	rh	rh

Field	Bits	Type	Description
STR	15	rh	**Multi-Channel Mode Shadow Transfer Request** This bit is set when a shadow transfer from MCMOUTS to MCMOUT takes places in multi-channel mode. 0_B **No**, The shadow transfer has not yet taken place. 1_B **Yes**, The shadow transfer has taken place.
IDLE	14	rh	**IDLE State** This bit is set together with bit WHE (wrong hall event) and it must be reset by software. 0_B **No action**, 1_B **Idle**, Bit field MCMP is cleared and held to 0, the selected outputs are set to passive state.

Field	Bits	Type	Description
WHE	13	rh	**Wrong Hall Event** On every valid hall edge, the contents of EXPH are compared with the pattern on pin CCPOSx. If both comparisons (CURH and EXPH with CCPOSx) are not true, bit WHE (wrong hall event) is set. 0_B **Not detected**, A transition to a wrong hall event (not the expected one) has not yet been detected since this bit has been reset for the last time. 1_B **Detected**, A transition to a wrong hall event (not the expected one) has been detected.
CHE	12	rh	**Correct Hall Event** On every valid hall edge, the contents of EXPH are compared with the pattern on pin CCPOSx and if equal bit CHE is set. 0_B **Not detected**, A transition to a correct (= expected) hall event has not yet been detected since this bit has been reset for the last time. 1_B **Detected**, A transition to a correct (= expected) hall event has been detected.
TRPS	11	rh	**Trap State** During the trap state, the selected outputs are set to the passive state. The logic level driven during the passive state is defined by the corresponding bit in register PSLR. Bit TRPS = 1 and TRPF = 0 can occur if the trap condition is no longer active but the selected synchronization has not yet taken place. 0_B **Not active**, The trap state is not active. 1_B **Active**, The trap state is active. Bit TRPS is set while bit TRPF = 1. It is reset according to the mode selected in register TRPCTR.
TRPF	10	rh	**Trap Flag** The trap flag TRPF will be set by hardware if TRPPEN = 1 and $\overline{\text{CTRAP}}$ = 0 or by software. If TRPM2 = 0, bit TRPF is reset by hardware if the input $\overline{\text{CTRAP}}$ becomes inactive (TRPPEN = 1). If TRPM2 = 1, bit TRPF must be reset by software in order to leave the trap state. 0_B **Not detected**, The trap condition has not been detected. 1_B **Detected**, The trap condition has been detected (input $\overline{\text{CTRAP}}$ has been 0 or by software).
T13PM	9	rh	**Timer T13 Period-Match Flag** 0_B **Not detected**, A timer T13 period-match has not yet been detected since this bit has been reset for the last time. 1_B **Detected**, A timer T13 period-match has been detected.
T13CM	8	rh	**Timer T13 Compare-Match Flag** 0_B **Not detected**, A timer T13 compare-match has not yet been detected since this bit has been reset for the last time. 1_B **Detected**, A timer T13 compare-match has been detected.

Field	Bits	Type	Description
T12PM	7	rh	**Timer T12 Period-Match Flag** 0_B **Not detected**, A timer T12 period-match (while counting up) has not yet been detected since this bit has been reset for the last time. 1_B **Detected**, A timer T12 period-match (while counting up) has been detected.
T12OM	6	rh	**Timer T12 One-Match Flag** 0_B **Not detected**, A timer T12 one-match (while counting down) has not yet been detected since this bit has been reset for the last time. 1_B **Detected**, A timer T12 one-match (while counting down) has been detected.
ICC62F	5	rh	**Capture, Compare-Match Falling Edge Flag** In compare mode, a compare-match has been detected while T12 was counting down. In capture mode, a falling edge has been detected at the input CC62. 0_B **Not occurred**, The event has not yet occurred since this bit has been reset for the last time. 1_B **Detected**, The event described above has been detected.
ICC62R	4	rh	**Capture, Compare-Match Rising Edge Flag** In compare mode, a compare-match has been detected while T12 was counting up. In capture mode, a rising edge has been detected at the input CC62. 0_B **Not occurred**, The event has not yet occurred since this bit has been reset for the last time. 1_B **Detected**, The event described above has been detected.
ICC61F	3	rh	**Capture, Compare-Match Falling Edge Flag** In compare mode, a compare-match has been detected while T12 was counting down. In capture mode, a falling edge has been detected at the input CC61. 0_B **Not occurred**, The event has not yet occurred since this bit has been reset for the last time. 1_B **Detected**, The event described above has been detected.
ICC61R	2	rh	**Capture, Compare-Match Rising Edge Flag** In compare mode, a compare-match has been detected while T12 was counting up. In capture mode, a rising edge has been detected at the input CC61. 0_B **Not occurred**, The event has not yet occurred since this bit has been reset for the last time. 1_B **Detected**, The event described above has been detected.
ICC60F	1	rh	**Capture, Compare-Match Falling Edge Flag** In compare mode, a compare-match has been detected while T12 was counting down. In capture mode, a falling edge has been detected at the input CC60. 0_B **Not occurred**, The event has not yet occurred since this bit has been reset for the last time. 1_B **Detected**, The event described above has been detected.

Field	Bits	Type	Description
ICC60R	0	rh	**Capture, Compare-Match Rising Edge Flag** In compare mode, a compare-match has been detected while T12 was counting up. In capture mode, a rising edge has been detected at the input CC60. 0_B **Not occurred**, The event has not yet occurred since this bit has been reset for the last time. 1_B **Detected**, The event described above has been detected.

Table 340 RESET of CCU6_IS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Capture/Compare Interrupt Status Set Register

Register ISS contains individual interrupt request set bits to generate a Mod_Name interrupt request by software. Writing a 1 sets the bit(s) in register IS at the corresponding bit position(s) and can generate an interrupt event (if available and enabled). All bit positions read as 0.

CCU6_ISS	Offset	Reset Value
Capture/Compare Interrupt Status Set Register	$4C_H$	see Table 341

15	14	13	12	11	10	9	8
SSTR	SIDLE	SWHE	SCHE	SWHC	STRPF	ST13PM	ST13CM
w	w	w	w	w	w	w	w
7	6	5	4	3	2	1	0
ST12PM	ST12OM	SCC62F	SCC62R	SCC61F	SCC61R	SCC60F	SCC60R
w	w	w	w	w	w	w	w

Field	Bits	Type	Description
SSTR	15	w	**Set STR Flag** 0_B **No action**, 1_B **Set**, Bit STR in register IS will be set.
SIDLE	14	w	**Set IDLE Flag** 0_B **No action**, 1_B **Set**, Bit IDLE in register IS will be set.
SWHE	13	w	**Set Wrong Hall Event Flag** 0_B **No action**, 1_B **Set**, Bit WHE in register IS will be set.
SCHE	12	w	**Set Correct Hall Event Flag** 0_B **No action**, 1_B **Set**, Bit CHE in register IS will be set.

Field	Bits	Type	Description
SWHC	11	w	**Software Hall Compare** 0_B **No action**, 1_B **Set**, The Hall compare action is triggered.
STRPF	10	w	**Set Trap Flag** 0_B **No action**, 1_B **Set**, Bits TRPF and TRPS in register IS will be set.
ST13PM	9	w	**Set Timer T13 Period-Match Flag** 0_B **No action**, 1_B **Set**, Bit T13PM in register IS will be set.
ST13CM	8	w	**Set Timer T13 Compare-Match Flag** 0_B **No action**, 1_B **Set**, Bit T13CM in register IS will be set.
ST12PM	7	w	**Set Timer T12 Period-Match Flag** 0_B **No action**, 1_B **Set**, Bit T12PM in register IS will be set.
ST12OM	6	w	**Set Timer T12 One-Match Flag** 0_B **No action**, 1_B **Set**, Bit T12OM in register IS will be set.
SCC62F	5	w	**Set Capture, Compare-Match Falling Edge Flag** 0_B **No action**, 1_B **Set**, Bit CC62F in register IS will be set.
SCC62R	4	w	**Set Capture, Compare-Match Rising Edge Flag** 0_B **No action**, 1_B **Set**, Bit CC62R in register IS will be set.
SCC61F	3	w	**Set Capture, Compare-Match Falling Edge Flag** 0_B **No action**, 1_B **Set**, Bit CC61F in register IS will be set.
SCC61R	2	w	**Set Capture, Compare-Match Rising Edge Flag** 0_B **No action**, 1_B **Set**, Bit CC61R in register IS will be set.
SCC60F	1	w	**Set Capture, Compare-Match Falling Edge Flag** 0_B **No action**, 1_B **Set**, Bit CC60F in register IS will be set.
SCC60R	0	w	**Set Capture, Compare-Match Rising Edge Flag** 0_B **No action**, 1_B **Set**, Bit CC60R in register IS will be set.

Table 341 RESET of CCU6_ISS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Capture/Compare Interrupt Status Reset Register

Register ISR contains bits to individually clear the interrupt event flags by software. Writing a 1 clears the bit(s) in register IS at the corresponding bit position(s). All bit positions read as 0.

CCU6_ISR	Offset	Reset Value
Capture/Compare Interrupt Status Reset Register	$0C_H$	see Table 342

15	14	13	12	11	10	9	8
RSTR	RIDLE	RWHE	RCHE	RES	RTRPF	RT13PM	RT13CM
w	w	w	w	r	w	w	w
7	6	5	4	3	2	1	0
RT12PM	RT12OM	RCC62F	RCC62R	RCC61F	RCC61R	RCC60F	RCC60R
w	w	w	w	w	w	w	w

Field	Bits	Type	Description
RSTR	15	w	**Reset STR Flag** 0_B **No action**, 1_B **Reset**, Bit STR in register IS will be reset.
RIDLE	14	w	**Reset IDLE Flag** 0_B **No action**, 1_B **Reset**, Bit IDLE in register IS will be reset.
RWHE	13	w	**Reset Wrong Hall Event Flag** 0_B **No action**, 1_B **Reset**, Bit WHE in register IS will be reset.
RCHE	12	w	**Reset Correct Hall Event Flag** 0_B **No action**, 1_B **Reset**, Bit CHE in register IS will be reset.
RES	11	r	**Reserved** Returns 0 if read.
RTRPF	10	w	**Reset Trap Flag** 0_B **No action**, 1_B **Reset**, Bit TRPF in register IS will be reset (not taken into account while input $\overline{\text{CTRAP}}$ = 0 and TRPPEN = 1.
RT13PM	9	w	**Reset Timer T13 Period-Match Flag** 0_B **No action**, 1_B **Reset**, Bit T13PM in register IS will be reset.
RT13CM	8	w	**Reset Timer T13 Compare-Match Flag** 0_B **No action**, 1_B **Reset**, Bit T13CM in register IS will be reset.
RT12PM	7	w	**Reset Timer T12 Period-Match Flag** 0_B **No action**, 1_B **Reset**, Bit T12PM in register IS will be reset.
RT12OM	6	w	**Reset Timer T12 One-Match Flag** 0_B **No action**, 1_B **Reset**, Bit T12OM in register IS will be reset.

Field	Bits	Type	Description
RCC62F	5	w	**Reset Capture, Compare-Match Falling Edge Flag** 0_B **No action**, 1_B **Reset**, Bit CC62F in register IS will be reset.
RCC62R	4	w	**Reset Capture, Compare-Match Rising Edge Flag** 0_B **No action**, 1_B **Reset**, Bit CC62R in register IS will be reset.
RCC61F	3	w	**Reset Capture, Compare-Match Falling Edge Flag** 0_B **No action**, 1_B **Reset**, Bit CC61F in register IS will be reset.
RCC61R	2	w	**Reset Capture, Compare-Match Rising Edge Flag** 0_B **No action**, 1_B **Reset**, Bit CC61R in register IS will be reset.
RCC60F	1	w	**Reset Capture, Compare-Match Falling Edge Flag** 0_B **No action**, 1_B **Reset**, Bit CC60F in register IS will be reset.
RCC60R	0	w	**Reset Capture, Compare-Match Rising Edge Flag** 0_B **No action**, 1_B **Reset**, Bit CC60R in register IS will be reset.

Table 342 RESET of CCU6_ISR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Capture/Compare Interrupt Enable Register

Register IEN contains the interrupt enable bits and a control bit to enable the automatic idle function in the case of a wrong hall pattern.

CCU6_IEN	Offset	Reset Value
Capture/Compare Interrupt Enable Register	**44_H**	**see Table 343**

15	14	13	12	11	10	9	8
ENSTR	**ENIDLE**	**ENWHE**	**ENCHE**	**RES**	**ENTRPF**	**ENT13PM**	**ENT13CM**
rw	rw	rw	rw	r	rw	rw	rw
7	6	5	4	3	2	1	0
ENT12PM	**ENT12OM**	**ENCC62F**	**ENCC62R**	**ENCC61F**	**ENCC61R**	**ENCC60F**	**ENCC60R**
rw	rw	rw	rw	rw	rw	rw	rw

Field	Bits	Type	Description
ENSTR	15	rw	**Enable Multi-Channel Mode Shadow Transfer Interrupt** 0_B **No interrupt**, No interrupt will be generated if the set condition for bit STR in register IS occurs. 1_B **Interrupt**, An interrupt will be generated if the set condition for bit STR in register IS occurs. The interrupt line that will be activated is selected by bit field INPCHE.
ENIDLE	14	rw	**Enable Idle** This bit enables the automatic entering of the idle state (bit IDLE will be set) after a wrong hall event has been detected (bit WHE is set). During the idle state, the bit field MCMP is automatically cleared. 0_B **IDLE not set**, The bit IDLE is not automatically set when a wrong hall event is detected. 1_B **IDLE set**, The bit IDLE is automatically set when a wrong hall event is detected.
ENWHE	13	rw	**Enable Interrupt for Wrong Hall Event** 0_B **No interrupt**, No interrupt will be generated if the set condition for bit WHE in register IS occurs. 1_B **Interrupt**, An interrupt will be generated if the set condition for bit WHE in register IS occurs. The interrupt line that will be activated is selected by bit field INPERR.
ENCHE	12	rw	**Enable Interrupt for Correct Hall Event** 0_B **No interrupt**, No interrupt will be generated if the set condition for bit CHE in register IS occurs. 1_B **Interrupt**, An interrupt will be generated if the set condition for bit CHE in register IS occurs. The interrupt line that will be activated is selected by bit field INPCHE.
RES	11	r	**Reserved** Returns 0 if read.
ENTRPF	10	rw	**Enable Interrupt for Trap Flag** 0_B **No interrupt**, No interrupt will be generated if the set condition for bit TRPF in register IS occurs. 1_B **Interrupt**, An interrupt will be generated if the set condition for bit TRPF in register IS occurs. The interrupt line that will be activated is selected by bit field INPERR.
ENT13PM	9	rw	**Enable Interrupt for T13 Period-Match** 0_B **No interrupt**, No interrupt will be generated if the set condition for bit T13PM in register IS occurs. 1_B **Interrupt**, An interrupt will be generated if the set condition for bit T13PM in register IS occurs. The interrupt line that will be activated is selected by bit field INPT13.

Field	Bits	Type	Description
ENT13CM	8	rw	**Enable Interrupt for T13 Compare-Match** 0_B **No interrupt**, No interrupt will be generated if the set condition for bit T13CM in register IS occurs. 1_B **Interrupt**, An interrupt will be generated if the set condition for bit T13CM in register IS occurs. The interrupt line that will be activated is selected by bit field INPT13.
ENT12PM	7	rw	**Enable Interrupt for T12 Period-Match** 0_B **No interrupt**, No interrupt will be generated if the set condition for bit T12PM in register IS occurs. 1_B **Interrupt**, An interrupt will be generated if the set condition for bit T12PM in register IS occurs. The interrupt line that will be activated is selected by bit field INPT12.
ENT12OM	6	rw	**Enable Interrupt for T12 One-Match** 0_B **No interrupt**, No interrupt will be generated if the set condition for bit T12OM in register IS occurs. 1_B **Interrupt**, An interrupt will be generated if the set condition for bit T12OM in register IS occurs. The interrupt line that will be activated is selected by bit field INPT12.
ENCC62F	5	rw	**Capture, Compare-Match Falling Edge Interrupt Enable for Channel 2** 0_B **No interrupt**, No interrupt will be generated if the set condition for bit CC62F in register IS occurs. 1_B **Interrupt**, An interrupt will be generated if the set condition for bit CC62F in register IS occurs. The interrupt line that will be activated is selected by bit field INPCC62.
ENCC62R	4	rw	**Capture, Compare-Match Rising Edge Interrupt Enable for Channel 2** 0_B **No interrupt**, No interrupt will be generated if the set condition for bit CC62R in register IS occurs. 1_B **Interrupt**, An interrupt will be generated if the set condition for bit CC62R in register IS occurs. The interrupt line that will be activated is selected by bit field INPCC62.
ENCC61F	3	rw	**Capture, Compare-Match Falling Edge Interrupt Enable for Channel 1** 0_B **No interrupt**, No interrupt will be generated if the set condition for bit CC61F in register IS occurs. 1_B **Interrupt**, An interrupt will be generated if the set condition for bit CC61F in register IS occurs. The interrupt line that will be activated is selected by bit field INPCC61.
ENCC61R	2	rw	**Capture, Compare-Match Rising Edge Interrupt Enable for Channel 1** 0_B **No interrupt**, No interrupt will be generated if the set condition for bit CC61R in register IS occurs. 1_B **Interrupt**, An interrupt will be generated if the set condition for bit CC61R in register IS occurs. The interrupt line that will be activated is selected by bit field INPCC61.

Field	Bits	Type	Description
ENCC60F	1	rw	**Capture, Compare-Match Falling Edge Interrupt Enable for Channel 0** 0_B **No interrupt**, No interrupt will be generated if the set condition for bit CC60F in register IS occurs. 1_B **Interrupt**, An interrupt will be generated if the set condition for bit CC60F in register IS occurs. The interrupt line that will be activated is selected by bit field INPCC60.
ENCC60R	0	rw	**Capture, Compare-Match Rising Edge Interrupt Enable for Channel 0** 0_B **No interrupt**, No interrupt will be generated if the set condition for bit CC60R in register IS occurs. 1_B **Interrupt**, An interrupt will be generated if the set condition for bit CC60R in register IS occurs. The interrupt line that will be activated is selected by bit field INPCC60.

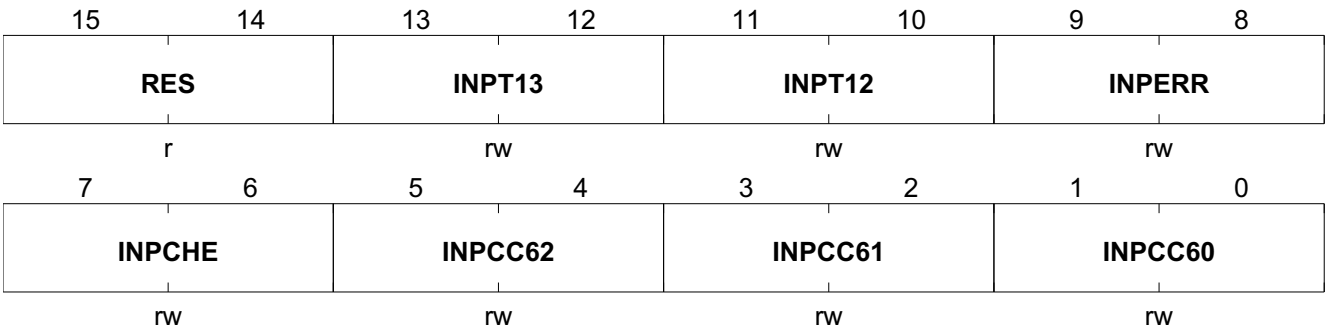
Table 343 RESET of CCU6_IEN

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000_H	RESET_TYPE_3		

Capture/Compare Interrupt Node Pointer Register

Register INP contains the interrupt node pointers allowing a flexible interrupt handling. These bit fields define which service request output will be activated if the corresponding interrupt event occurs and the interrupt generation for this event is enabled.

CCU6_INP	Offset	Reset Value
Capture/Compare Interrupt Node Pointer Register	48_H	see Table 344



Field	Bits	Type	Description
RES	15:14	r	**Reserved** Returns 0 if read.

Field	Bits	Type	Description
INPT13	13:12	rw	**Interrupt Node Pointer for Timer T13 Interrupts** This bit field defines the interrupt output line, which is activated due to a set condition for bit T13CM (if enabled by bit ENT13CM) or for bit T13PM (if enabled by bit ENT13PM). 00_B **SR0**, Interrupt output line SR0 is selected. 01_B **SR1**, Interrupt output line SR1 is selected. 10_B **SR2**, Interrupt output line SR2 is selected. 11_B **SR3**, Interrupt output line SR3 is selected.
INPT12	11:10	rw	**Interrupt Node Pointer for Timer T12 Interrupts** This bit field defines the interrupt output line, which is activated due to a set condition for bit T12OM (if enabled by bit ENT12OM) or for bit T12PM (if enabled by bit ENT12PM). 00_B **SR0**, Interrupt output line SR0 is selected. 01_B **SR1**, Interrupt output line SR1 is selected. 10_B **SR2**, Interrupt output line SR2 is selected. 11_B **SR3**, Interrupt output line SR3 is selected.
INPERR	9:8	rw	**Interrupt Node Pointer for Error Interrupts** This bit field defines the interrupt output line, which is activated due to a set condition for bit TRPF (if enabled by bit ENTRPF) or for bit WHE (if enabled by bit ENWHE). 00_B **SR0**, Interrupt output line SR0 is selected. 01_B **SR1**, Interrupt output line SR1 is selected. 10_B **SR2**, Interrupt output line SR2 is selected. 11_B **SR3**, Interrupt output line SR3 is selected.
INPCHE	7:6	rw	**Interrupt Node Pointer for the CHE Interrupt** This bit field defines the interrupt output line, which is activated due to a set condition for bit CHE (if enabled by bit ENCHE) or for bit STR (if enabled by bit ENSTR). 00_B **SR0**, Interrupt output line SR0 is selected. 01_B **SR1**, Interrupt output line SR1 is selected. 10_B **SR2**, Interrupt output line SR2 is selected. 11_B **SR3**, Interrupt output line SR3 is selected.
INPCC62	5:4	rw	**Interrupt Node Pointer for Channel 2 Interrupts** This bit field defines the interrupt output line, which is activated due to a set condition for bit CC62R (if enabled by bit ENCC62R) or for bit CC62F (if enabled by bit ENCC62F). 00_B **SR0**, Interrupt output line SR0 is selected. 01_B **SR1**, Interrupt output line SR1 is selected. 10_B **SR2**, Interrupt output line SR2 is selected. 11_B **SR3**, Interrupt output line SR3 is selected.

Field	Bits	Type	Description
INPCC61	3:2	rw	**Interrupt Node Pointer for Channel 1 Interrupts** This bit field defines the interrupt output line, which is activated due to a set condition for bit CC61R (if enabled by bit ENCC61R) or for bit CC61F (if enabled by bit ENCC61F). 00_B **SR0**, Interrupt output line SR0 is selected. 01_B **SR1**, Interrupt output line SR1 is selected. 10_B **SR2**, Interrupt output line SR2 is selected. 11_B **SR3**, Interrupt output line SR3 is selected.
INPCC60	1:0	rw	**Interrupt Node Pointer for Channel 0 Interrupts** This bit field defines the interrupt output line, which is activated due to a set condition for bit CC60R (if enabled by bit ENCC60R) or for bit CC60F (if enabled by bit ENCC60F). 00_B **SR0**, Interrupt output line SR0 is selected. 01_B **SR1**, Interrupt output line SR1 is selected. 10_B **SR2**, Interrupt output line SR2 is selected. 11_B **SR3**, Interrupt output line SR3 is selected.

Table 344 RESET of CCU6_INP

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	3940_H	RESET_TYPE_3		

18.11 TLE985xQX Module Implementation Details

This section describes the CCU6 module interfaces with the clock control, port connections, interrupt control, and address decoding.

18.11.1 Interfaces of the CCU6 Module

An overview of the Mod_Name kernel I/O interface is shown in **Figure 166**.

The Bus Peripheral Interface (BPI) enables the Mod_Name kernel to be attached to the 8-bit Bus. The BPI consists of a clock control logic which gates the clock input to the kernel, and an address decoder for Special Function Registers (SFRs) in the Mod_Name kernel.

The interrupt lines of the Mod_Name are connected to the CPU interrupt controller via the SCU. An interrupt pulse can be generated at one of the four interrupt output lines SRCx (x=0 to 4) of the module. More than one CCU6 interrupt source can be connected to each CCU6 interrupt line.

The General Purpose IO (GPIO) Ports provide the interface from the Mod_Name to the external world. Please refer to **Chapter 15** for Port implementation details.

The CCU6 kernel is clocked on PCLK frequency where $f_{CCU} = f_{PCLK}$.

Debug Suspend of Timers

The timers of CCU6, T12 and T13, can be suspended immediately when OCDS enters Monitor Mode and has the Debug-Suspend signal activated – provided the respective timer suspend bits, T12SUSP and T13SUSP (in SCU SFR MODSUSP), are set. When suspended, the respective timer stops and its PWM outputs enabled for the trap condition (**CCU6_TRPCTR**.TRPENx = 1) are set to respective passive levels (similar to TRAP state). In addition, all CCU6 inputs are frozen. Refer to SCU **Chapter 7.10** and OCDS chapter.

Flexible Peripheral Management (Kernel Clock Gating) of CCU6

When not in use, the CCU6 kernel may be disabled where the kernel clock input is gated. When the **SCU_PMCON**.CCU_DIS request bit is set, both T12 and T13 are immediately stopped and PWM outputs enabled for the trap condition (**CCU6_TRPCTR**.TRPENx = 1) are set to respective passive levels (similar to TRAP state). In addition, all CCU6 inputs are frozen. Finally, the kernel clock input is gated. Refer to SCU **Chapter 7.9**.

Table 345 CCU6/T21CCU Interconnection

CCU6 Input	T21CCU Output
T12HRD	**T21CCU_CCTCON**.CCTST
T13HRD	**T21CCU_CCTCON**.CCTST

Figure 166 shows all interrupt and interface signals and GPIO interface associated with the Mod_Name module kernel.

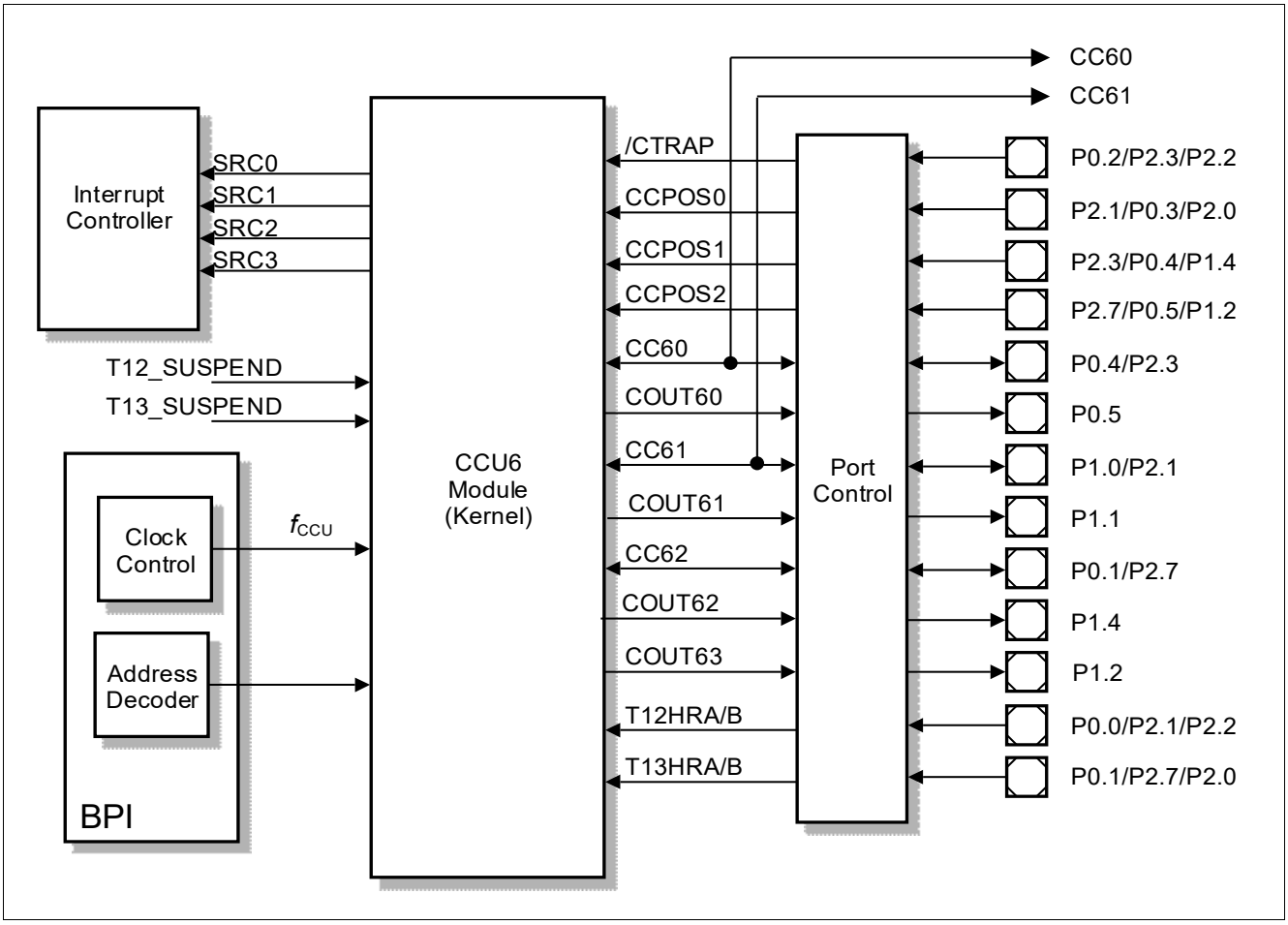


Figure 166 Interconnections of the CCU6 Module

19 UART1/2

19.1 Features

- Full-duplex asynchronous modes
  - 8-Bit or 9-Bit data frames, LSB first
  - fixed or variable baud rate
- Receive buffered (1 Byte)
- Multiprocessor communication
- Interrupt generation on the completion of a data transmission or reception
- Baud-rate generator with fractional divider for generating a wide range of baud rates, e.g. 9.6kBaud, 19.2kBaud, 115.2kBaud, 125kBaud, 250kBaud, 500kBaud
- Hardware logic for break and sync byte detection
- for UART1: LIN support: connected to timer channel for synchronization to LIN baud rate

In all modes, transmission is initiated by any instruction that uses SBUF as a destination register. Reception is initiated in the modes by the incoming start bit if REN = 1.

The serial interface also provides interrupt requests when transmission or reception of the frames has been completed. The corresponding interrupt request flags are TI or RI, respectively. If the serial interrupt is not used (i.e., serial interrupt not enabled), TI and RI can also be used for polling the serial interface.

19.2 Introduction

The UART1/2 provide a full-duplex asynchronous receiver/transmitter, i.e., it can transmit and receive simultaneously. They are also receive-buffered, i.e., they can commence reception of a second byte before a previously received byte has been read from the receive register. However, if the first byte still has not been read by the time reception of the second byte is complete, the previous byte will be lost. The serial port receive and transmit registers are both accessed at Special Function Register (SFR) SBUF. Writing to SBUF loads the transmit register, and reading SBUF accesses a physically separate receive register.

19.2.1 Block Diagram

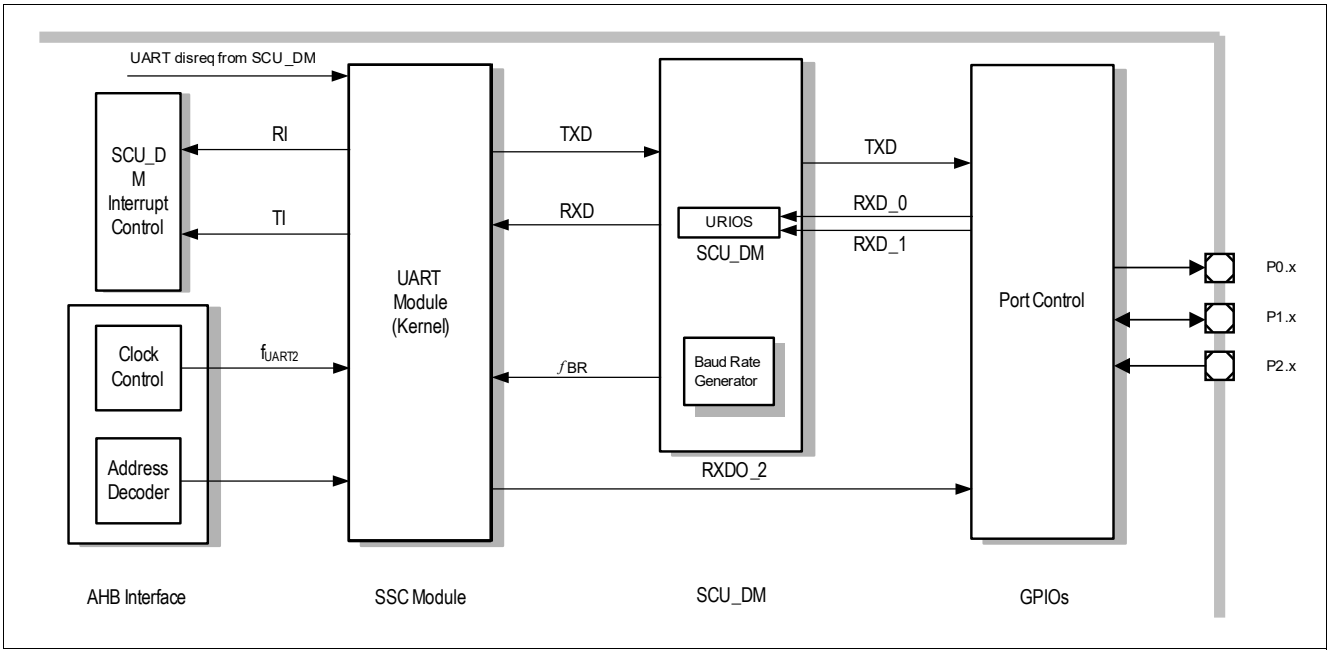


Figure 167 UART Block Diagram

19.3 UART Modes

The UART1/2 can be used in four different modes. In mode 0, it operates as an 8-Bit shift register. In mode 1, it operates as an 8-Bit serial port. In modes 2 and 3, it operates as a 9-Bit serial port. The only difference between mode 2 and mode 3 is the baud rate, which is fixed in mode 2 but variable in mode 3. The variable baud rate is set by the underflow rate on the dedicated baud-rate generator.

The different modes are selected by setting bits SM0 and SM1 to their corresponding values, as shown in **Table 346**.

Mode 1 example: 8 data bits, 1 start bit, 1 stop bit, no parity selection, 16 times oversampled, receive & transmit register double buffered, Tx/Rx IRQ(s).

Table 346 UART Modes

SM0	SM1	Operating Mode	Baud Rate
0	0	Mode 0: 8-Bit shift register	$f_{sys}/2$
0	1	Mode 1: 8-Bit shift UART	Variable
1	0	Mode 2: 9-Bit shift UART	$f_{sys}/64$ or $f_{sys}/32$
1	1	Mode 3: 9-Bit shift UART	Variable

19.3.1 Mode 0, 8-Bit Shift Register, Fixed Baud Rate

In mode 0, the serial port behaves as an 8-bit shift register. Data is shifted in through RXD, and out through RXDO, while the TXD line is used to provide a shift clock which can be used by external devices to clock data in and out.

The transmission cycle is activated by a write to SBUF. The data will be written to the transmit shift register with a 1 at the 9th bit position . For the next seven machine cycles, the contents of the transmit shift register are shifted right one position and a zero shifted in from the left so that when the MSB of the data byte is at the

output position, it has a 1 and a sequence of zeros to its left. The control block then executes one last shift before setting the TI bit.

Reception is started by the condition REN = 1 and RI = 0. At the start of the reception cycle, 11111110_B is written to the receive shift register. In each machine cycle that follows, the contents of the shift register are shifted left one position and the value sampled on the RXD line in the same machine cycle is shifted in from the right. When the 0 of the initial byte reaches the leftmost position, the control block executes one last shift, loads SBUF and sets the RI bit.

The baud rate for the transfer is fixed at $f_{sys}/2$ where f_{sys} is the input clock frequency, i.e. one bit per machine cycle.

19.3.2 Mode 1, 8-Bit UART, Variable Baud Rate

In mode 1, the UART behaves as an 8-bit serial port. A start bit (0), 8 data bits, and a stop bit (1) are transmitted on TXD or received on RXD at a variable baud rate.

The transmission cycle is activated by a write to SBUF. The data are transferred to the transmit shift register and a 1 is loaded to the 9th bit position (as in mode 0) . At phase 1 of the machine cycle after the next rollover in the divide-by-16 counter, the start bit is copied to TXD, and data is activated one bit time later. One bit time after the data is activated, the data starts getting shifted right with zeros shifted in from the left. When the MSB gets to the output position, the control block executes one last shift and sets the TI bit.

Reception is started by a high to low transition on RXD (sampled at 16 times the baud rate). The divide-by-16 counter is then reset and $1111\ 1111_B$ is written to the receive register. If a valid start bit (0) is then detected (based on two out of three samples), it is shifted into the register followed by 8 data bits. If the transition is not followed by a valid start bit, the controller goes back to looking for a high to low transition on RXD. When the start bit reaches the leftmost position, the control block executes one last shift, then loads SBUF with the 8 data bits, loads RB8 (SCON.2) with the stop bit, and sets the RI bit, provided RI = 0 (SCON.0), and either SM2 = 0 (SCON.5) (see **Section 19.4**) or the received stop bit = 1. If none of these conditions is met, the received byte is lost.

The associated timings for transmit/receive in mode 1 are illustrated in **Figure 168**.

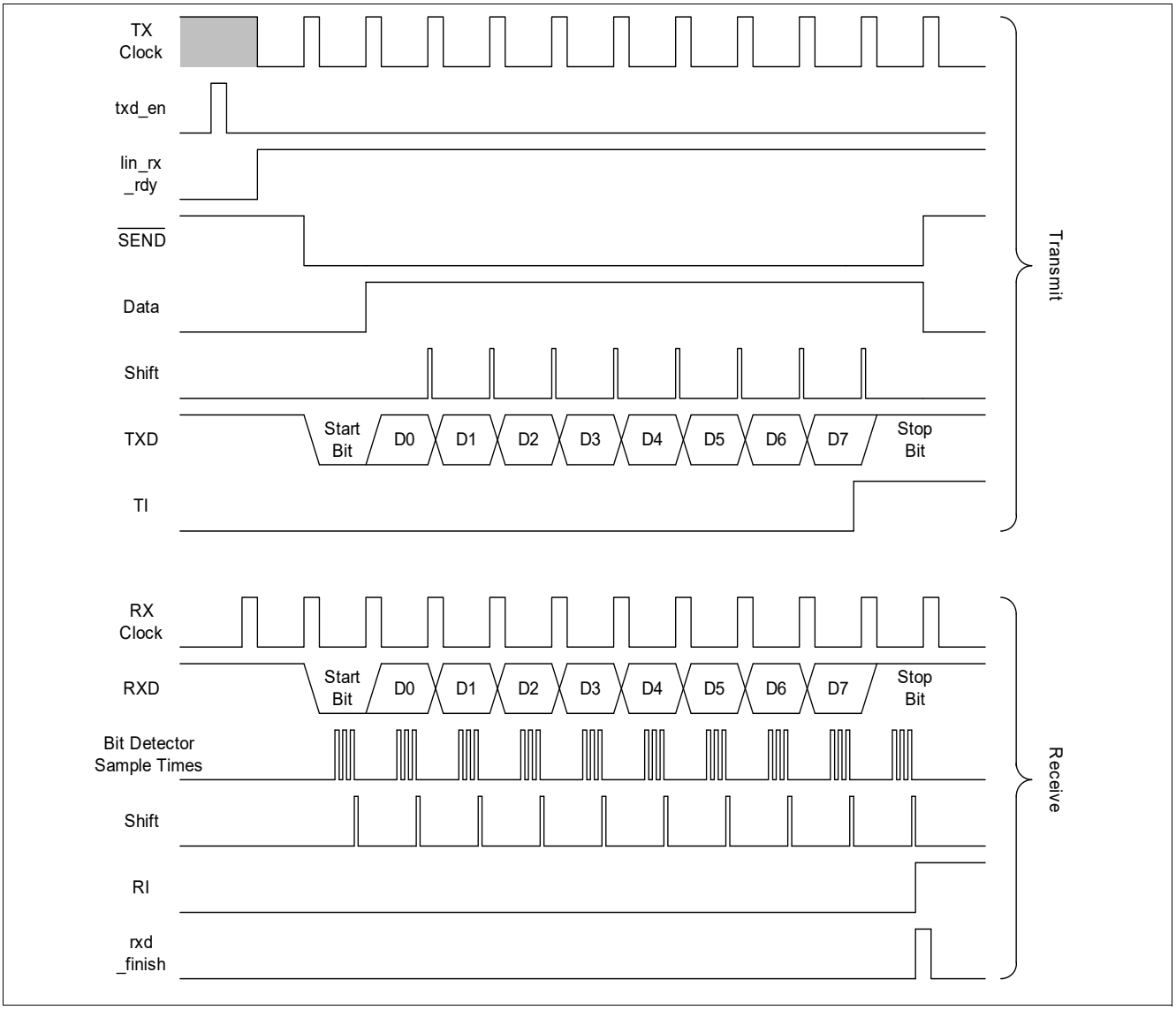


Figure 168 Serial Interface, Mode 1, Timing Diagram

19.3.3 Mode 2, 9-Bit UART, Fixed Baud Rate

In mode 2, the UART behaves as a 9-bit serial port. A start bit (0), 8 data bits plus a programmable 9th bit and a stop bit (1) are transmitted on TXD or received on RXD. The 9th bit for transmission is taken from TB8 (SCON.3) while for reception, the 9th bit received is placed in RB8 (SCON.2).

The transmission cycle is activated by a write to SBUF. The data is transferred to the transmit shift register and TB8 is copied into the 9th bit position. At phase 1 of the machine cycle following the next rollover in the divide-by-16 counter, the start bit is copied to TXD and data is activated one bit time later. One bit time after the data is activated, the data starts shifting right. For the first shift, a stop bit (1) is shifted in from the left and for subsequent shifts, zeros are shifted in. When the TB8 bit gets to the output position, the control block executes one last shift and sets the TI bit.

Reception is started by a high to low transition on RXD (sampled at 16 times of the baud rate). The divide-by-16 counter is then reset and 1111 1111_B is written to the receive register. If a valid start bit (0) is then detected (based on two out of three samples), it is shifted into the register followed by 8 data bits. If the transition is not followed by a valid start bit, the controller goes back to looking for a high to low transition on RXD. When the start bit reaches the leftmost position, the control block executes one last shift, then loads SBUF with the 8 data bits, loads RB8 (SCON.2) with the 9th data bit, and sets the RI bit, provided RI = 0 (SCON.0), and either SM2 = 0 (SCON.5) (see **Section 19.4**) or the 9th bit = 1. If none of these conditions is met, the received byte is lost.

The baud rate for the transfer is fixed at $f_{sys}/64$ or $f_{sys}/32$.

19.3.4 Mode 3, 9-Bit UART, Variable Baud Rate

Mode 3 is the same as mode 2 in all respects except that the baud rate is variable.

The associated timings for transmit/receive in modes 2 and 3 are illustrated in **Figure 169**.

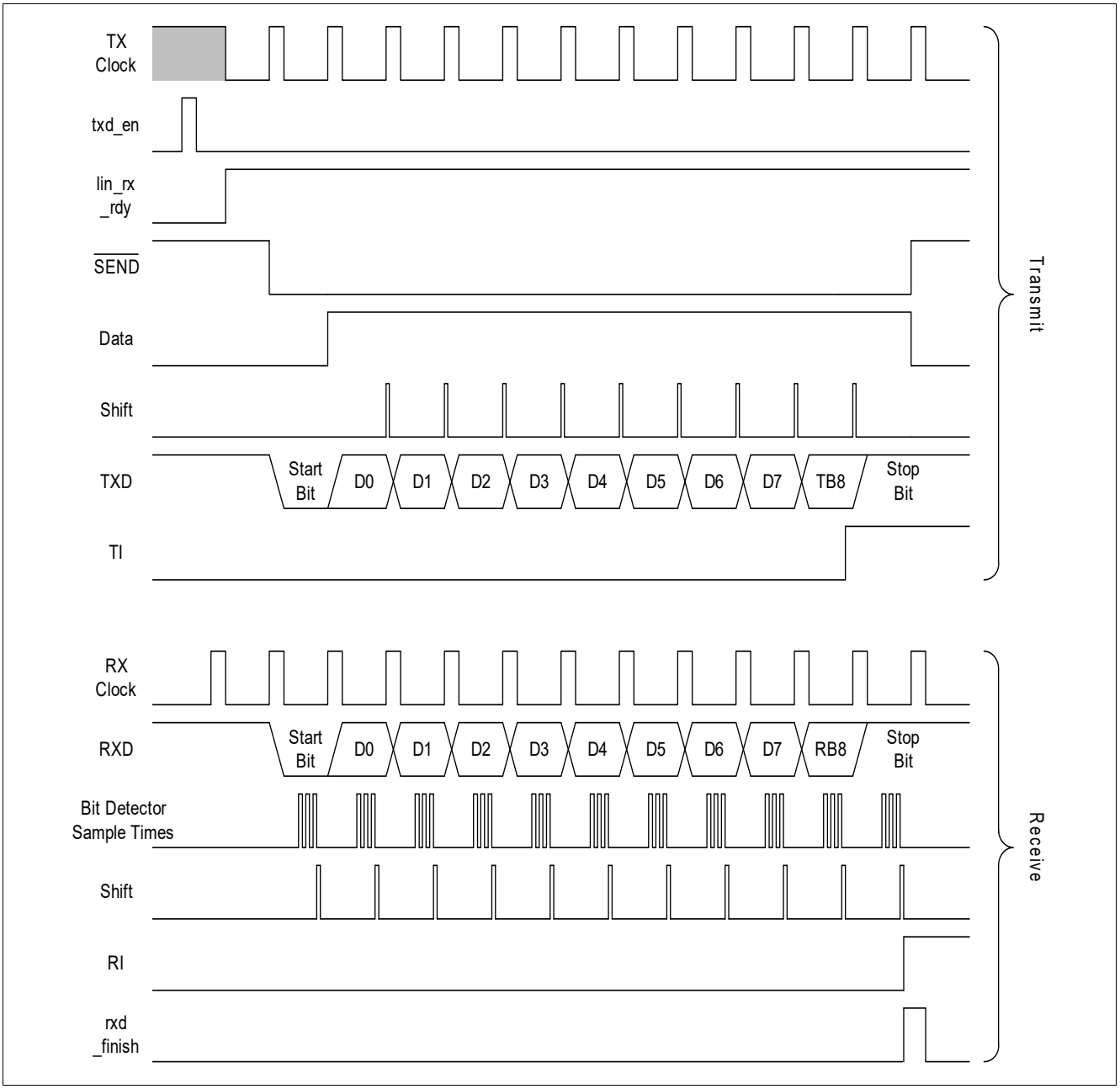


Figure 169 Serial Interface, Modes 2 and 3, Timing Diagram

19.4 Multiprocessor Communication

Modes 2 and 3 have a special provision for multiprocessor communication using a system of address bytes with bit 9 = 1 and data bytes with bit 9 = 0. In these modes, 9 data bits are received. The 9th data bit goes into RB8 (SCON.2). The communication always ends with one stop bit. The port can be programmed such that when the stop bit is received, the serial port interrupt will be activated only if RB8 = 1.

This feature is enabled by setting bit SM2 in register SCON. One of the ways to use this feature in multiprocessor systems is described in the following paragraph.

When the master processor wants to transmit a block of data to one of several slaves, it first sends out an address byte that identifies the target slave. An address byte differs from a data byte in the 9th bit. The 9th bit in an address byte is 1 and in a data byte the 9th bit is 0. With SM2 = 1, no slave will be interrupted by a data byte. An address byte, however, will interrupt all slaves, so that each slave can examine the received byte and see if it is being addressed. The addressed slave will clear its SM2 bit and prepare to receive the data bytes that will be coming. The slaves that were not being addressed retain their SM2 bits as set and ignore the incoming data bytes.

Note: Bit SM2 has no effect in mode 0. SM2 can be used in mode 1 to check the validity of the stop bit. In a mode 1 reception, if SM2 = 1, the receive interrupt will not be activated unless a valid stop bit is received.

19.5 Interrupts

The two UART interrupts can be separately enabled or disabled by setting or clearing their corresponding enable bits in SCU SFR MODIEN. An overview of the UART interrupt sources is shown in **Table 347**.

Table 347 UART Interrupt Sources

Interrupt	Flag	Interrupt Enable Bit
Reception completed	SCON.RI	SCU_MODIEN.RIEN
Transmission completed	SCON.TI	SCU_MODIEN.TIEN

19.6 Baud Rate Generation

There are several ways to generate the baud rate clock for the serial port, depending on the mode in which they are operating.

The baud rates in modes 0 and 2 are fixed to $f_{sys}/2$ and $f_{sys}/64$ respectively, while the variable baud rate in modes 1 and 3 is generated based on the setting of the baud-rate generator in SCU (see **Section 19.6.1**).

“Baud rate clock” and “baud rate” must be distinguished from each other. The serial interface requires a clock rate that is 16 times the baud rate for internal synchronization. Therefore, the UART baud-rate generator must provide a “baud rate clock” to the serial interface where it is divided by 16 to obtain the actual “baud rate”. The abbreviation f_{sys} refers to the input clock frequency.

19.6.1 Baud-Rate Generator

The baud-rate generator in SCU is used to generate the variable baud rate for the UART in modes 1 and 3. It has programmable 11-bit reload value, 3-bit prescaler and 5-bit fractional divider.

The baud-rate generator clock is derived via a prescaler (f_{DIV}) from the input clock f_{sys}. The baud rate timer counts downwards and can be started or stopped through the baud rate control run bit BCON.R. Each underflow of the timer provides one clock pulse to the serial channel. The timer is reloaded with the 11-bit BR_VALUE stored in its reload register BGL/BGH each time it underflows. The duration between underflows depends on the ‘n’ value in the fractional divider, which can be selected by the bits BGL.FD_SEL. ‘n’ times out of 32, the timer counts one cycle more than specified by BR_VALUE. The prescaler is selected by the bits BCON.BRPRE.

Register BGL/BH is the dual-function baud-rate Generator/Reload register. Reading BGL/GBH returns the contents of the timer, while writing to BGL (low byte) always updates the reload register.

The register BGL/BGH should be written only when BCON.R is 0. An auto-reload of the timer with the contents of the reload register is performed one instruction cycle after the next time BCON.R is set. Any write to BGL/GBH, while BCON.R is set, will be ignored.

The baud rate of the baud-rate generator depends on the following bits and register values:

- Input clock f_{sys}
- Value of bit field BCON.BRPRE.
- Value of bit field BGL.FD_SEL
- Value of the 11-bit reload value BGL/BGH.BR_VALUE

Figure 170 shows a simplified block diagram of the baud-rate generator.

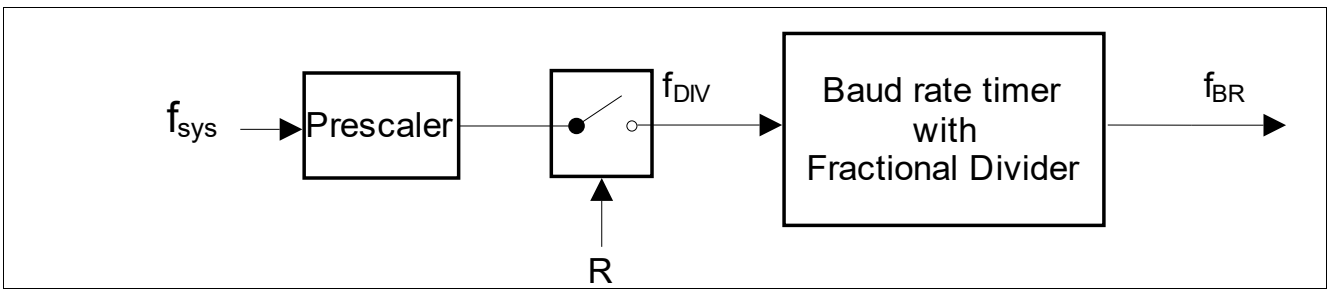


Figure 170 Simplified Baud-Rate Generator Block Diagram

The following formula calculate the final baud rate.

$$\text{Baud rate} = \frac{f_{sys}}{16 \bullet \text{PRE} \bullet (\text{BR\_VALUE} + \frac{n}{32})} \quad (19.1)$$

The value of PRE (prescaler) is chosen by the bit field BCON.BRPRE. BR_VALUE represents the contents of the reload value, taken as unsigned 11-bit integer from the bit field BGL/BGH.BR_VALUE. n/32 is defined by the fractional divider selection in bit field BGL.FDSEL.

The maximum baud rate that can be generated is limited to $f_{sys}/32$. Hence, for module clocks of 40 MHz and 24 MHz, the maximum achievable baud rate is 1.25 MBaud and 0.75 MBaud respectively.

Table 348 and **Table 349** list various commonly used baud rates together with their corresponding parameter settings and the deviation errors compared to the intended baud rate.

Table 348 Typical Baud Rates of UART (f_{sys} = 40 MHz)

Baud Rate (f_{sys} = 40 MHz)	PRE	Reload Value (BR_VALUE)	Numerator of Fractional Value (FD_SEL)	BG Register[1)]	Deviation Error
250.4 kBaud	1 (BRPRE = 000)	9(9_H)	31 ($1F_H$)	$0xxx_H$	+0.12%
115.2 kBaud	1 (BRPRE = 000)	21 (15_H)	22 (16_H)	$02B6_H$	+0.06%
20 kBaud	1 (BRPRE = 000)	125 ($7D_H$)	0 (0_H)	$0FA0_H$	0.00%
19.2 kBaud	1 (BRPRE = 000)	130 (82_H)	7 (7_H)	1047_H	-0.01%
9600 Baud	2 (BRPRE = 001)	130 (82_H)	7 (7_H)	1047_H	-0.01%
4800 Baud	4 (BRPRE = 010)	130 (82_H)	7 (7_H)	1047_H	-0.01%
2400 Baud	8 (BRPRE = 011)	130 (82_H)	7 (7_H)	1047_H	-0.01%

1) The value of the 16-bit BG register is obtained by concatenation the 11-bit BRVALUE and 5-bit FD_SEL into a 16-bit value.

Table 349 Typical Baud Rates of UART (f_{sys} = 24 MHz)

Baud Rate (f_{sys} = 24 MHz)	PRE	Reload Value (BR_VALUE)	Numerator of Fractional Value (FD_SEL)	BG Register[1)]	Deviation Error
115.2 kBaud	1 (BRPRE = 000)	13 ($0D_H$)	1 (01_H)	$01A1_H$	-0.08%
20 kBaud	1 (BRPRE = 000)	75 ($4B_H$)	0 (00_H)	0960_H	+0.00%
19.2 kBaud	1 (BRPRE = 000)	78 ($4E_H$)	4 (04_H)	$09C4_H$	+0.00%
9600 Baud	2 (BRPRE = 001)	78 ($4E_H$)	4 (04_H)	$09C4_H$	+0.00%
4800 Baud	4 (BRPRE = 010)	78 ($4E_H$)	4 (04_H)	$09C4_H$	+0.00%
2400 Baud	8 (BRPRE = 011)	78 ($4E_H$)	4 (04_H)	$09C4_H$	+0.00%

1) The value of the 16-bit BG register is obtained by concatenation the 11-bit BRVALUE and 5-bit FD_SEL into a 16-bit value.

19.7 LIN Support in UART

The UART module can be used to support the Local Interconnect Network (LIN) protocol for both master and slave operations. The LIN baud rate detection feature, which consists of the hardware logic for Break and Synch Byte detection, provides the capability to detect the baud rate within LIN protocol using Timer 2. This allows the UART module to be synchronized to the LIN baud rate for data transmission and reception.

19.7.1 LIN Protocol

LIN is a holistic communication concept for local interconnected networks in vehicles. The communication is based on the SCI (UART) data format, a single-master/multiple-slave concept, a clock synchronization for nodes without stabilized time base. An attractive feature of LIN is the self-synchronization of the slave nodes without a crystal or ceramic resonator, which significantly reduces the cost of hardware platform. Hence, the baud rate must be calculated and returned with every message frame.

The structure of a LIN frame is shown in **Figure 171**. The frame consists of the:

- header, which comprises a Sync Break (13-bit time low), Synch Byte (55_H), and ID field
- response time
- data bytes (according to UART protocol)
- checksum

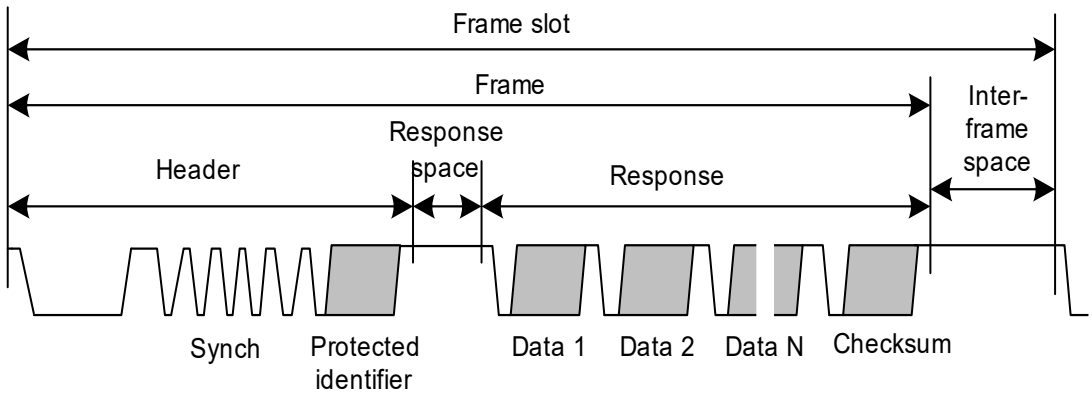


Figure 171 The Structure of LIN Frame

Each byte field is transmitted as a serial byte, as shown in **Figure 172**. The LSB of the data is sent first and the MSB is sent last. The start bit is encoded as a bit with value zero (dominant) and the stop bit is encoded as a bit with value one (recessive).

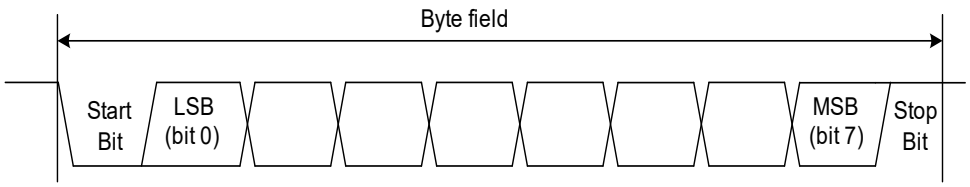


Figure 172 The Structure of Byte Field

The Sync Break is used to signal the beginning of a new frame. It is the only field that does not comply with **Figure 172**. A Sync Break is always generated by the master task (in the Master Mode) and it must be at least

13 bits of dominant value, including the start bit, followed by a Sync Break Delimiter, as shown in **Figure 173**. The Sync Break Delimiter will be at least one nominal bit time long.

A slave node will use a Sync Break detection threshold of 11 nominal bit times.

Figure 173 The Sync Break Field

The Synch Byte is a specific pattern for the determination of the time base. The Sync Byte field consists of the data value 55_H, as shown in **Figure 174**.

A slave task is always able to detect the Sync Break/Synch sequence, even if it expects a byte field (assuming the byte fields are separated from each other). If this happens, detection of the Sync Break/Synch sequence will abort the transfer in progress and processing of the new frame will commence.

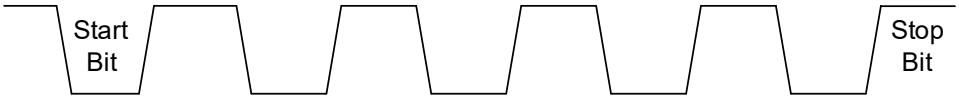


Figure 174 The Synch Byte Field

The slave task will receive and transmit data when an appropriate ID is sent by the master:

1. The slave waits for the Synch Break
2. The slave synchronizes on the Synch Byte
3. The slave snoops for the ID
4. According to the ID, the slave determines whether to receive or transmit data, or do nothing
5. When transmitting, the slave sends 2, 4 or 8 data bytes, followed by a Check Byte

19.7.2 LIN Header Transmission

LIN header transmission is only applicable in Master Mode. In the LIN communication, a master task decides when and which frame is to be transferred on the bus. It also identifies a slave task to provide the data transported by each frame. The information needed for the handshaking between the master and slave, tasks is provided by the master task through the header part of the frame.

The header consists of a Sync Break and Sync Byte pattern followed by an identifier. Among these three fields, only the Sync Break pattern cannot be transmitted as a normal 8-bit UART data. The Sync Break must contain a dominant value of 13 bits or more to ensure proper synchronization of slave nodes.

In the LIN communication, a slave task is required to be synchronized at the beginning of the protected identifier field of the frame. For this purpose, every frame starts with a sequence consisting of a Sync Break followed by a Synch Byte field. This sequence is unique and provides enough information for any slave task to detect the beginning of a new frame and to be synchronized at the start of the identifier field.

19.7.3 Automatic Synchronization to the Host

Upon entering LIN communication, a connection is established and the transfer speed (baud rate) of the serial communication partner (host) is automatically synchronized in the following steps that are to be included in the user software:

STEP 1: Initialize interface for reception and timer for baud rate measurement

STEP 2: Wait for an incoming LIN frame from host

STEP 3: Synchronize the baud rate to the host

STEP 4: Enter for Master Request Frame or for Slave Response Frame

The next sections, **Section 19.7.4**, **Section 19.7.5** and **Section 19.7.6** provide some hints on setting up the microcontroller for baud rate detection of LIN.

*Note: Re-synchronization and setup of the baud rate has always to be done for **every** Master Request Header or Slave Response Header LIN frame by user software.*

19.7.4 Initialization of Break/Synch Field Detection Logic

The LIN baud rate detection feature provides the capability to detect the baud rate within the LIN protocol using Timer 2. Initialization consists of:

- Setting of the serial port of the microcontroller to Mode 1 (8-bit UART, variable baud rate) for communication.
- Providing the baud rate range via bit field SCU_LINST.BGSEL.
- Toggling of the SCU_LINST.BRDIS bit (set the bit to 1 before clearing it back to 0) to initialize the Sync Break/Synch detection logic.
- Clearing all status flags SCU_LINST.BRK, SCU_LINST.EOFSYN and SCU_LINST.ERRSYN to 0.
- Setting of Timer 2 to capture mode with falling edge trigger at pin T2EX. Setting of the bits T2MOD.EDGESEL to 0 by default and T2CON.CP/$\overline{RL2}$ to 1.
- Enabling Timer 2 external events. T2CON. EXEN2 is set to 1. (EXF2 flag is set when a negative transition occurs at pin T2EX)
- *Configuring of* f_{T2} by bit field T2MOD.T2PRE.

19.7.5 Baud-Rate Range Selection

The Sync Break/Synch Field detection logic supports a maximum number of bits in the Sync Break field as defined by **Equation (19.2)**.

$$\text{Maximum number of bits} = \text{Baud rate} \bullet \frac{4095}{\text{Sample Frequency}} \tag{19.2}$$

The sample frequency is given by **Equation (19.3)**.

$$\text{Sample Frequency} = \frac{fsys}{8 \bullet 2^{BGSEL}} \tag{19.3}$$

If the maximum number of bits in the Break field is exceeded, the internal counter will overflow, which results in a baud rate detection error. Therefore, an appropriate SCU_LINST.BGSEL value has to be selected for the required baud rate detection range.

The baud rate range defined by different SCU_LINST.BGSEL settings is shown in **Table 350**.

Table 350 BGSEL Bit Field Definition for Different Input Frequencies

f_{sys}	BGSEL	Baud Rate Select for Detection $f_{sys}/(2184*2^{BGSEL})$ to $f_{sys}/(72*2^{BGSEL})$
40 MHz	00_B	18.3 kHz to 555.6 kHz
	01_B	9.2 kHz to 277.8 kHz
	10_B	4.6 kHz to 138.9 kHz
	11_B	2.3 kHz to 69.4 kHz
24 MHz	00_B	11 kHz to 333.3 kHz
	01_B	5.5 kHz to 166.7 kHz
	10_B	2.8 kHz to 83.3 kHz
	11_B	1.4 kHz to 41.7 kHz

Each BGSEL setting supports a range of baud rate for detection. If the baud rate used is outside the defined range, the baud rate may not be detected correctly.

When f_{sys} = 40 MHz, the baud rate range between 18.3 kHz to 555.6 kHz can be detected. The following examples serve as a guide to select the BGSEL value:

- If the baud rate falls in the range of 2.3 kHz to 4.6 kHz, selected BGSEL value is "11_B".
- If the baud rate falls in the range of 4.6 kHz to 9.2 kHz, selected BGSEL value is "10_B".
- If the baud rate falls in the range of 9.2 kHz to 18.3 kHz, selected BGSEL value is "01_B".
- If the baud rate falls in the range of 18.3 kHz to 555.6 kHz, selected BGSEL value is "00_B".
- If the baud rate is 20 kHz, the possible values of BGSEL that can be selected are "00_B", "01_B", "10_B", and "11_B". However, it is advisable to select "00_B" for better detection accuracy.

The baud rate can also be detected when f_{sys} = 24 MHz, for which the baud rate range that can be detected is between 1.4 kHz to 333.3 kHz.

19.7.6 LIN Baud Rate Detection

The baud rate detection for LIN is shown in **Figure 175**, the Header LIN frame consists of the:

- Sync Break (13 bit times low)
- Sync Byte (55_H)
- Protected ID field

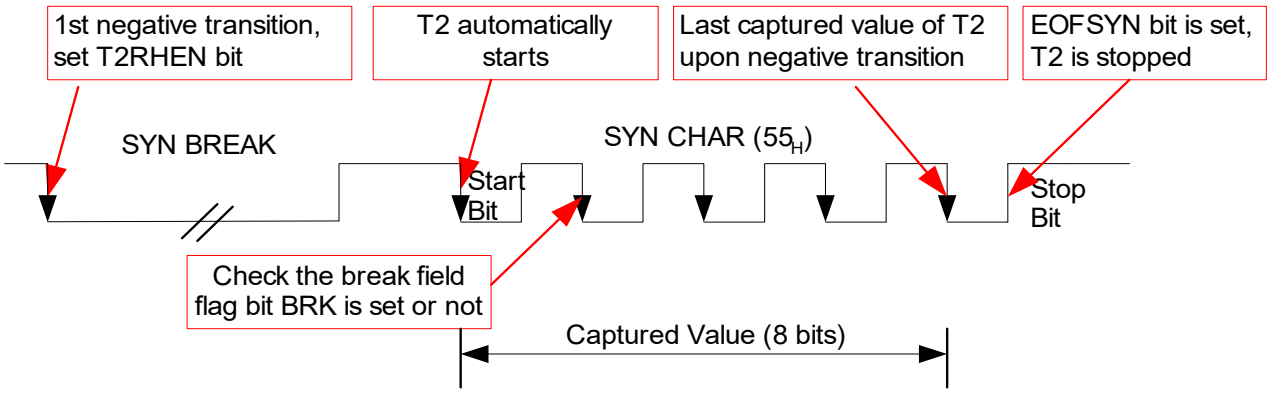


Figure 175 LIN Auto Baud Rate Detection

With the first falling edge:

- The Timer 2 External Start Enable bit (T2MOD.T2RHEN) is set. The falling edge at pin T2EX is selected by default for Timer 2 External Start (bit T2MOD.T2REGS is 0).

With the second falling edge:

- Start Timer 2 by the hardware.

With the third falling edge:

- Timer 2 captures the timing of 2 bits of SYN byte.
- Check the Break Field Flag bit SCU_LINST.BRK.

If the Sync Break Field Flag SCU_LINST.BRK is set, software may continue to capture 4/6/8 bits of Sync Byte. Finally, the End of Sync Byte Flag (SCU_LINST.EOFSYN) is set, Timer 2 is stopped. T2 Reload/Capture register (RC2H/L) is the time taken for 2/4/6/8 bits according to the implementation. Then the LIN routine calculates the actual baud rate, sets the BRPRE and BGL/BGH values if the UART module uses the baud-rate generator for baud rate generation.

After the third falling edge, the software may discard the current operation and continue to detect the next header LIN frame if the following conditions were detected:

- The Sync Break Field Flag SCU_LINST.BRK is not set, or
- The Sync Byte Error Flag SCU_LINST.ERRSYN is set

19.8 Register Description

Table 351 Register Address Space

Module	Base Address	End Address	Note
UART1	4802 0000$_H$	4802 1FFF$_H$	UART1
UART2	4802 2000$_H$	4802 3FFF$_H$	UART2

Table 352 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
UART Registers,			
UART_SBUF	Serial Data Buffer	04$_H$	0000 0000$_H$
UART_SCON	Serial Channel Control Register	00$_H$	0000 0000$_H$
UART_SCONCLR	Serial Channel Control Clear Register	08$_H$	0000 0000$_H$

The registers are addressed wordwise.

19.8.1 UART Registers

UART uses the Special Function Registers (SFRs), SCON, SBUF, BCON, LINST, BGL and BGH. SCON is the control register and SBUF is the data register. On reset, both SCON and SBUF return 00$_H$. The serial port control and status register is the SFR SCON. This register contains not only the mode selection bits, but also the 9th data bit for transmit and receive (TB8 and RB8) and the serial port interrupt bits (TI and RI).

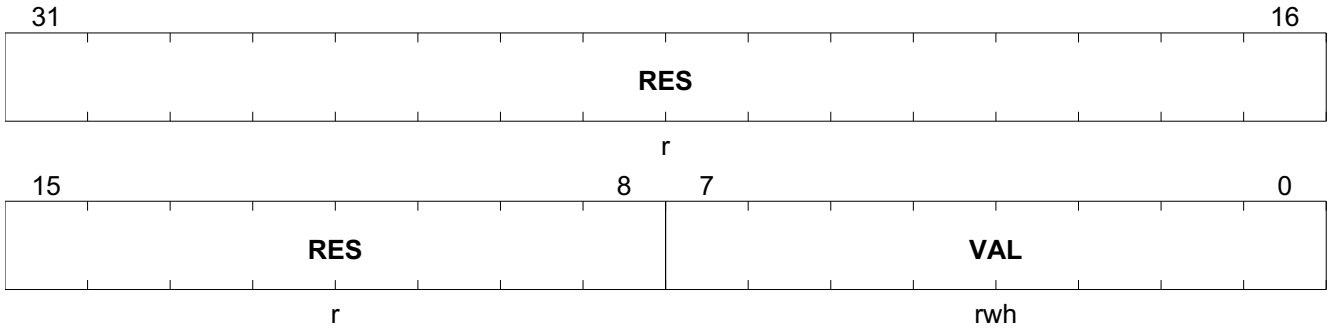
SBUF is the receive and transmit buffer of the serial interface. Writing to SBUF loads the transmit register and initiates transmission. This register is used for both transmit and receive data. Transmit data is written to this location and receive data is read from this location, but the two paths are independent.

Reading out SBUF accesses a physically separate receive register. The registers BCON, LINST, BGL and BGH are paged SFRs and are described in **Chapter 7.11.1** and **Chapter 7.12.1.1**.

UART1/2

Serial Data Buffer

UART_SBUF	Offset	Reset Value
Serial Data Buffer	04_H	see Table 353



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
VAL	7:0	rwh	**Serial Interface Buffer Register**

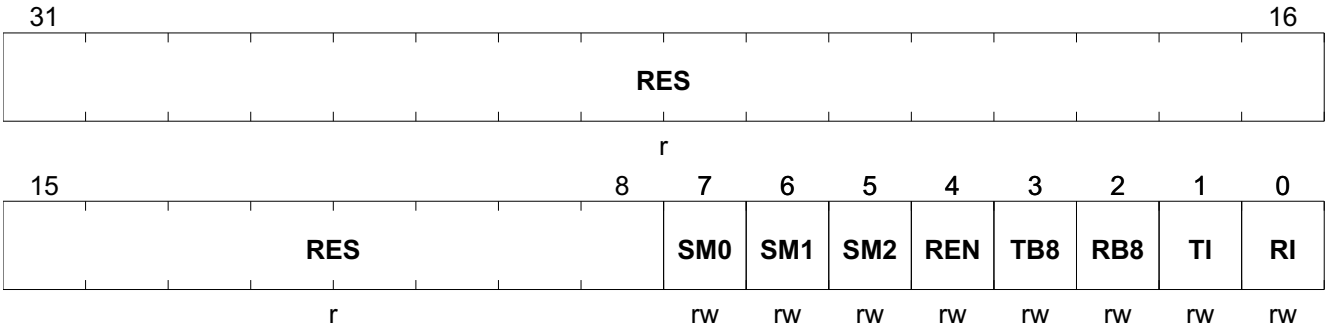
Table 353 Reset of UART_SBUF

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

UART1/2

Serial Channel Control Register

UART_SCON	Offset	Reset Value
Serial Channel Control Register	00_H	see Table 354



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Returns 0 if read; should be written with 0.
SM0	7	rw	**Serial Port Operating Mode Selection** see Table 346
SM1	6	rw	**Serial Port Operating Mode Selection** see Table 346
SM2	5	rw	**Enable Serial Port Multiprocessor Communication in Modes 2 and 3** Mode 2 or 3: - if SM2 = 1: RI will not be activated if the received 9th data bit (RB8) is 0. Mode 1: - if SM2 = 1: RI will not be activated if no valid stop bit (RB8) was received. Mode 0: - SM2 should be 0.
REN	4	rw	**Enable Receiver of Serial Port** 0_B **Disable**, Serial reception is disabled. 1_B **Enable**, Serial reception is enabled.
TB8	3	rw	**Serial Port Transmitter Bit 9** In modes 2 and 3, this is the 9th data bit sent. In mode 1, this bit is set to 1 In mode 0, this bit is set to 1
RB8	2	rw	**Serial Port Receiver Bit 9** In modes 2 and 3, this is the 9th data bit received. In mode 1, this is the stop bit received. In mode 0, this bit is not used. Must be cleared by flag SCONCLR.RB8CLR. This flag can also be set by software.

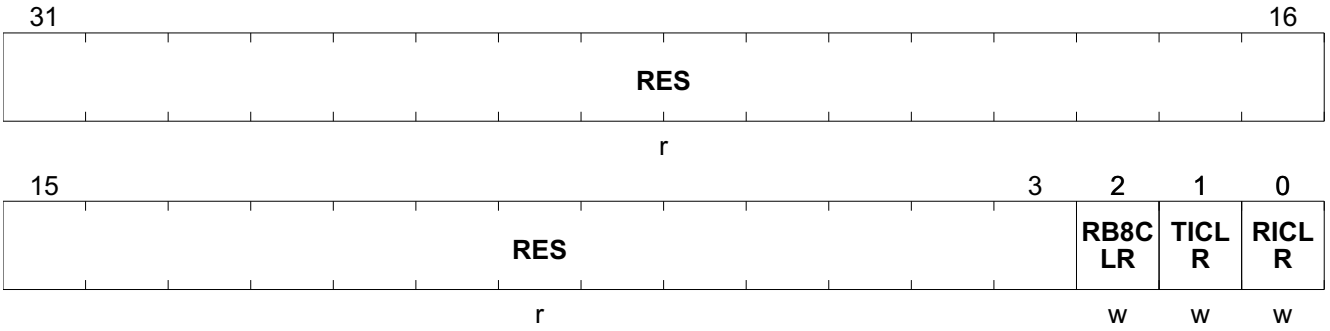
Field	Bits	Type	Description
TI	1	rw	**Transmit Interrupt Flag** This is set by hardware at the end of the 8th bit in mode 0, or at the beginning of the stop bit in modes 1, 2, and 3. Must be cleared by flag SCONCLR.TICLR. This flag can also be set by software.
RI	0	rw	**Receive Interrupt Flag** This is set by hardware at the end of the 8th bit on mode 0, or at the half point of the stop bit in modes 1, 2, and 3. Must be cleared by flag SCONCLR.RICLR. This flag can also be set by software.

Table 354 Reset of UART_SCON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Serial Channel Control Clear Register

UART_SCONCLR	Offset	Reset Value
Serial Channel Control Clear Register	08_H	see Table 355



Field	Bits	Type	Description
RES	31:3	r	**Reserved** Returns 0 if read; should be written with 0.
RB8CLR	2	w	**SCON.RB8 Clear Flag**[1)] 0_B **No Clear**, RB8 Flag is not cleared. 1_B **Clear**, RB8 Flag is cleared.
TICLR	1	w	**SCON.TI Clear Flag**[1)] 0_B **No Clear**, TI Flag is not cleared. 1_B **Clear**, TI Flag is cleared.
RICLR	0	w	**SCON.RI Clear Flag**[1)] 0_B **No Clear**, RI Flag is not cleared. 1_B **Clear**, RI Flag is cleared.

1) Flag is always read as 0

Table 355 Reset of UART_SCONCLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

19.8.2 Baud-Rate Generator Control and Status Registers

The UART module is also used to support LIN communication. For this purpose the UART is equipped with a LIN Break recognition, Sync Byte detector and special baud-rate generator including a fractional divider. The control registers for this support hardware are located in the SCU_DM module. The figure below sketches the structure of the complete UART - LIN support hardware.

19.9 Interfaces of the UART Module Mod_Name

An overview of the Mod_Name I/O interface is shown in **Figure 176**.

In mode 0 (the serial port behaves as a shift register) data is shifted in through RXD_1 and out through RXDO, while the TXD_1 line is used to provide a shift clock which can be used by external devices to clock data in and out. In modes 1, 2 and 3, the port behaves as a UART. Data is transmitted on TXD and received on RXD.

Data that is shifted into and out of the UART through RXD and TXD respectively, can be selected from different sources. This selection is performed by the SCU via SFR-bit MODPISEL.URIOS.

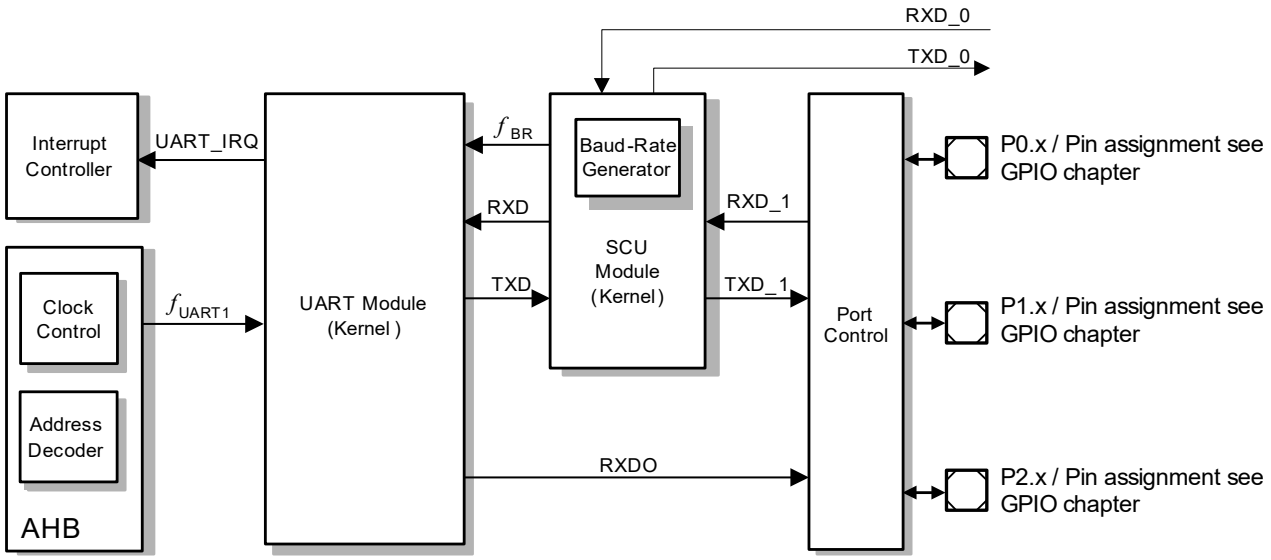


Figure 176 UART Module I/O Interface

20 LIN Transceiver

20.1 Features

General Functional Features

- Compliant to LIN2.2 Standard, backward compatible to LIN1.3, LIN2.0 and LIN 2.1
- Compliant to SAE J2602 (Slew Rate, Receiver hysteresis)

Special Features

- Measurement of LIN Master baudrate via Timer 2
- LIN can be used as Input/Output with SFR bits.
- TxD Timeout Feature (optional, on by default)
- Overcurrent limitation and overtemperature protection
- LIN module fully resettable via global enable bit

Operation Modes Features

- LIN Sleep Mode (LSLM)
- LIN Receive-Only Mode (LROM)
- LIN Normal Mode (LNM)
- High Voltage Input / Output Mode (LHVIO)

Slope Modes Features

- Normal Slope Mode (20 kbit/s)
- Low Slope Mode (10.4 kbit/s)
- Fast Slope Mode (62.5 kbit/s)
- Flash Mode (115 kbit/s, 250 kbit/s)

Wake-Up Features

- LIN Bus wake-up

20.2 Introduction

The LIN Module is a transceiver for the Local Interconnect Network (LIN) compliant to the LIN2.2 Standard, backward compatible to LIN1.3, LIN2.0 and LIN2.1. It operates as a bus driver between the protocol controller and the physical network. The LIN bus is a single wire, bi-directional bus typically used for in-vehicle networks, using baud rates between 2.4 kBaud and 20 kBaud. Additionally baud rates up to 62.5 kBaud are implemented.

The LIN Module offers several different operation modes, including a LIN Sleep Mode and the LIN Normal Mode. The integrated slope control allows to use several data transmission rates with optimized EMC performance. For data transfer at the end of line, a Flash Mode up to 115 kBaud is implemented. This Flash Mode can be used for data transfer under special conditions for up to 250 kbit/s (in production environment, point-to-point communication with reduced wire length and limited supply voltage).

20.2.1 Block Diagram

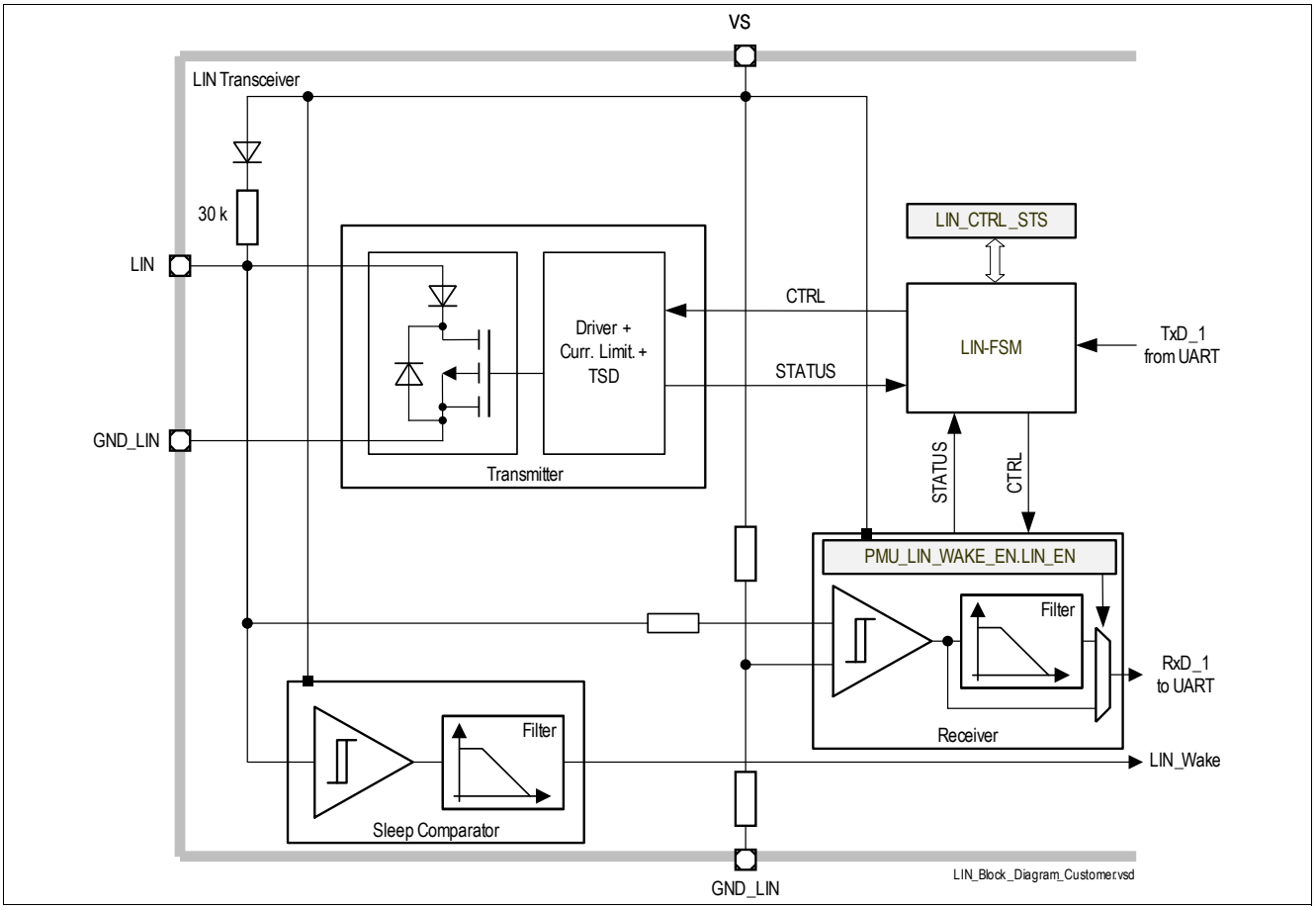


Figure 177 LIN Transceiver Block Diagram

20.3 Functional Description

The supported baud rates are:

- Low Slope Mode for a transmission up to 10.4 kBaud
- Normal Slope Mode for a transmission up to 20 kBaud
- Fast Slope Mode for a transmission up to 40 kBaud
- Flash Mode for a transmission up to 115 kBaud

20.3.1 LIN Normal Mode

The LIN Module is controlled by an internal state machine which determines the actual state of the transceiver. This state machine is controllable by the SFR interface.

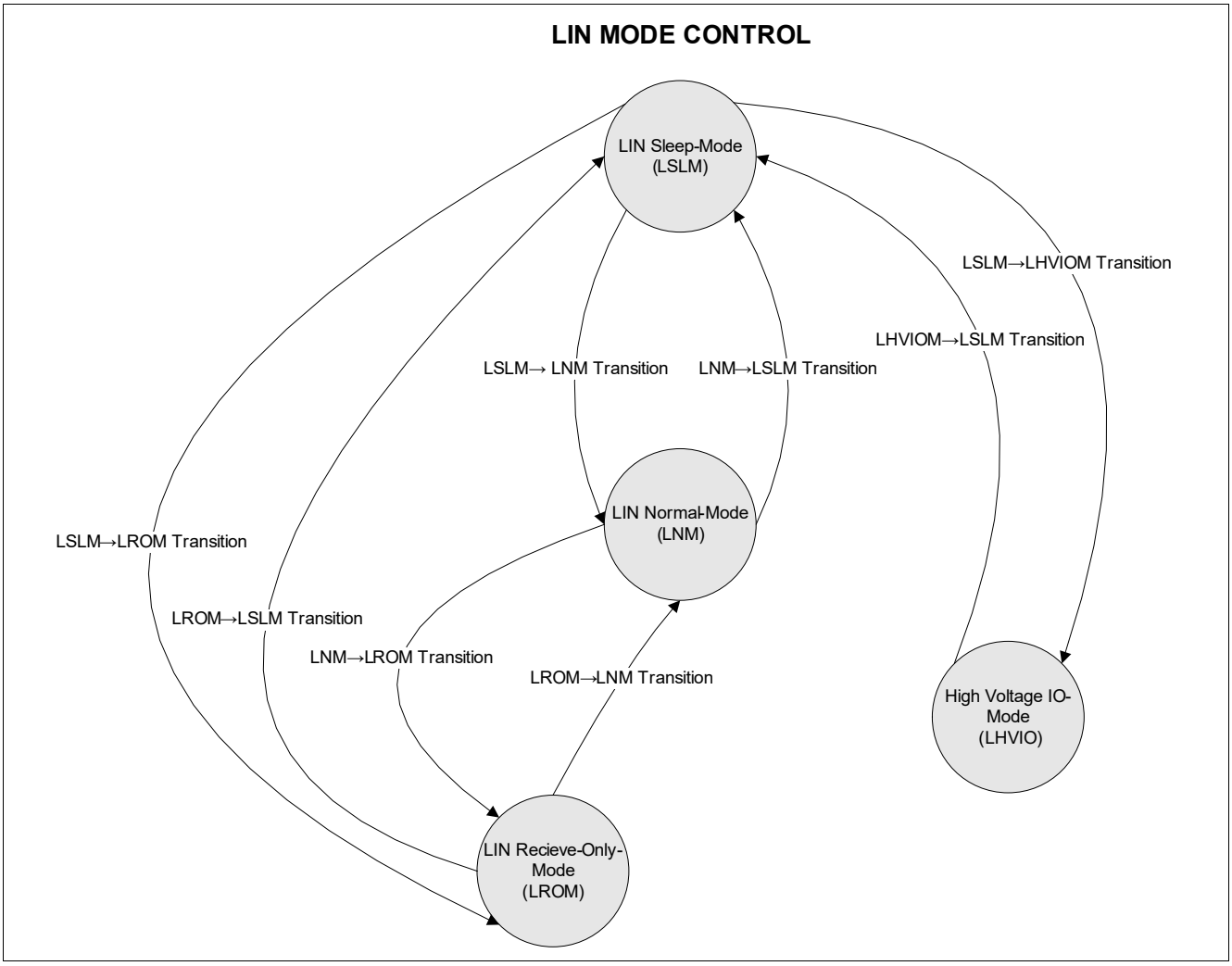


Figure 178 SFR controlled LIN Transceiver State machine

LIN Normal Mode (LNM)

In this mode it is possible to receive and transmit data with low slope, normal slope, fast slope or flash mode. Slope Setting is locked during LIN Normal Mode to avoid destruction of Communication Process. This is blocked by hardware.

LIN Receive-Only Mode (LROM)

In LIN Receive-Only Mode the transmitter is disabled. The receiver is active. This mode can be directly selected by application software or is automatically set upon error detection.

LIN Sleep Mode (LSLM)

In this mode, the transmit and receive functions are disabled, the wake receiver is active. Minimum current consumption is achieved. Wake up via LIN is possible. To disable the wake capability via LIN, within the PMU the LIN wake can be disabled.

LIN High Voltage Input / Output (LHVIO)

This mode is dedicated for using the LIN Transceiver as high voltage input/output. In LHVIO Mode the transceiver can be controlled by 2 SFR bits, **LIN_CTRL**.TXD and **LIN_CTRL**.RXD.

The transitions between the described states can only be executed when corresponding conditions are fulfilled. The detailed description of the transitions can be found below.

LIN Sleep Mode (LSLM) - LIN Receive-Only Mode (LROM) Transition Description

- LSLM - LROM transition is executed when:
  - LIN_CTRL.MODE is configured to LIN Receive-Only Mode and
  - Feedback Signals of Mode and Slope Mode are ok and
  - HV-Mode bit is not set
- LROM - LSLM transition is executed when:
  - LIN_CTRL.MODE is configured to LIN Sleep Mode

LIN Sleep Mode (LSLM) - LIN Normal Mode (LNM) Transition Description

- LSLM - LNM transition is executed when:
  - LIN_CTRL.MODE is configured to LIN Normal Mode and
  - Feedback Signals of Mode and Slope Mode are ok and
  - HV-Mode bit is not set and
  - VS-Undervoltage Flag is not set
  - LIN Transceiver LIN_CTRL.OT_STS and LIN_CTRL.OC_IS are not set and
  - no LIN_CTRL.TXD_TMOUT is set and
- LNM - LSLM transition is executed when:
  - LIN_CTRL.MODE is configured LIN Sleep Mode

LIN Normal Mode (LNM) - LIN Receive-Only Mode (LROM) Transition Description

- LNM - LROM transition is executed when
  - LIN_CTRL.MODE is configured to LIN Receive-Only Mode or
  - Feedback Signals of Mode and Slope Mode are not ok or
  - LIN_CTRL.OC_IS Flag is set or
  - VS-Undervoltage Flag is set or
  - LIN Transceiver LIN_CTRL.OT_STS or LIN_CTRL.OC_IS are set or
  - LIN_CTRL.TXD_TMOUT is set
- LROM - LNM transition is executed when:
  - LIN_CTRL.MODE is configured to LIN Normal Mode and
  - Feedback Signals of Mode and Slope Mode are ok and
  - LIN_CTRL.OC_STS Flag is not set and
  - VS-Undervoltage Flag is not set and
  - LIN Transceiver LIN_CTRL.OT_STS and LIN_CTRL.OC_IS are not set and
  - no LIN_CTRL.TXD_TMOUT is set

LIN Sleep Mode (LSLM) - LIN High Voltage Input / Output Mode (LHVIO) Transition Description

- LSLM - LHVIO transition is executed when
  - LIN_CTRL.HV_MODE flag is set and
  - LIN_CTRL.MODE is configured to LIN Normal Mode after LIN_CTRL.HV_MODE flag was set and

  - Feedback Signals of Mode and Slope Mode are ok and
  - LIN Transceiver LIN_CTRL.OT_STS and LIN_CTRL.OC_STS are not set
- LHVIO - LSLM transition is executed when:
  - LIN_CTRL.MODE is configured to LIN Sleep Mode and
  - LIN_CTRL.HV_MODE flag is set or
  - Feedback Signals of Mode and Slope Mode are not ok or
  - LIN Transceiver LIN_CTRL.OT_STS or LIN_CTRL.OC_STS are set

LIN Specifications 1.3 and 2.0, 2.1

The LIN specification 2.0 is a superset of the 1.3 version offering some additional features. However, it is possible to use the LIN 1.3 slave node in a 2.0 node cluster, as long as the new features are not used. Vice versa it is possible to use a LIN 2.0 node in the 1.3 cluster without using the new features.

The latest version of the LIN specification 2.1 has no changes regarding the physical layer specification of LIN 2.0.

20.3.2 LIN Transceiver Error Handling

The LIN Module provides error handling for three different cases:

LIN Transceiver TxD Timeout

If the internal UART TxD signal is dominant for the time $t > t_{timeout}$, the TxD timeout function deactivates the LIN transmitter output stage temporarily, by entering the LIN Receive-Only Mode. The transceiver remains in recessive state. The TxD timeout function prevents the LIN bus from being blocked by a permanent low signal on the TxD pin, caused by a failure. The failure is stored in the TXD_TMOUT flag. The transmitter stage is activated again after the dominant timeout condition is removed and after the TXD_TMOUT flag is cleared by software.

LIN Transmitter Overcurrent

If the LIN transmitter detects an overcurrent condition $I > I_{BUS,sc}$, the LIN transceiver enters LIN Receive Only Mode and the overcurrent status will be stored in the LIN_OC_STS flag. The short circuit current is limited to $I_{BUS,sc}$. The LIN_OC_IS flag can be cleared by software and will be set again as long as the above condition remains.

To generate an interrupt in case of LIN overcurrent detection, the corresponding interrupt can be enabled by setting the LIN_IRQEN.OC_IEN in the LIN_IRQEN register. This interrupt is routed to INTISR_10.

LIN Transmitter Overtemperature

If the LIN transmitter detects an overtemperature condition the transmitter will be deactivated temporarily, by entering the LIN Receive-Only Mode. The transceiver remains in recessive state. The failure is stored in the LIN_OT_IS flag. The transmitter stage is activated again after the overtemperature condition is gone and after the LIN_OT_STS flag is cleared by software.

To generate an interrupt in case of LIN overtemperature detection, the corresponding interrupt can be enabled by setting the LIN_IRQEN.OT_IEN in the LIN_IRQEN register. This interrupt is routed to INTISR_10.

20.3.3 Slope Modes

The LIN Module provides some additional slope mode features which can be used for EoL (End of Line) programming or to reduce emission in case of usage of lower baud rates. The configurable slope modes are:

Normal Slope Mode

This mode is usually used to transmit and receive messages on the bus. The selected slew rate setting allows a transmission rate of up to 20 kBaud.

Low Slope Mode

The usage of this mode is linked to a communication with lower baud rate. With this setting the emission of the transmitter can be reduced. The selected slew rate setting allows a transmission rate of up to 10.4 kBaud.

Fast Slope Mode

In this mode it is also possible to transmit and receive messages on the bus. The selected slew rate setting allows a transmission rate of up to 40 kBaud.

Flash Mode

In this mode it is possible to transmit and receive messages on the bus. Transmission rates of up to 115 kBaud are allowed due the internal slew rate control. This mode can be used for EoL programming.

Change of Slope modes

It is not possible to change the slope modes if the module is operating in LIN Normal Mode to avoid transmission errors. To change the slope mode for example from Normal Slope Mode to Flash Mode, it is necessary to change to LIN Receive-Only Mode or LIN Sleep Mode, configure the desired slope mode and to go back to LIN Normal Mode.

20.3.4 LIN Transceiver Status for Mode Selection

The LIN transceiver provides the possibility to monitor the on chip status through internally generated feedback signals. This provides additional protection functionality for the application to avoid wrong configuration of the transceiver, which may lead to a blocking of communication on the LIN Bus. The table below shows the decoding of feedback signals to check the current status of the transceiver.

Table 356 Decoding of Feedback Signals for LIN Transmitter Mode Settings

LIN_MODE_FB_<2>	LIN_MODE_FB<1>	LIN_MODE_FB<0>	Remarks
0	0	0	Mode Error or LIN module disabled.
0	0	1	LIN Sleep Mode
0	1	0	Mode Error
0	1	1	Mode Error
1	0	0	Mode Error
1	0	1	LIN Receive-Only Mode
1	1	0	Mode Error
1	1	1	LIN Normal Mode

A Mode Error indicates a problem in the LIN configuration. If that applies, check the LIN software configuration, and whenever this does not improve the feedback mode it is recommended to enter Sleep Mode.

20.3.5 LIN Transceiver Slope Mode Status

The LIN transceiver provides the possibility to monitor the on chip status of the slope control through internally generated feedback signals. The table shows the decoding of the feedback signals.

Table 357 Slope Mode Status

LIN_FB_SM3	LIN_FB_SM2	LIN_FB_SM1	Remarks
0	0	0	LIN module not enabled
0	0	1	Low Slope Mode
0	1	0	Normal Slope Mode
0	1	1	Fast Slope mode
1	0	0	Flash Mode
1	0	1	Slope Mode Error
1	1	0	Slope Mode Error
1	1	1	Slope Mode Error

20.4 Register Definition

Register Functionality according **Table 359**

Reset value according **Table 359**

Table 358 shows the module base addresses.

Table 358 Register Address Space

Module	Base Address	End Address	Note
LIN	$4801E000_H$	$4801FFFF_H$	

Table 359 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Register Definition,			
LIN_CTRL	LIN Transceiver Control	00_H	0000 0000 0001 1000 xxx0 0x10 $0000\ 0111_B$
LIN_IRQS	LIN Transceiver Interrupt Status	04_H	$0000\ 0000_H$
LIN_IRQCLR	LIN Transceiver Interrupt Status Register Clear	08_H	$0000\ 0000_H$
LIN_IRQEN	LIN Transceiver Interrupt Enable Register	$0C_H$	$0000\ 0000_H$

The registers are addressed wordwise.

The LIN-Transceiver and the controlling finite state machine can be fully controlled by the following SFR Register.

LIN Transceiver Control

LIN_CTRL	Offset	Reset Value
LIN Transceiver Control	00_H	see **Table 360**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res									Res	HV_MODE	Res		Res		
r									r	rw	r		r		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FB_SM3	FB_SM2	FB_SM1	SM		RXD	TXD	Res		MODE_FB			Res	MODE		EN
r	r	r	rw		r	rw	r		r			r	rw		rw

Field	Bits	Type	Description
Res	31:23	r	**Reserved** Always read as 0
Res	22	r	**Reserved** Always read as 1
HV_MODE	21	rw	**LIN Transceiver High Voltage Input - Output Mode** *Note: switching to HVIO-Mode (this configuration bit gets effective) is only possible when transceiver is in Sleep Mode.* 0_B **DISABLE**, High Voltage Mode Entry is disabled 1_B **ENABLE**, High Voltage Mode Entry is enabled
Res	20:19	r	**Reserved** Always read as “11”
Res	18:16	r	**Reserved** Always read as “00”
FB_SM3	15	r	**Feedback Signal 3 for Slope Mode Setting** Coding see **Table 357**
FB_SM2	14	r	**Feedback Signal 2 for Slope Mode Setting** Coding see **Table 357**
FB_SM1	13	r	**Feedback Signal 1 for Slope Mode Setting** Coding see **Table 357**
SM	12:11	rw	**LIN Transmitter Slope mode control** 00_B **Normal Slope Mode**, for max. 20 kBaud 01_B **Fast Slope Mode**, for max. 40 kBaud 10_B **Low Slope Mode**, for max. 10.4 kBaud 11_B **Flash Mode**, for max. 150 $kBaud_B$ *Note: Slope Mode can not be changed in Normal Mode*
RXD	10	r	**Output Signal of Receiver** Can be used to monitor the Receiver Output
TXD	9	rw	**LIN Transmitter switch on (only used when LIN_HV_MODE is set)** 0_B **Pull Down LIN Line**, Transmitter is switched on 1_B **Pull Up Resistor is active**, Transmitter is switched off
Res	8:7	r	**Reserved** Always read as 0
MODE_FB	6:4	r	**Feedback Signals for LIN Transmitter Mode Settings** Coding see **Table 356** *Note: Always read as “000” if the LIN module is disabled.*
Res	3	r	**Reserved** Always read as 0

Field	Bits	Type	Description
MODE	2:1	rw	**LIN transceiver power mode control** 00_B **LIN Sleep Mode**, LIN module switched to LIN Sleep Mode 01_B **LIN Receive-Only Mode**, LIN module switched to LIN Receive Only Mode 10_B **n.u.**, not used 11_B **LIN Normal Mode**, LIN module switched to LIN Normal Mode
EN	0	rw	**LIN Transceiver enable** 0_B **DISABLE**, LIN module disable 1_B **ENABLE**, LIN module enable

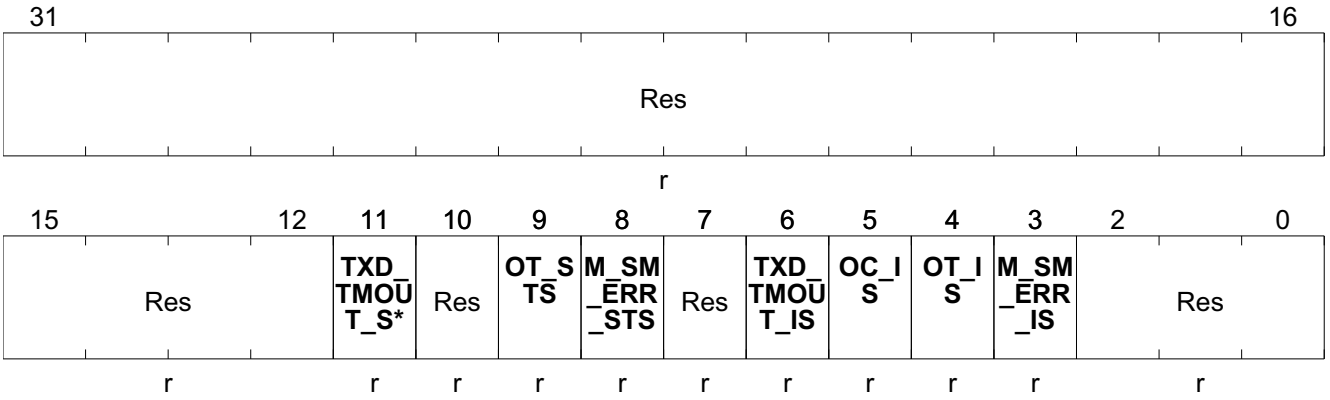
Table 360 RESET of LIN_CTRL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000 0001 1000 xxx0 0x10 0000 0111_B	RESET_TYPE_3		

LIN Transceiver

LIN Transceiver Interrupt Status

LIN_IRQS	Offset	Reset Value
LIN Transceiver Interrupt Status	04_H	see Table 361



Field	Bits	Type	Description
Res	31:12	r	**Reserved** Always read as 0
TXD_TMOUT_STS	11	r	**LIN TXD time-out Status** 0_B **NO_TIMEOUT**, no time-out occurred 1_B **TIMEOUT**, time-out occurred
Res	10	r	**Reserved** Always read as 0
OT_STS	9	r	**LIN Receiver Overtemperature Status** 0_B **no Overtemperature**, overtemperature occurred 1_B **Overtemperature**, overtemperature occurred
M_SM_ERR_STS	8	r	**LIN Transceiver Mode Error - Slope Mode Error Status** 0_B **no Mode Error - Slope Mode** , status occurred 1_B **Mode Error** , status occurred
Res	7	r	**Reserved** Always read as 0
TXD_TMOUT_IS	6	r	**LIN TXD time-out Interrupt Status** 0_B **NO_TIMEOUT**, no time-out occurred 1_B **TIMEOUT**, time-out occurred
OC_IS	5	r	**LIN Receiver Overcurrent Interrupt Status** 0_B **no Overcurrent**, overcurrent status occurred 1_B **Overcurrent**, overcurrent status occurred
OT_IS	4	r	**LIN Receiver Overtemperature Interrupt Status** 0_B **no Overtemperature**, overtemperature occurred 1_B **Overtemperature**, overtemperature occurred
M_SM_ERR_IS	3	r	**LIN Transceiver Mode Error - Slope Mode Error Interrupt Status** 0_B **no Mode Error - Slope Mode** , status occurred 1_B **Mode Error** , status occurred

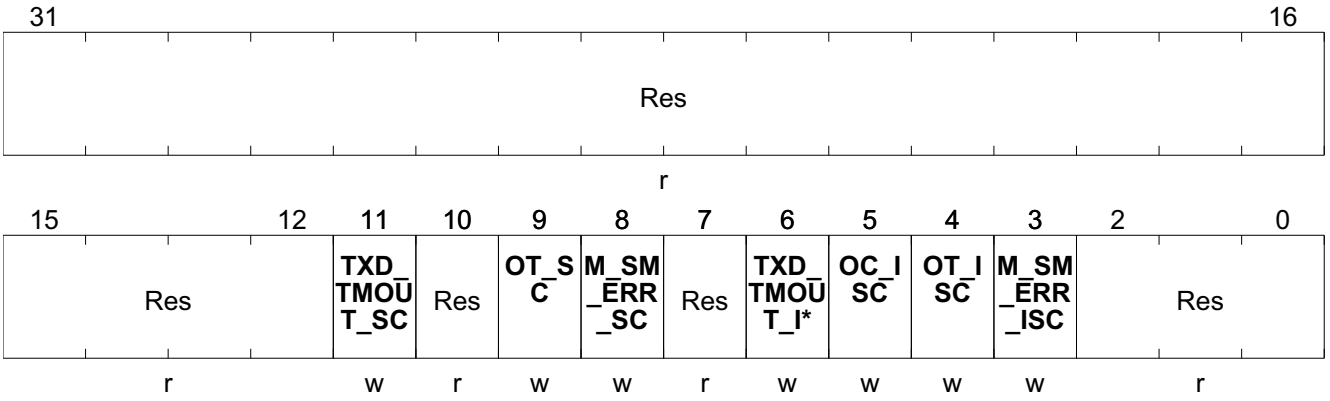
Field	Bits	Type	Description
Res	2:0	r	**Reserved** Always read as 1

Table 361 RESET of LIN_IRQS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

LIN Transceiver Interrupt Status Register Clear

LIN_IRQCLR	Offset	Reset Value
LIN Transceiver Interrupt Status Register Clear	08_H	see Table 362



Field	Bits	Type	Description
Res	31:12	r	**Reserved** Always read as 0
TXD_TMOUT_SC	11	w	**LIN TXD time-out Status Clear** 0_B **NO_Clear**, no time-out cleared 1_B **Clear**, time-out cleared
Res	10	r	**Reserved** Always read as 0
OT_SC	9	w	**LIN Receiver Overtemperature Status Clear** 0_B **NO_Clear**, overtemperature not cleared 1_B **Clear**, overtemperature cleared
M_SM_ERR_SC	8	w	**LIN Transceiver Mode Error - Slope Mode Error Status Clear** 0_B **NO_Clear**, overtemperature not cleared 1_B **Clear**, overtemperature cleared
Res	7	r	**Reserved** Always read as 0
TXD_TMOUT_ISC	6	w	**LIN TXD time-out Interrupt Status Clear** 0_B **NO_Clear**, no time-out cleared 1_B **Clear**, time-out cleared
OC_ISC	5	w	**LIN Receiver Overcurrent Interrupt Status Clear** 0_B **NO_Clear**, overcurrent status not cleared 1_B **Clear**, overcurrent status cleared
OT_ISC	4	w	**LIN Receiver Overtemperature Interrupt Status / Status Clear** 0_B **NO_Clear**, overtemperature not cleared 1_B **Clear**, overtemperature cleared

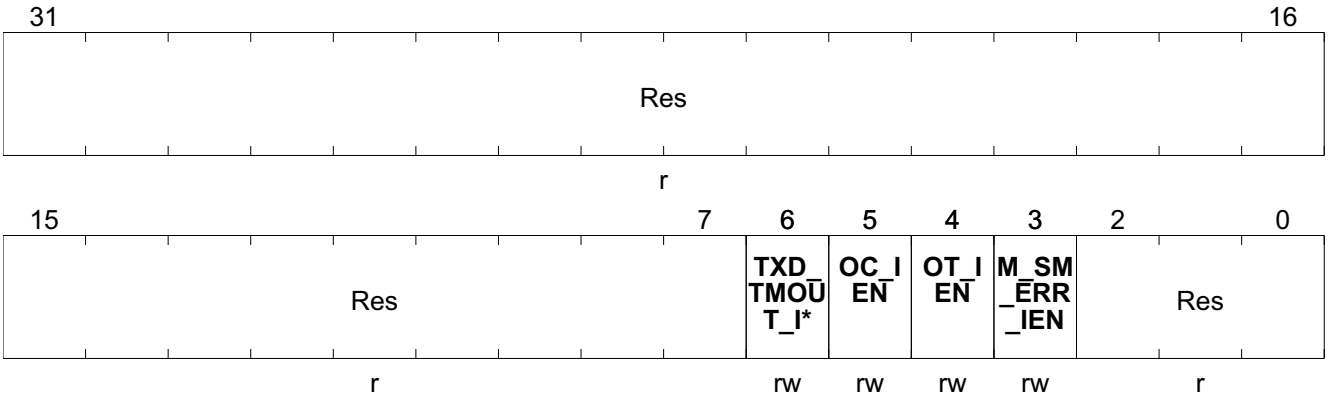
Field	Bits	Type	Description
M_SM_ERR_ISC	3	w	**LIN Transceiver Mode Error - Slope Mode Error Interrupt Status Clear** 0_B **NO_Clear**, overtemperature not cleared 1_B **Clear**, overtemperature cleared
Res	2:0	r	**Reserved** Always read as 1

Table 362 RESET of LIN_IRQCLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000_H	RESET_TYPE_3		

LIN Transceiver Interrupt Enable Register

LIN_IRQEN	Offset	Reset Value
LIN Transceiver Interrupt Enable Register	**$0C_H$**	**see Table 363**



Field	Bits	Type	Description
Res	31:7	r	**Reserved** Always read as 0
TXD_TMOUT_IEN	6	rw	**LIN Transceiver TxD-Timeout interrupt enable** 0_B **disable**, 1_B **enable**,
OC_IEN	5	rw	**LIN Transceiver Overcurrent interrupt enable** 0_B **disable**, 1_B **enable**,
OT_IEN	4	rw	**LIN Transceiver Overtemperature interrupt enable** 0_B **disable**, 1_B **enable**,
M_SM_ERR_IEN	3	rw	**LIN Transceiver Mode - Slope Mode Error interrupt enable** 0_B **disable**, 1_B **enable**,

Field	Bits	Type	Description
Res	2:0	r	**Reserved** Always read as 1

Table 363 RESET of LIN_IRQEN

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000_H	RESET_TYPE_3		

20.5 LIN Transceiver Interrupts

The LIN Transceiver has four different interrupt sources:

LIN-Interrupt Sources:

- overcurrent interrupt; will occur when current limitation is active
- overtemperature interrupt; will occur when thermal sensor detects overtemperature
- TxD-Timeout; will occur when LIN TxD internal signal is dominant for a defined period of time
- LIN Control Signal Feedback failure; will occur when the feedback control signals of LIN Transceiver are not correct

The output interrupt signal to Node 10 look like:

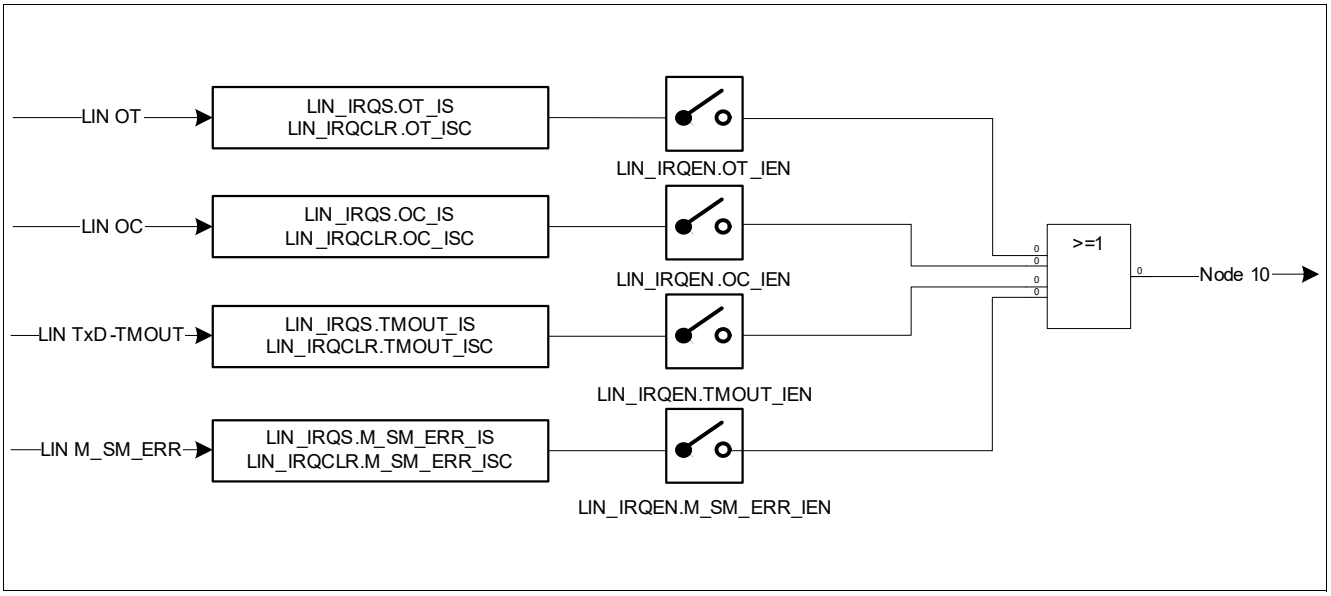


Figure 179 LIN Interrupt Signal Generation

21 High-Speed Synchronous Serial Interface SSC1/2

21.1 Features

- Master and Slave Mode operation
  - Full-duplex or half-duplex operation
- Transmit and receive double buffered
- Flexible data format
  - Programmable number of data bits: 2 to 16 bits
  - Programmable shift direction: Least Significant Bit (LSB) or Most Significant Bit (MSB) shift first
  - Programmable clock polarity: idle low or high state for the shift clock
  - Programmable clock/data phase: data shift with leading or trailing edge of the shift clock
- Variable baud rate, e.g. 250kBaud - 8MBaud
- Compatible with Serial Peripheral Interface (SPI)
- Interrupt generation
  - On a transmitter empty condition
  - On a receiver full condition
  - On an error condition (receive, phase, baud rate, transmit error)
  - On a transfer complete condition
- Port direction selection, see **Chapter 15**

21.2 Introduction

The High-Speed Synchronous Serial Interface (SSC) supports both full-duplex and half-duplex serial synchronous communication. The serial clock signal can be generated by the SSC internally (master mode), using its own 16-Bit baud-rate generator, or can be received from an external master (slave mode). Data width, shift direction, clock polarity, and phase are programmable. This allows communication with SPI-compatible devices or devices using other synchronous serial interfaces.

Data is transmitted or received on lines TXD and RXD, which are normally connected to the pins MTSR (MasterTransmit/Slave Receive) and MRST (Master Receive/Slave Transmit). The clock signal is output via line MS_CLK (Master Serial Shift Clock) or input via line SS_CLK (Slave Serial Shift Clock). Both lines are normally connected to the pin SCLK. Transmission and reception of data are double-buffered.

21.2.1 Block Diagram

Figure 180 shows all functional relevant interfaces associated with the SSC Kernel.

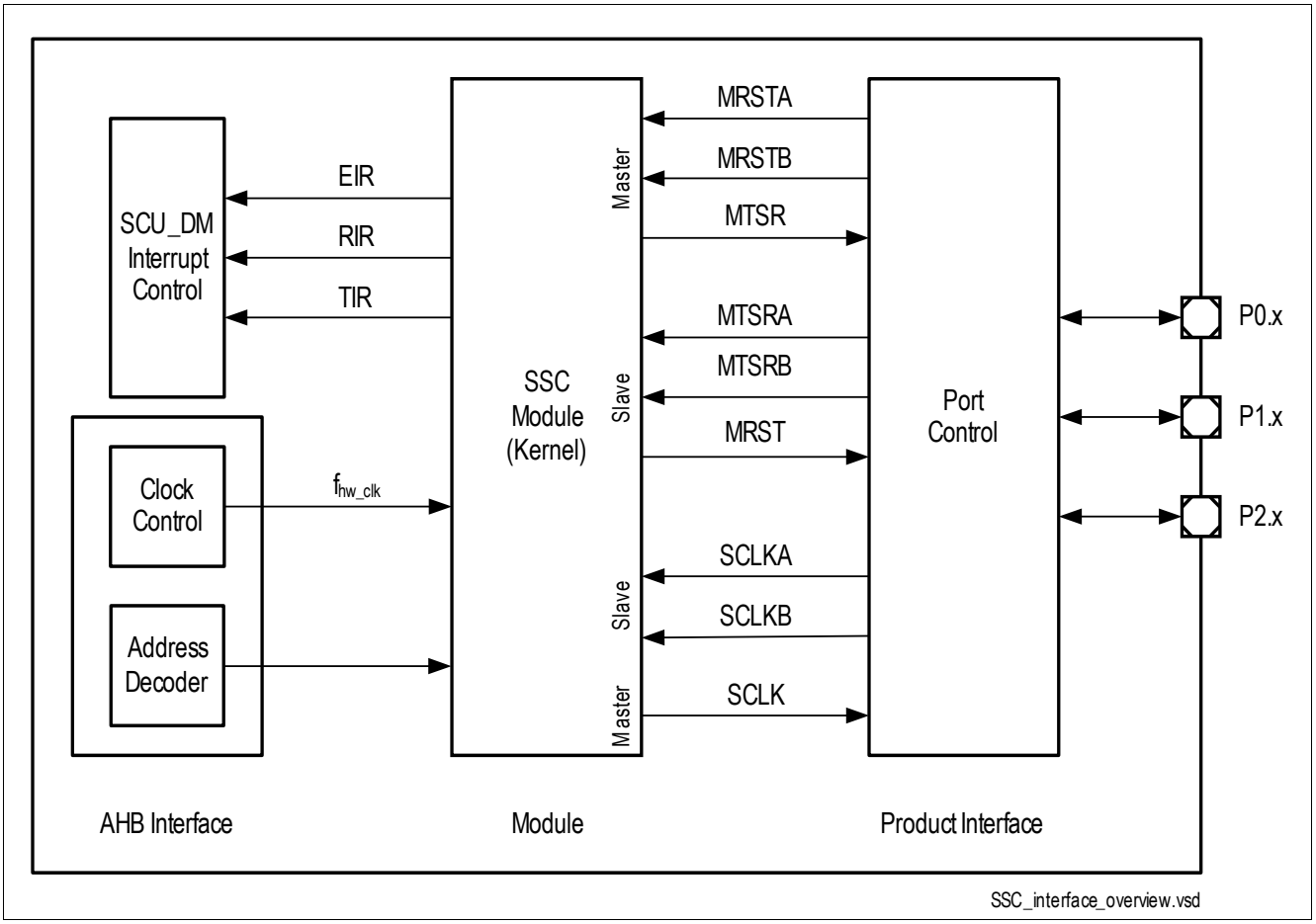


Figure 180 SSC Interface Diagram

21.3 Functional Description

21.3.1 SSC1 and SSC2 Mode Overview

The SSC supports full-duplex and half-duplex synchronous communication up to 20 MBaud (@ 40 MHz module clock). The serial clock signal can be generated by the SSC itself (Master Mode) or can be received from an external master (Slave Mode). Data width, shift direction, clock polarity, and phase are programmable. This allows communication with SPI-compatible devices. Transmission and reception of data is double-buffered. A 16-bit baud-rate generator provides the SSC with a separate serial clock signal.

The SSC can be configured in a very flexible way, so it can be used with other synchronous serial interfaces, can serve for master/slave or multimaster interconnections or can operate compatible with the popular SPI interface. Thus, the SSC can be used to communicate with shift registers (I/O expansion), peripherals (e.g. EEPROMs, etc.) or other controllers (networking). The SSC supports half-duplex and full-duplex communication. Data is transmitted or received on lines TXD and RXD, normally connected with pins MTSR (Master Transmit/Slave Receive) and MRST (Master Receive/Slave Transmit). The clock signal is output via line MS_CLK (Master Serial Shift Clock) or input via line SS_CLK (Slave Serial Shift Clock). Both lines are normally connected to pin SCLK.

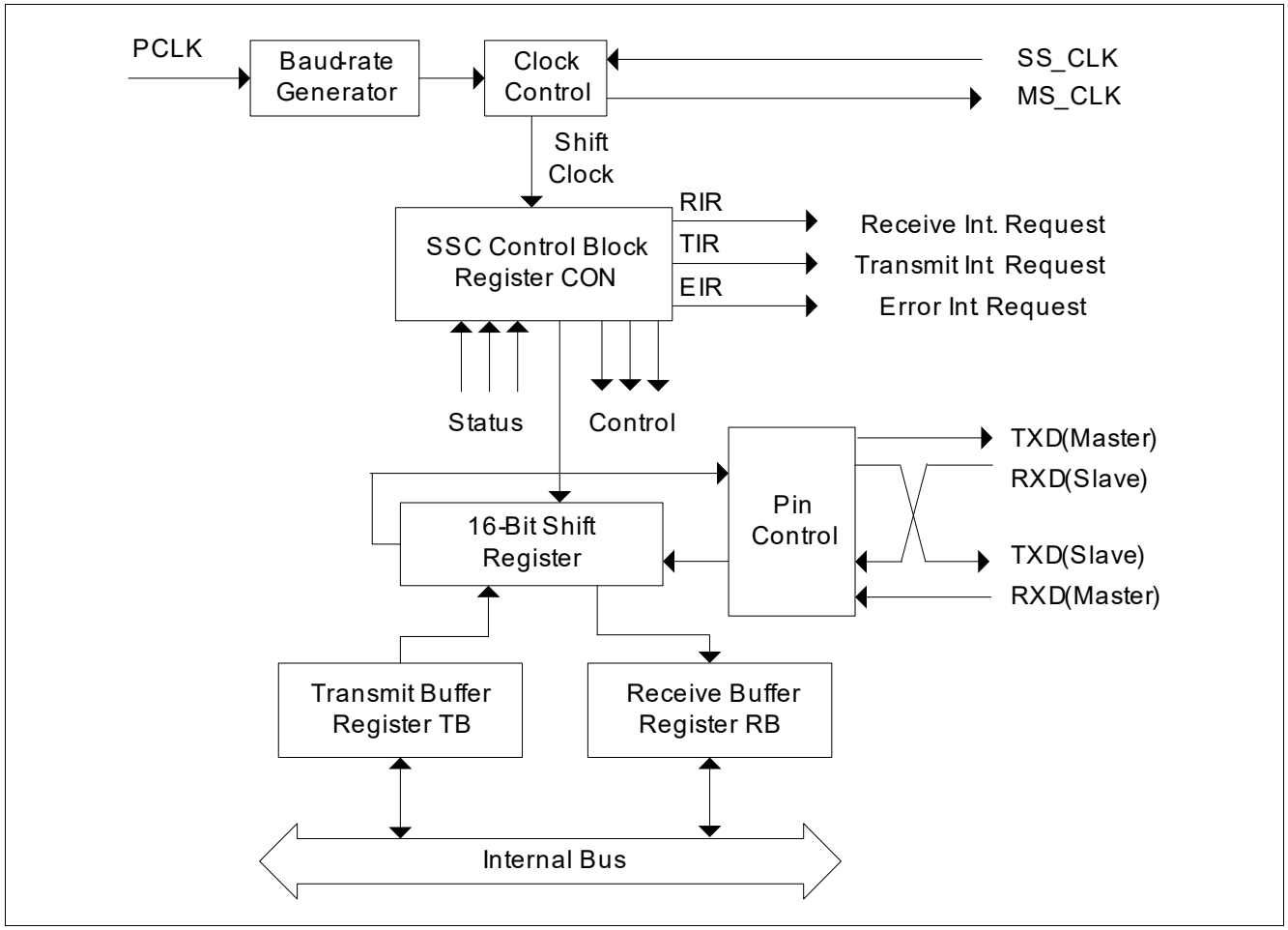


Figure 181 Synchronous Serial Channel SSC Block Diagram

21.3.2 Operating Mode Selection

The operating mode of the serial channel SSC is controlled by its control register CON. This register serves two purposes:

- During programming (SSC disabled by CON.EN = 0), it provides access to a set of control bits
- During operation (SSC enabled by CON.EN = 1), it provides access to a set of status flags.

The shift register of the SSC is connected to both the transmit lines and the receive lines via the pin control logic (see block diagram in **Figure 181**). Transmission and reception of serial data are synchronized and take place at the same time, i.e. the same number of transmitted bits is also received. Transmit data is written into the Transmit Buffer (TB) and is moved to the shift register as soon as this is empty. An SSC master (CON.MS = 1) immediately begins transmitting, while an SSC slave (CON.MS = 0) will wait for an active shift clock. When the transfer starts, the busy flag CON.BSY is set and the Transmit Interrupt Request line TIR will be activated to indicate that register TB may be reloaded again. When the programmed number of bits (2 … 16) has been transferred, the contents of the shift register are moved to the Receive Buffer RB and the Receive Interrupt Request line RIR will be activated. If no further transfer is to take place (TB is empty), CON.BSY will be cleared at the same time. Software should not modify CON.BSY, as this flag is hardware controlled.

Note: The SSC starts transmission and sets CON.BSY minimum two clock cycles after transmit data is written into TB. Therefore, it is not recommended to poll CON.BSY to indicate the start and end of a single transmission. Instead, interrupt service routine should be used if interrupts are enabled, or the interrupt flags IRCON1.TIR and IRCON1.RIR should be polled if interrupts are disabled.

Note: Only one SSC (etc.) can be master at a given time.

The transfer of serial data bits can be programmed in many respects:

- The data width can be specified from 2 bits to 16 bits
- A transfer may start with either the LSB or the MSB
- The shift clock may be idle low or idle high
- The data bits may be shifted with the leading edge or the trailing edge of the shift clock signal
- The baud rate may be set from 305.18 Baud up to 20 MBaud (@ 40 MHz module clock)
- The shift clock can be generated (MS_CLK) or can be received (SS_CLK)

These features allow the adaptation of the SSC to a wide range of applications requiring serial data transfer.

The Data Width Selection supports the transfer of frames of any data length, from 2-bit “characters” up to 8-bit “characters”. Starting with the LSB (CON.HB = 0) allows communication with SSC devices in Synchronous Mode or with 8051 like serial interfaces for example. Starting with the MSB (CON.HB = 1) allows operation compatible with the SPI interface.

Regardless of the data width selected and whether the MSB or the LSB is transmitted first, the transfer data is always right-aligned in registers TB and RB, with the LSB of the transfer data in bit 0 of these registers. The data bits are rearranged for transfer by the internal shift register logic. The unselected bits of TB are ignored; the unselected bits of RB will not be valid and should be ignored by the receiver service routine.

The Clock Control allows the adaptation of transmit and receive behavior of the SSC to a variety of serial interfaces. A specific shift clock edge (rising or falling) is used to shift out transmit data, while the other shift clock edge is used to latch in receive data. Bit CON.PH selects the leading edge or the trailing edge for each function. Bit CON.PO selects the level of the shift clock line in the idle state. Thus, for an idle-high clock, the leading edge is a falling one, a 1-to-0 transition (see **Figure 182**).

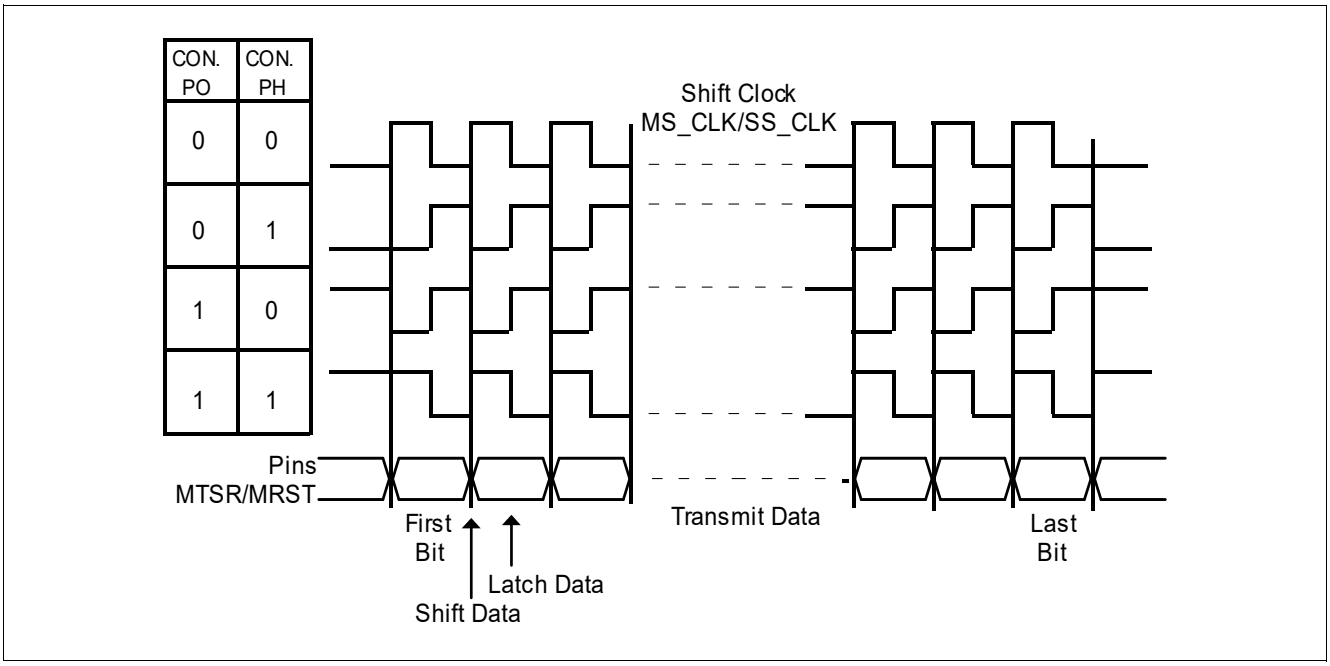


Figure 182 Serial Clock Phase and Polarity Options

21.3.3 Full-Duplex Operation

The various devices are connected through three lines. The definition of these lines is always determined by the master: the line connected to the master’s data output line TXD is the transmit line; the receive line is connected to its data input line RXD; the shift clock line is either MS_CLK or SS_CLK. Only the device selected

for master operation generates and outputs the shift clock on line MS_CLK. Since all slaves receive this clock, their pin SCLK must be switched to input mode. The output of the master's shift register is connected to the external transmit line, which in turn is connected to the slaves' shift register input. The output of the slaves' shift register is connected to the external receive line in order to enable the master to receive the data shifted out of the slave. The external connections are hard-wired, the function and direction of these pins is determined by the master or slave operation of the individual device.

Note: The shift direction shown in the figure applies for MSB-first operation as well as for LSB-first operation.

When initializing the devices in this configuration, one device must be selected for master operation while all other devices must be programmed for slave operation. Initialization includes the operating mode of the device's SSC and also the function of the respective port lines.

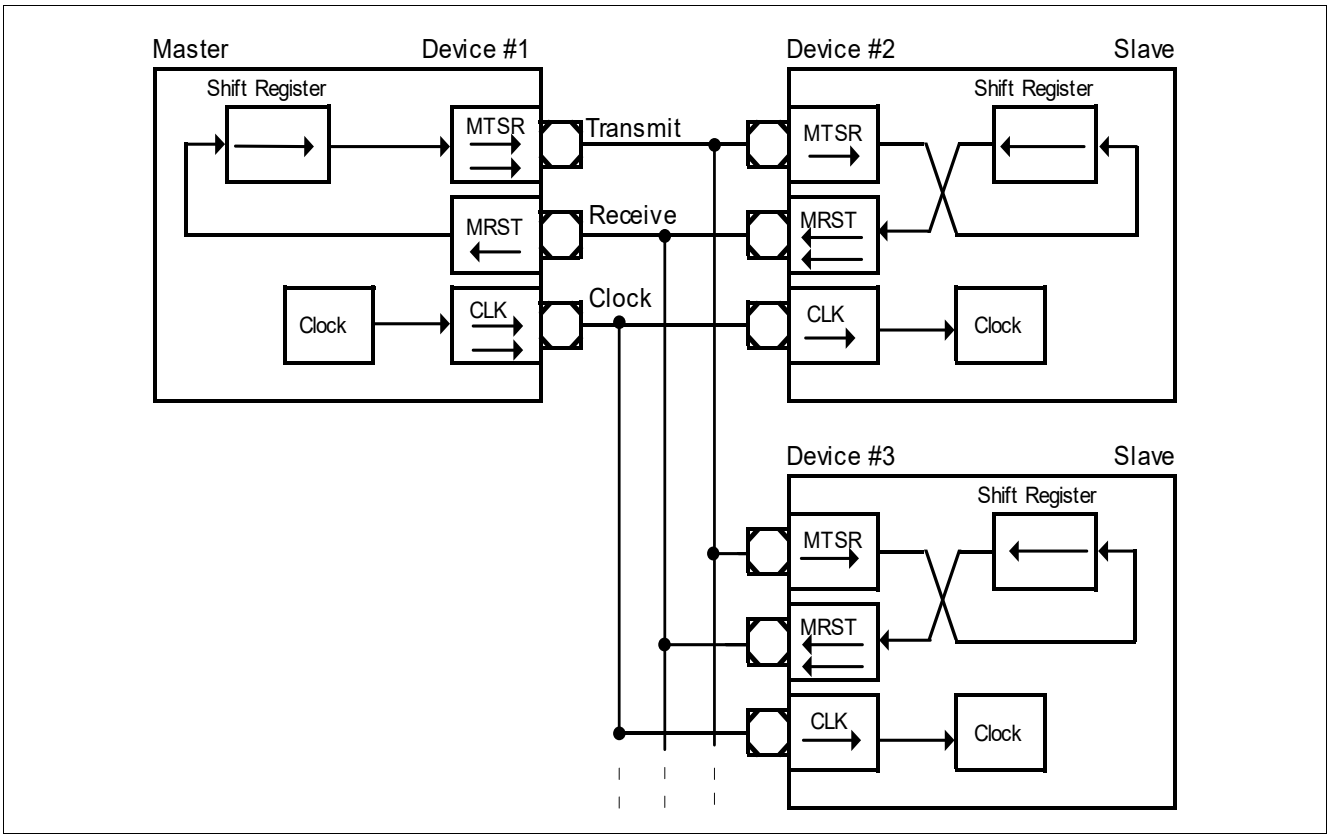


Figure 183 SSC Full-Duplex Configuration

The data output pins MRST of all slave devices are connected together onto the one receive line in the configuration shown in **Figure 183**. During a transfer, each slave shifts out data from its shift register. There are two ways to avoid collisions on the receive line due to different slave data:

- Only one slave drives the line, i.e. enables the driver of its MRST pin. All the other slaves must have their MRST pins programmed as input so only one slave can put its data onto the master's receive line. Only receiving data from the master is possible. The master selects the slave device from which it expects data either by separate select lines, or by sending a special command to this slave. The selected slave then switches its MRST line to output until it gets a de-selection signal or command.
- The slaves use open drain output on MRST. This forms a wired-AND connection. The receive line needs an external pull-up in this case. Corruption of the data on the receive line sent by the selected slave is avoided when all slaves not selected for transmission to the master only send ones (1s). Because this high level is

not actively driven onto the line, but only held through the pull-up device, the selected slave can pull this line actively to a low-level when transmitting a zero bit. The master selects the slave device from which it expects data either by separate select lines or by sending a special command to this slave.

After performing the necessary initialization of the SSC, the serial interfaces can be enabled. For a master device, the alternate clock line will now go to its programmed polarity. The alternate data line will go to either 0 or 1 until the first transfer starts. After a transfer, the alternate data line will always remain at the logic level of the last transmitted data bit.

When the serial interfaces are enabled, the master device can initiate the first data transfer by writing the transmit data into register TB. This value is copied into the shift register (assumed to be empty at this time), and the selected first bit of the transmit data will be placed onto the TXD line on the next clock from the baud-rate generator (transmission starts only if CON.EN = 1). Depending on the selected clock phase, a clock pulse will also be generated on the MS_CLK line. At the same time, with the opposite clock edge, the master latches and shifts in the data detected at its input line RXD. This “exchanges” the transmit data with the receive data. Because the clock line is connected to all slaves, their shift registers will be shifted synchronously with the master’s shift register — shifting out the data contained in the registers, and shifting in the data detected at the input line. After the preprogrammed number of clock pulses (via the data width selection), the data transmitted by the master is contained in all the slaves’ shift registers, while the master’s shift register holds the data of the selected slave. In the master and all slaves, the contents of the shift register are copied into the receive buffer RB and the receive interrupt line RIR is activated.

A slave device will immediately output the selected first bit (MSB or LSB of the transfer data) at line RXD when the contents of the transmit buffer are copied into the slave's shift register. Bit CON.BSY is not set until the first clock edge at SS_CLK appears. The slave device will not wait for the next clock from the baud-rate generator, as the master does. The reason for this is that, depending on the selected clock phase, the first clock edge generated by the master may already be used to clock in the first data bit. Thus, the slave’s first data bit must already be valid at this time.

*Note: On the SSC, a transmission **and** a reception takes place at the same time, regardless of whether valid data has been transmitted or received.*

Note: The initialization of the CLK pin on the master requires some attention in order to avoid undesired clock transitions, which may disturb the other devices. Before the clock pin is switched to output via the related direction control register, the clock output level will be selected in the control register CON and the alternate output be prepared via the related ALTSEL register, or the output latch must be loaded with the clock idle level.

21.3.4 Half-Duplex Operation

In a Half-Duplex Mode, only one data line is necessary for both receiving **and** transmitting of data. The data exchange line is connected to both the MTSR and MRST pins of each device, the shift clock line is connected to the SCLK pin.

The master device controls the data transfer by generating the shift clock, while the slave devices receive it. Due to the fact that all transmit and receive pins are connected to the one data exchange line, serial data may be moved between arbitrary stations.

Similar to Full-Duplex Mode, there are two ways to avoid collisions on the data exchange line:

- Only the transmitting device may enable its transmit pin driver
- The non-transmitting devices use open drain output and send only ones.

Because the data inputs and outputs are connected together, a transmitting device will clock in its own data at the input pin (MRST for a master device, MTSR for a slave). By this method, any corruptions on the common data exchange line are detected if the received data is not equal to the transmitted data.

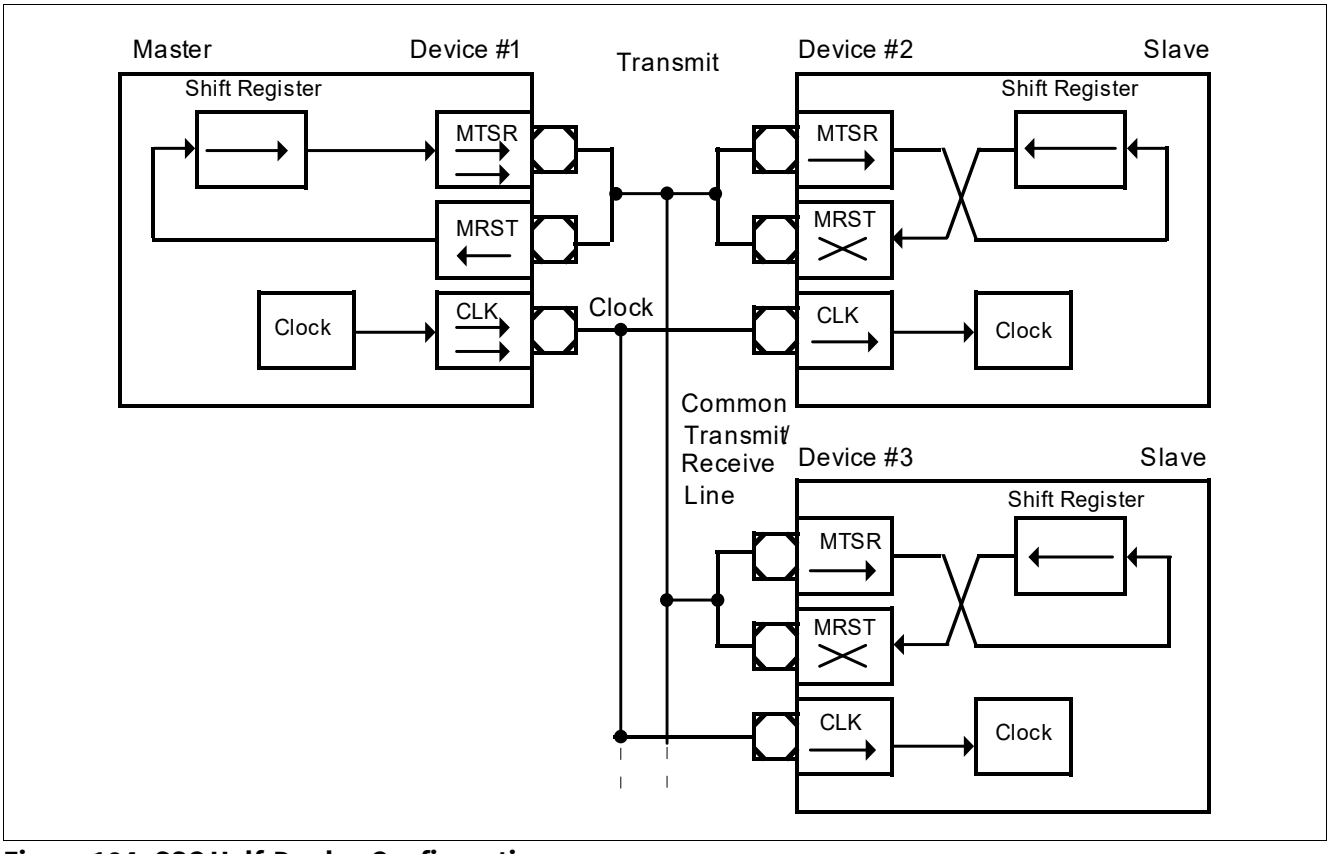


Figure 184 SSC Half-Duplex Configuration

21.3.5 Continuous Transfers

When the transmit interrupt request flag is set, it indicates that the transmit buffer TB is empty and ready to be loaded with the next transmit data. If TB has been reloaded by the time the current transmission is finished, the data is immediately transferred to the shift register and the next transmission will start without any additional delay. On the data line, there is no gap between the two successive frames. For example, two byte transfers would look the same as one word transfer. This feature can be used to interface with devices that can operate with or require more than 8 data bits per transfer. It is just a matter of software, how long a total data frame length can be. This option can also be used to interface to byte-wide and word-wide devices on the same serial bus, for instance.

Note: Of course, this can happen only in multiples of the selected basic data width, because it would require disabling/enabling of the SSC to reprogram the basic data width on-the-fly.

21.3.5.1 Port Control

The SSC uses three lines to communicate with the external world. Pin SCLK serves as the clock line, while pins MRST (Master Receive/Slave Transmit) and MTSR (Master Transmit/Slave Receive) serve as the serial data input/output lines. As shown in **Figure 180** these three lines (SCLK as input, Master Receive, Slave Receive) have all two inputs at the SSC Module kernel. Three bits in register PISEL define which of the two kernel inputs (A or B) are connected. This feature allows for each of the three SSC communication lines to be connected to two inputs coming from different port pins.

Operation of the SSC I/O lines depends on the selected operating mode (master or slave). The direction of the port lines depends on the operating mode. The SSC will automatically use the correct kernel output or kernel input line of the ports when switching modes. Port pins assigned as SSC I/O lines can be controlled in two ways:

- By hardware
- By software

When the SSC I/O lines are connected with dedicated pins typically hardware I/O control should be used. In this case, the two output signals reflect directly the state of the CON.EN and CON.MS bits (the M/S select line is inverted to the CON.MS bit definition).

When the SSC I/O lines are connected with bidirectional lines of general purpose I/O ports, typically software I/O control should be used. In this case port registers must be programmed for alternate output and input selection. When switching between master and slave mode, port registers must be reprogrammed.

21.3.6 Baud Rate Generation

The serial channel SSC has its own dedicated 16-bit baud-rate generator with 16-bit reload capability, allowing baud rate generation independent of the timers. **Figure 181** shows the baud-rate generator. **Figure 185** shows the baud-rate generator of the SSC in more detail.

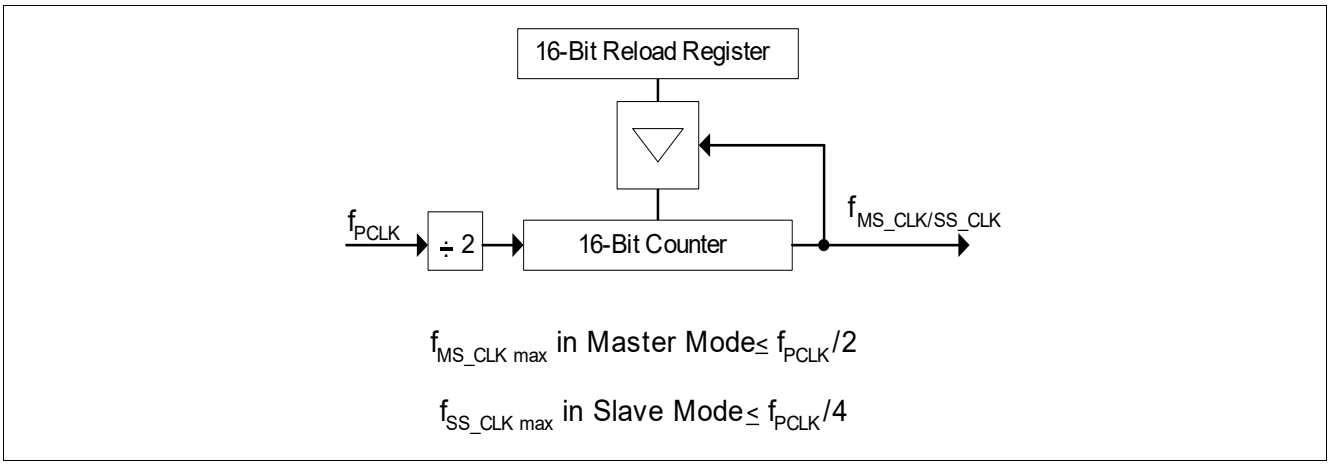


Figure 185 SSC Baud-rate Generator

The baud-rate generator is clocked with the module clock f_{hw_clk}. The timer counts downwards. Register BR is the dual-function Baud-rate Generator/Reload register. Reading BR, while the SSC is enabled, returns the contents of the timer. Reading BR, while the SSC is disabled, returns the programmed reload value. In this mode, the desired reload value can be written to BR.

Note: Never write to BR while the SSC is enabled.

The formulas below calculate either the resulting baud rate for a given reload value, or the required reload value for a given baud rate:

$$\text{Baud rate} = \frac{f_{hw\_clk}}{2 \bullet (
 + 1)} \tag{21.1}$$

$$BR = \frac{f_{hw\_clk}}{2 \bullet \text{Baud rate}} - 1 \tag{21.2}$$

 represents the contents of the reload register, taken as an unsigned 16-bit integer, while baud rate is equal to f_{MS_CLK/SS_CLK} as shown in **Figure 185**.

The maximum baud rate that can be achieved when using a module clock of 40 MHz is 20 MBaud in Master Mode (with
 = 0000_H) or 10 MBaud in Slave Mode (with
 = 0001_H).

Table 364 lists some possible baud rates together with the required reload values and the resulting bit times, assuming a module clock of 40 MHz.

Table 364 Typical Baud Rates of the SSC (f_{hw_clk} = 40 MHz)

Reload Value	Baud Rate (= f_{MS_CLK/SS_CLK})	Deviation
0000_H	20 MBaud (only in Master Mode)	0.0%
0001_H	10 MBaud	0.0%
0013_H	1 MBaud	0.0%
0027_H	500 kBaud	0.0%

Table 364 Typical Baud Rates of the SSC (f_{hw_clk} = 40 MHz)

Reload Value	Baud Rate (= f_{MS_CLK/SS_CLK})	Deviation
$00C7_H$	100 kBaud	0.0%
$07CF_H$	10 kBaud	0.0%
$4E1F_H$	1 kBaud	0.0%
$FFFF_H$	305.18 Baud	0.0%

21.3.7 Error Detection Mechanisms

The SSC is able to detect four different error conditions. Receive Error and Phase Error are detected in all modes; Transmit Error and Baud Rate Error apply only to Slave Mode. When an error is detected, the respective error flag is/can be set and an error interrupt request will be generated by activating the EIR line (see **Figure 186**) if enabled. The error interrupt handler may then check the error flags to determine the cause of the error interrupt. The error flags are not reset automatically but rather must be cleared by software after servicing. This allows servicing of some error conditions via interrupt, while the others may be polled by software.

Note: The error interrupt handler must clear the associated (enabled) error flag(s) to prevent repeated interrupt requests.

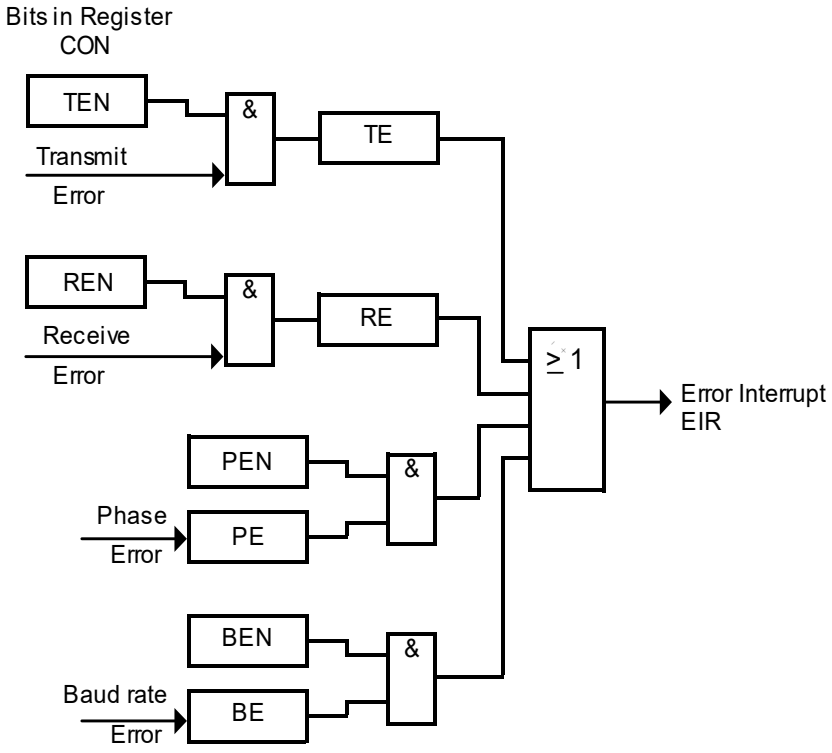


Figure 186 SSC Error Interrupt Control

A **Receive Error** (Master or Slave Mode) is detected when a new data frame is completely received but the previous data was not read out of the receive buffer register RB. This condition sets the error flag CON.RE and the error interrupt request line EIR, when enabled via CON.REN. The old data in the receive buffer RB will be overwritten with the new value and is irretrievably lost.

A **Phase Error** (Master or Slave Mode) is detected when the incoming data at pin MRST (Master Mode) or MTSR (Slave Mode), sampled with the same frequency as the module clock, changes between one cycle before and two cycles after the latching edge of the shift clock signal SCLK. This condition sets the error flag CON.PE and, when enabled via CON.PEN, the error interrupt request line EIR.

Note: When receiving and transmitting data in parallel, phase errors occur if the baud rate is configured to f_{hw_clk} / 2.

A **Baud Rate Error** (Slave Mode) is detected when the incoming clock signal deviates from the programmed baud rate by more than 100%, i.e. it is either more than double or less than half the expected baud rate. This condition sets the error flag CON.BE and, when enabled via CON.BEN, the error interrupt request line EIR. Using this error detection capability requires that the slave's baud-rate generator is programmed to the same baud rate as the master device. This feature detects false additional, or missing pulses on the clock line (within a certain frame).

Note: If this error condition occurs and bit CON.REN = 1, an automatic reset of the SSC will be performed in case of this error. This is done to re-initialize the SSC if too few or too many clock pulses have been detected.

Note: This error can occur after any transfer if the communication is stopped. This is the case due to the fact that the SSC module supports back-to-back transfers for multiple transfers. In order to handle this, the baud rate detector expects after a finished transfer immediately a next clock cycle for a new transfer.

A **Transmit Error** (Slave Mode) is detected when a transfer was initiated by the master (SS_CLK gets active) but the transmit buffer TB of the slave was not updated since the last transfer. This condition sets the error flag CON.TE and the error interrupt request line EIR, when enabled via CON.TEN. If a transfer starts while the transmit buffer is not updated, the slave will shift out the 'old' contents of the shift register, which normally is the data received during the last transfer. This may lead to corruption of the data on the transmit/receive line in half-duplex mode (open drain configuration) if this slave is not selected for transmission. This mode requires that slaves not selected for transmission only shift out ones; that is, their transmit buffers must be loaded with '$FFFF_H$' prior to any transfer.

Note: A slave with push/pull output drivers not selected for transmission, will normally have its output drivers switched. However, in order to avoid possible conflicts or misinterpretations, it is recommended to always load the slave's transmit buffer prior to any transfer.

The cause of an error interrupt request (receive, phase, baud rate, transmit error) can be identified by the error status flags in control register CON.

Note: In contrast to the error interrupt request line EIR, the error status flags CON.TE, CON.RE, CON.PE, and CON.BE, are not reset automatically upon entry into the error interrupt service routine, but must be cleared by software.

21.4 Interrupts

The three SSC interrupts can be separately enabled or disabled by setting or clearing their corresponding enable bits in SFR SCU_MODIEN.

For a detailed description of the various interrupts see **Section 21.3**. An overview is given in **Table 365**.

Table 365 SSC Interrupt Sources

Interrupt	Signal	Description
Transmission starts	TIR	Indicates that the transmit buffer can be reloaded with new data.
Transmission ends	RIR	The configured number of bits have been transmitted and shifted to the receive buffer.
Receive Error	EIR	This interrupt occurs if a new data frame is completely received and the last data in the receive buffer was not read.
Phase Error	EIR	This interrupt is generated if the incoming data changes between one cycle before and two cycles after the latching edge of the shift clock signal SCLK.
Baud Rate Error (Slave Mode only)	EIR	This interrupt is generated when the incoming clock signal deviates from the programmed baud rate by more than 100%.
Transmit Error (Slave Mode only)	EIR	This interrupt is generated when TB was not updated since the last transfer if a transfer is initiated by a master.

21.5 SSC Kernel Registers

There are two SSC kernels in the TLE985xQX, namely SSC1 and SSC2. **Table 366** shows the SSC module base addresses.

Table 366 Register Address Space

Module	Base Address	End Address	Note
SSC1	48024000_H	$48025FFF_H$	Synchronous Serial Interface 1
SSC2	48026000_H	$48027FFF_H$	Synchronous Serial Interface 2

Table 367 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
SSC Kernel Registers, Port Input Select Register			
SSC_PISEL	Port Input Select Register	00_H	$0000\ 0000_H$
SSC Kernel Registers, Configuration Register			
SSC_CON	Control Register	04_H	$0000\ 0000_H$
SSC_ISRCLR	Interrupt Status Register Clear	14_H	$0000\ 0000_H$
SSC Kernel Registers, Baud Rate Timer Reload Register			
SSC_BR	Baud Rate Timer Reload Register	10_H	$0000\ 0000_H$
SSC Kernel Registers, Transmitter Buffer Register			
SSC_TB	Transmitter Buffer Register	08_H	$0000\ 0000_H$
SSC Kernel Registers, Receiver Buffer Register			
SSC_RB	Receiver Buffer Register	$0C_H$	$0000\ 0000_H$

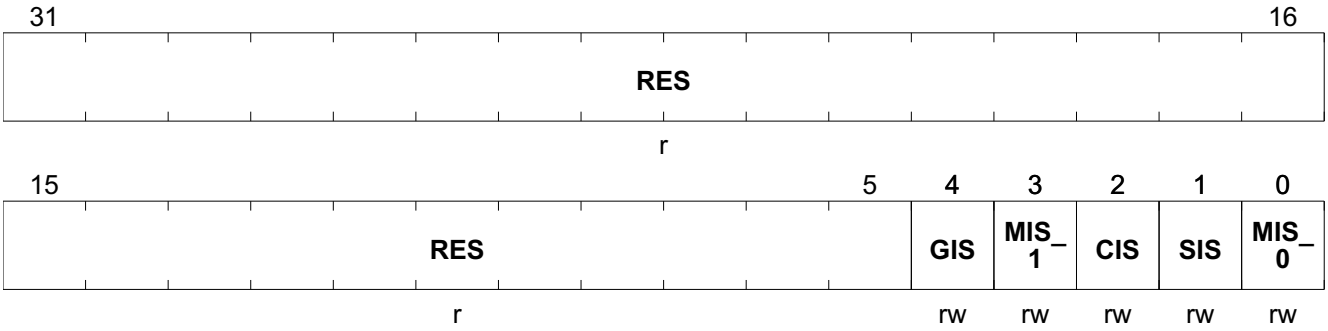
The registers are addressed wordwise.

21.5.1 Port Input Select Register

Port Input Select Register

The PISEL register controls the receiver input selection of the SSC module. In the implementation of TLE985xQX, the PISEL register is not used.

SSC_PISEL	Offset	Reset Value
Port Input Select Register	**00_H**	**see Table 368**



Field	Bits	Type	Description
RES	31:5	r	**Reserved** Always read as 0; should be written with 0.
GIS	4	rw	**Global SSC12 Input Select** 0_B Inputs SSC12_S_SCK_0, SSC12_S_MTSR_0, and SSC12_M_MRST_0 are selected if CIS, SIS or MIS_O is 1. 1_B Inputs SSC12_S_SCK_1, SSC12_S_MTSR_1, and SSC12_M_MRST_1 are selected if CIS, SIS or MIS_O is 1. See **Chapter 15.3.2.3**
MIS_1	3	rw	**Master Mode Input Select Bit 1 (Master Mode only)** 0_B Default, Inputs selected according to MIS_0. 1_B Do not use, Connects to unused pins.
CIS	2	rw	**Clock Input Select (Slave Mode only)** 0_B **SSCx_S_SCK**, (x = 1 or 2, dependant from current SSC), see **Chapter 15.3.2**. 1_B **SSC12_S_SCK_x**, (x=0 or 1). See **Chapter 15.3.2**.
SIS	1	rw	**Slave Mode Input Select (Slave Mode only)** 0_B **SSCx_S_MTSR**, (x = 1 or 2, dependant from current SSC), see **Chapter 15.3.2**. 1_B **SSC12_S_MTSR_x**, (x=0 or 1). See **Chapter 15.3.2**.
MIS_0	0	rw	**Master Mode Input Select Bit 0 (Master Mode only)** 0_B **SSCx_M_MRST**, (x = 1 or 2, dependant from current SSC), see **Chapter 15.3.2**. 1_B **SSC12_M_MRST_x**, (x=0 or 1). See **Chapter 15.3.2**.

Table 368 RESET of SSC_PISEL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

21.5.2 Configuration Register

The operating mode of the serial channel SSC is controlled by the control register CON. This register contains control bits for mode and error check selection, and status flags for error identification. Depending on bit EN, either control functions or status flags and master/slave control are enabled.

Control Register

SSC_CON	Offset	Reset Value
Control Register	04_H	see Table 369

31	30	29	28	27	26	25	24
RES			BSY	BE	PE	RE	TE
r			r	r	r	r	r

23	22	21	20	19	18	17	16
RES				BC			
r				r			

15	14	13	12	11	10	9	8
EN	MS	RES	AREN	BEN	PEN	REN	TEN
rw	rw	r	rw	rw	rw	rw	rw

7	6	5	4	3	2	1	0
LB	PO	PH	HB	BM			
rw	rw	rw	rw	rw			

Field	Bits	Type	Description
RES	31:29	r	**Reserved** Always read as 0; should be written with 0.
BSY	28	r	**Busy Flag** Can only be read when EN=1 (operating mode). Invalid data when EN=0 (programming mode). Set while a transfer is in progress. *Note: This bit is not to be written to.*
BE	27	r	**Baud Rate Error Flag** Can only be read when EN=1 (operating mode). Invalid data when EN=0 (programming mode). 0_B **NO**, error. 1_B **ERROR**, More than factor 2 or 0.5 between slave's actual and expected baud rate.
PE	26	r	**Phase Error Flag** Can only be read when EN=1 (operating mode). Invalid data when EN=0 (programming mode). 0_B **NO**, error. 1_B **ERROR**, Received data changes around sampling clock edge.

Field	Bits	Type	Description
RE	25	r	**Receive Error Flag** Can only be read when EN=1 (operating mode). Invalid data when EN=0 (programming mode). 0_B **NO**, error. 1_B **ERROR**, Reception completed before the receive buffer was read.
TE	24	r	**Transmit Error Flag** Can only be read when EN=1 (operating mode). Invalid data when EN=0 (programming mode). 0_B **NO**, error. 1_B **ERROR**, Transfer starts with the slave's transmit buffer not being updated.
RES	23:20	r	**Reserved** Returns 0 if read; should be written with 0.
BC	19:16	r	**Bit Count Field** Can only be read when EN=1 (operating mode). Invalid data when EN=0 (programming mode). Shift counter is updated with every shift bit. *Note: This bit field is not to be written to.*
EN	15	rw	**Enable Bit** *Note: The effect of EN bit becomes visible on the next write to the CON register.* 0_B **Programming Mode**, Transmission and reception disabled. Access to control bits. 1_B **Operating Mode**, Transmission and reception enabled. Access to status flags and M/S control.
MS	14	rw	**Master Select** 0_B **SLAVE**, Mode. Operate on shift clock received via SCLK. 1_B **MASTER**, Mode. Generate shift clock and output it via SCLK.
RES	13	r	**Reserved** Returns 0 if read; should be written with 0.
AREN	12	rw	**Automatic Reset Enable** Can only be accessed when EN=0 (programming mode). Invalid data when EN=1 (operating mode). 0_B **N/A**, No additional action upon a baud rate error. 1_B **RESET**, The SSC is automatically reset upon a baud rate error.

Field	Bits	Type	Description
BEN	11	rw	**Baud Rate Error Enable** Can only be accessed when EN=0 (programming mode). Invalid data when EN=1 (operating mode). 0_B **IGNORE**, baud rate errors. 1_B **CHECK**, baud rate errors.
PEN	10	rw	**Phase Error Enable** Can only be accessed when EN=0 (programming mode). Invalid data when EN=1 (operating mode). 0_B **IGNORE**, phase errors. 1_B **CHECK**, phase errors.
REN	9	rw	**Receive Error Enable** Can only be accessed when EN=0 (programming mode). Invalid data when EN=1 (operating mode). 0_B **IGNORE**, receive errors. 1_B **CHECK**, receive errors.
TEN	8	rw	**Transmit Error Enable** Can only be accessed when EN=0 (programming mode). Invalid data when EN=1 (operating mode). 0_B **IGNORE**, transmit errors. 1_B **CHECK**, transmit errors.
LB	7	rw	**Loop Back Control** Can only be accessed when EN=0 (programming mode). Invalid data when EN=1 (operating mode). 0_B **NORMAL**, output. 1_B **LB**, Receive input is connected with transmit output (half-duplex mode).
PO	6	rw	**Clock Polarity Control** Can only be accessed when EN=0 (programming mode). Invalid data when EN=1 (operating mode). 0_B **LOW**, Idle clock line is low, leading clock edge is low-to-high transition. 1_B **HIGH**, Idle clock line is high, leading clock edge is high-to-low transition.
PH	5	rw	**Clock Phase Control** Can only be accessed when EN=0 (programming mode). Invalid data when EN=1 (operating mode). 0_B **SHIFT**, transmit data on the leading clock edge, latch on trailing edge. 1_B **LATCH**, receive data on leading clock edge, shift on trailing edge.
HB	4	rw	**Heading Control** Can only be accessed when EN=0 (programming mode). Invalid data when EN=1 (operating mode). 0_B **LSB**, Transmit/Receive LSB First. 1_B **MSB**, Transmit/Receive MSB First.

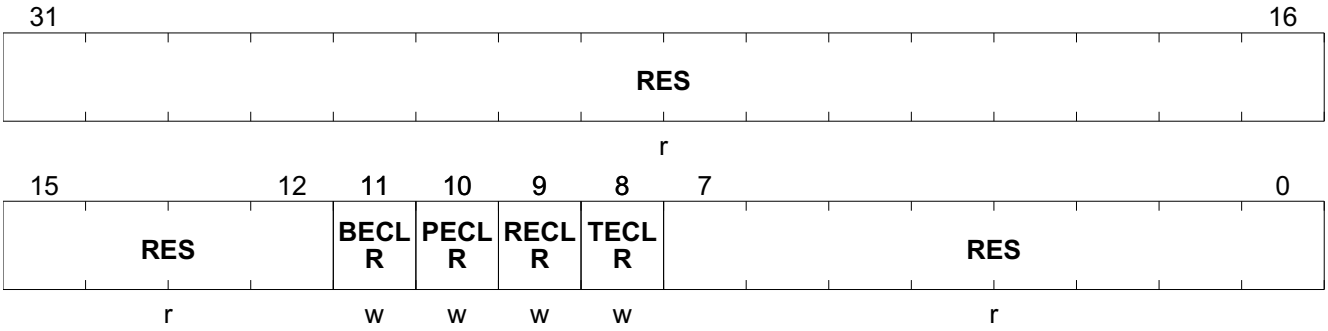
Field	Bits	Type	Description
BM	3:0	rw	**Data Width Selection** Can only be accessed when EN=0 (programming mode). Invalid data when EN=1 (operating mode). 0000_B**Reserved**, Do not use this combination. 0001_B**2**, Transfer Data Width is 2 (BM+1). 1111_B**16**, Transfer Data Width is 16 bits (BM+1).

Table 369 RESET of SSC_CON

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Interrupt Status Register Clear

SSC_ISRCLR	Offset	Reset Value
Interrupt Status Register Clear	14_H	see Table 370



Field	Bits	Type	Description
RES	31:12	r	**Reserved** Returns 0 if read; should be written with 0.
BECLR	11	w	**Baud Rate Error Flag Clear** 0_B **NO**, No error clear. 1_B **CLEAR**, Error clear.
PECLR	10	w	**Phase Error Flag Clear** 0_B **NO**, No error clear. 1_B **CLEAR**, Error clear.
RECLR	9	w	**Receive Error Flag Clear** 0_B **NO**, No error clear. 1_B **CLEAR**, Error clear.
TECLR	8	w	**Transmit Error Flag Clear** 0_B **NO**, No error clear. 1_B **CLEAR**, Error clear.
RES	7:0	r	**Reserved** Always read as 0

Table 370 RESET of SSC_ISRCLR

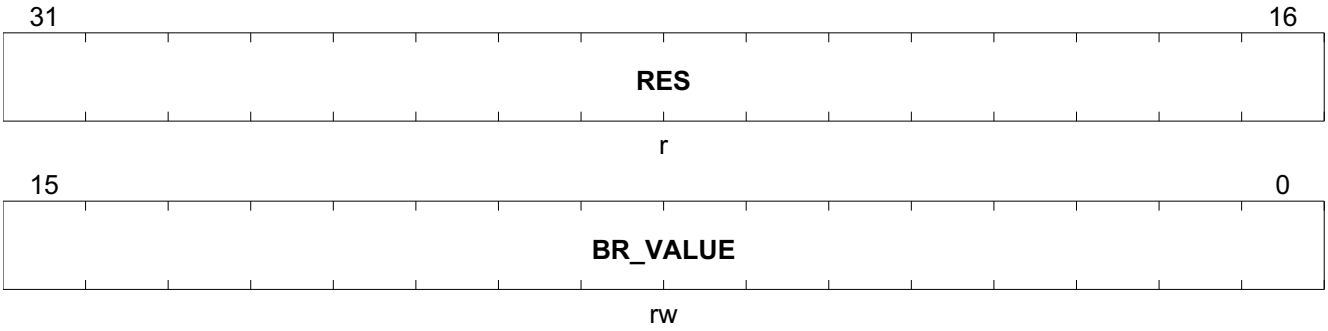
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

21.5.3 Baud Rate Timer Reload Register

The SSC baud rate timer reload register BR contains the 16-bit reload value for the baud rate timer.

Baud Rate Timer Reload Register

SSC_BR	Offset	Reset Value
Baud Rate Timer Reload Register	10_H	see Table 371



Field	Bits	Type	Description
RES	31:16	r	**Reserved** Returns 0 if read; should be written with 0.
BR_VALUE	15:0	rw	**Baud Rate Timer/Reload Register Value** Reading BR returns the 16-bit contents of the baud rate timer. Writing BR loads the baud rate timer reload register with BR_VALUE.

Table 371 RESET of SSC_BR

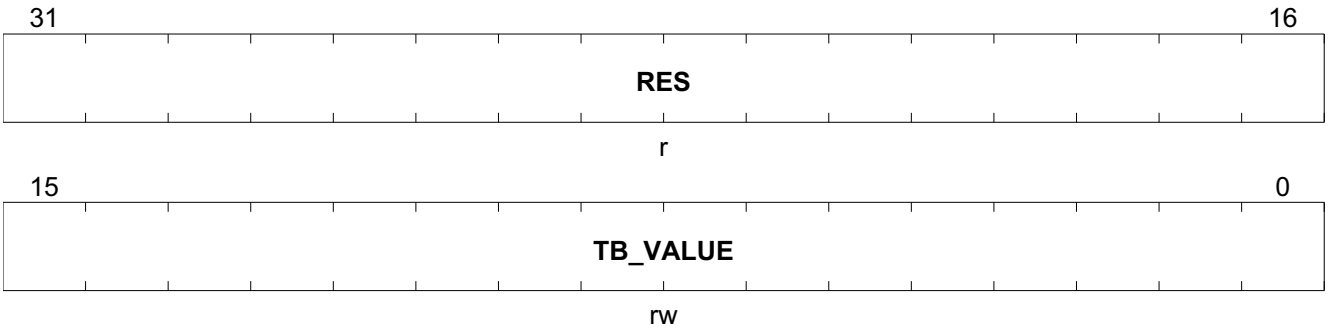
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

21.5.4 Transmitter Buffer Register

Transmitter Buffer Register

The SSC transmitter buffer register TB contains the transmit data value.

SSC_TB	Offset	Reset Value
Transmitter Buffer Register	08_H	Table 372



Field	Bits	Type	Description
RES	31:16	r	**Reserved** Returns 0 if read; should be written with 0.
TB_VALUE	15:0	rw	**Transmit Data Register Value** TB_VALUE is the data value to be transmitted. Unselected bits of TB are ignored during transmission.

Table 372 RESET of SSC_TB

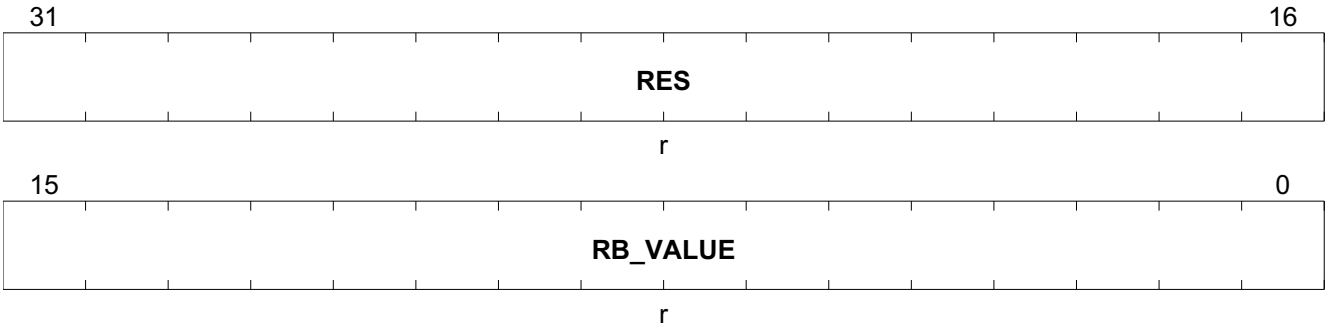
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

21.5.5 Receiver Buffer Register

Receiver Buffer Register

The SSC receiver buffer register RB contains the receive data value.

SSC_RB	Offset	Reset Value
Receiver Buffer Register	$0C_H$	see Table 373



Field	Bits	Type	Description
RES	31:16	r	**Reserved** Returns 0 if read; should be written with 0.
RB_VALUE	15:0	r	**Receive Data Register Value** RB contains the received data value RB_VALUE. Unselected bits of RB will be not valid and should be ignored.

Table 373 RESET of SSC_RB

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

21.6 Output multiplexing

In case the multiplexed SSC-Port (SSC12_*) should be used, the outputs can be selected (from SSC1 or from SSC2). Please use the bits SSC_* in register SCU_MODPISEL for this purpose.

22 Measurement Unit

22.1 Features

- 1 x 10-bit ADC with 12 inputs
- Supply Voltage Attenuators with attenuation of **VBAT_SENSE**, **VS**, **MONx**, **P2.x**, **CSA**.
- 1 x 8-bit ADC with 9 inputs
- Supply Voltage Attenuators with attenuation of **VS, VDDEXT, VSD, VCP, VDDP, VBG, VDDC, T_SENSE1 (Central Temperature Sensor), T_SENSE2 (Bridge Driver Charge Pump Temperature Sensor)**.
- Monitoring of PMU bandgap by 8-bit ADC to support functional safety requirements.
- Temperature Sensor to monitor the chip temperature and Bridge Driver Charge Pump temperature.
- Supplement Block with Reference Voltage Generation, Bias Current Generation, Voltage Buffer for NVM Reference Voltage, Voltage Buffer for Analog Module Reference Voltage and Test Interface.

22.2 Introduction

The measurement unit is a functional unit that comprises the following associated sub-modules:

Table 374 Measurement functions and associated modules

Module Name	Modules	Functions
Central Functions Unit	Bandgap reference circuit + current reference circuit	The bandgap-reference sub-module provides two reference voltages 1. an accurate reference voltage for the 10-bit and 8-bit ADCs. A local dedicated bandgap circuit is implemented to avoid deterioration of the reference voltage caused e.g. by crosstalk or ground voltage shift. 2. the reference voltage for the NVM module
10-bit ADC (ADC1)	10-bit ADC module with 12 multiplexed analog inputs	VBAT_SENSE, VS and MONx measurement. Five (5V) analog inputs from Port 2.x
8-bit ADC (ADC2)	8-bit ADC module with 9 multiplexed inputs	VS/VDDEXT/VSD/VCP/VDDP/VBG/VDDC/BDrv CP Temperature Sensor and Central Temperature Sensor measurement.
Temperature Sensor	Temperature sensor readout amplifier with two multiplexed ΔV_{be}-sensing elements	Generates output voltage which is a linear function of the local chip (Tj) temperature.
Measurement Core Module	Digital signal processing and ADC control unit	1. Generates the control signal for the 8-bit ADC 2 and the synchronous clock for the switched capacitor circuits (temperature sensor) 2. Performs digital signal processing functions and provides status outputs for interrupt generation.

22.2.1 Block Diagram

The Structure of the Measurement Functions Module is shown in the following figure.

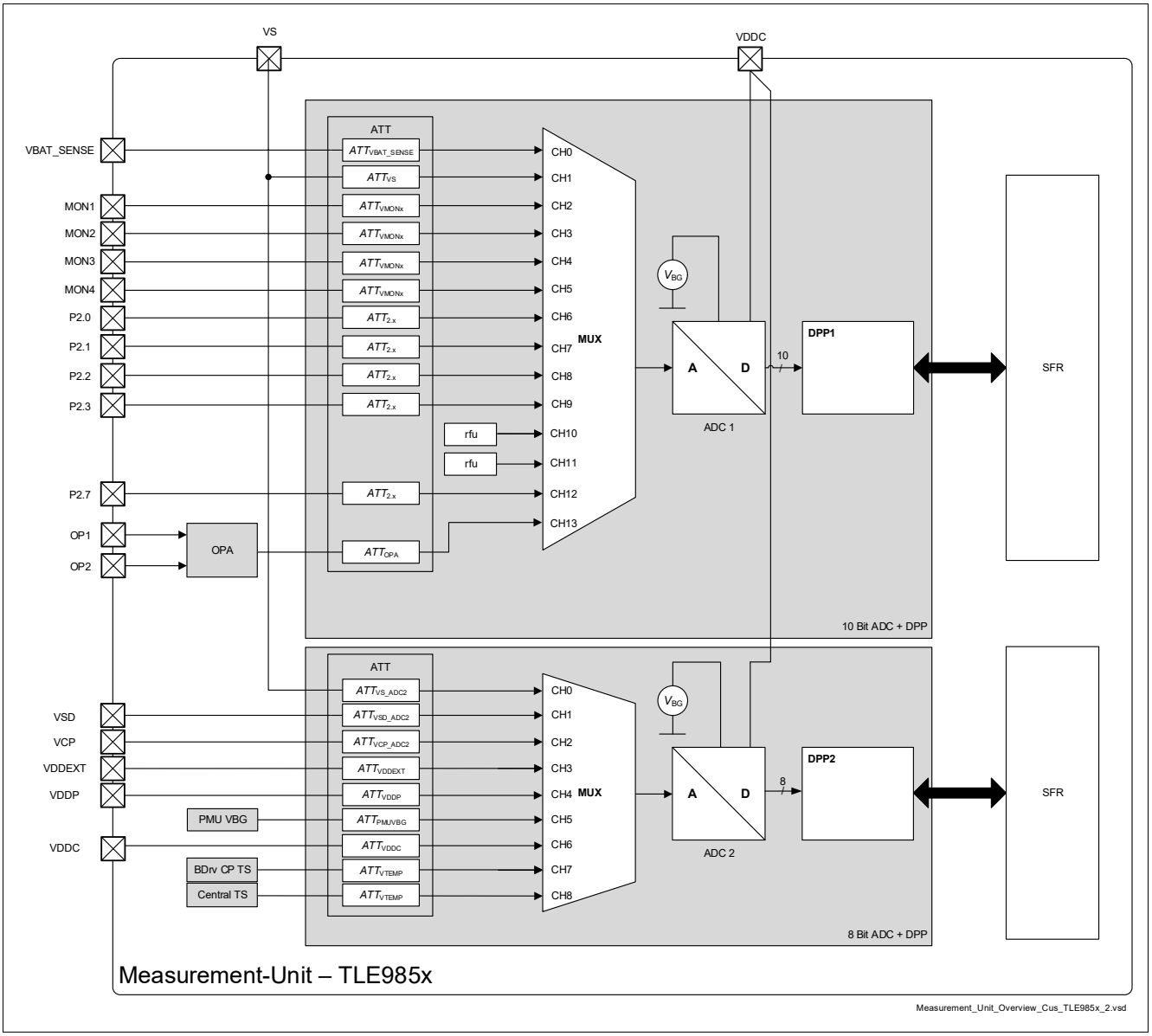


Figure 187 Measurement Unit-Overview

22.2.2 Measurement Unit Register Overview

Table 375 Register Address SpaceAddress Space for Measurement Unit Registers

Module	Base Address	End Address	Note
MF	48018000_H	$4801BFFF_H$	Measurement Unit

Table 376 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Supplement Modules Control and Status Register,			
MF_REF1_STS	Reference 1 Status Register	14_H	$0000\ 00C1_H$

The registers are addressed wordwise.

22.3 8-bit - 10 Channel ADC Core

The 8-bit ADC Core operates at the VDDC Supply Voltage. This enables the user to operate the measurement system down to reset threshold. The ADC can also be operated independently from the DPP unit. This enables the user to build up a software controlled measurement cycle. The main features of the 8-bit ADC core are listed below.

Module Features

- Conversion time = 15 system clock cycles.
- programmable sampling time (4 to 22 MCLK cycles, default: 12)
- Scalable clock frequency from 10 - 30 MHz.

The next chapter shows the channel allocation of the 8-bit ADC Core.

22.3.1 Transfer Characteristics of ADC2

The transfer function of ADC2 can be expressed by the equation below:

(22.1)

$$ADC2out = floor\left(\frac{Vin * Gain_{CHx}}{Vlsb} + 1\right)$$

where Vin is the input voltage and $Gain_{CHx}$ the individual Channel Gain. The LSB Voltage is calculated:

(22.2)

$$Vlsb = \frac{Vref}{256}$$

where **Vref** is **1,21 V @ 27 °C**.

A detailed specification of both A/D-converters is given in Chapter **Electrical Characteristics**. The Gain for each channel can be found in the table included in the following chapter.

22.3.2 ADC2 Measurement Channel- and Control Register Description

For more detailed description please refer to **Measurement Core Module (incl. ADC2)**

22.4 10-bit - 14 Channel ADC Core

The 10-bit ADC is using Port 2.x, MON's, CSA, VS and Vbat_sense as inputs. The configuration possibilities of the input channels are described in **Analog Digital Converter ADC10B (ADC1)**

22.4.1 Transfer Characteristics of ADC1

The transfer function of ADC1 can be expressed by the equation below:

(22.3)

$$ADC1out = floor\left(\frac{Vin * Gain_{CHx}}{Vlsb} + 1\right)$$

where Vin is the input voltage and $Gain_{CHx}$ the individual Channel Gain. The LSB Voltage is calculated:

(22.4)

$$Vlsb = \frac{Vref}{1024}$$

where **Vref** is **1,21 V @ 27 °C**.

A detailed specification of both A/D-converters is given in Chapter **Electrical Characteristics**. The Gain for each channel can be found in the table included in the following chapter.

22.5 Central and Charge Pump Temperature Sensor

This module is a quasi combination of a main on-chip temperature sensor and a charge pump temperature sensor.

Module Features

- 2 operation modes with
  Mode 1 - temperature range corresponds to differential output voltage range 0 …1.2V (output voltage shift enabled), resolution approximately 10°C.
  Mode 2 - temperature range corresponds to differential output voltage range 0.6 …1.2V, resolution approx. 15°C.
- The combined system temperature sensor plus ADC can be calibrated in software using calibration figures that are stored in the NVM at the production test.

This temperature sensor, including two sensing elements, monitors the chip temperature and PMU Regulator temperature. One sensing element is placed in the centre of the device to get the average device temperature status and the other sensing element is close to the PMU Regulator.

The voltage calculation of the Temperature is done with the following formula:

(22.5)

$$ADC2out = floor\left(\frac{Vtemp}{Vlsb} + 1\right)$$

The LSB Voltage is calculated by:

(22.6)

$$Vlsb = \frac{Vref}{256}$$

Vtemp depends on the absolute temperature *T* (given in K) and is calculated by:

(22.7)

$$V_{tEMP}(T) = a + b * (T - T_0)$$

For the **coefficients a and b please refer to the electrical characteristsics.T_0 =273 K**:

22.6 Supplement Modules

The purpose of the supplement modules is to enable a certain infrastructure on the device to guarantee a fail safe operation:

Module Features

- Bandgap Reference Voltage with accuracy ± 1.5%.
- Bandgap is monitored by an independent reference voltage.
- ADC1 Reference with accuracy ± 1%.
- ADC1 Reference has overload detection.

The next chapter lists the configuration possibilities of the on chip references.

22.6.1 Functional Safety Concept

8-bit ADC Module 2

- A known voltage, e.g. reference voltage of the main supply module, is periodically measured as part of the measurement sequence in normal operation. (The local ADC's reference voltage can, of course, not be used for this purpose since a local reference voltage error would not be detectable.)
- The conversion result of the functional safety measurement is evaluated in the postprocessing unit. If the results is not within the expected range an error is indicated.

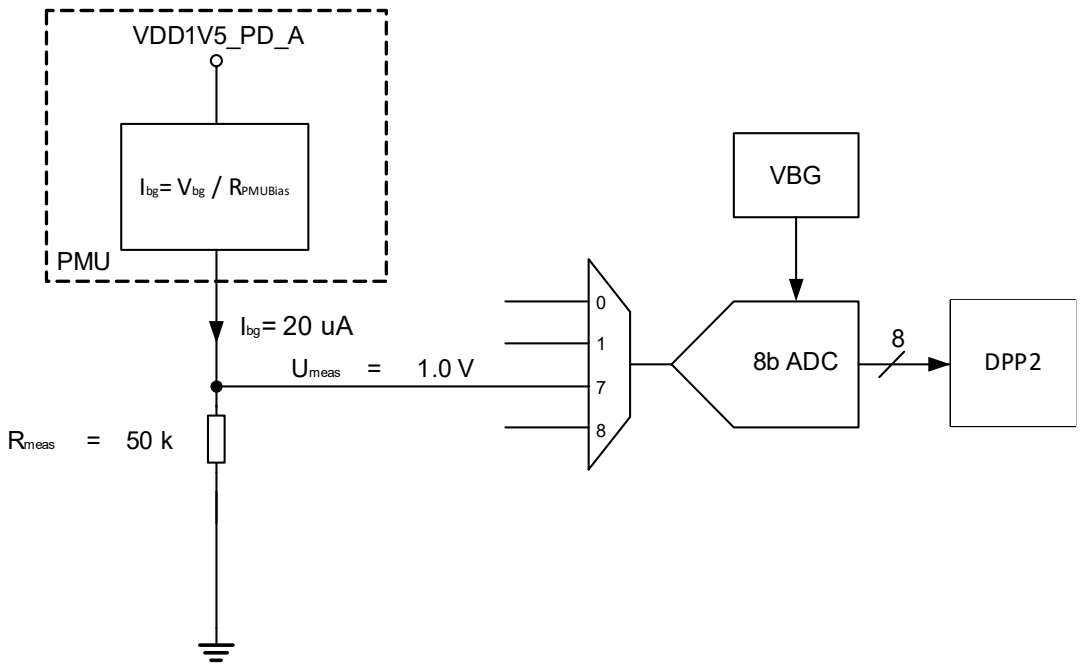


Figure 188 Principle of PMU Bandgap Measurement

22.6.2 Supplement Modules Control and Status Register

The next chapter lists the diagnosis and configuration possibilities of the supplement modules.

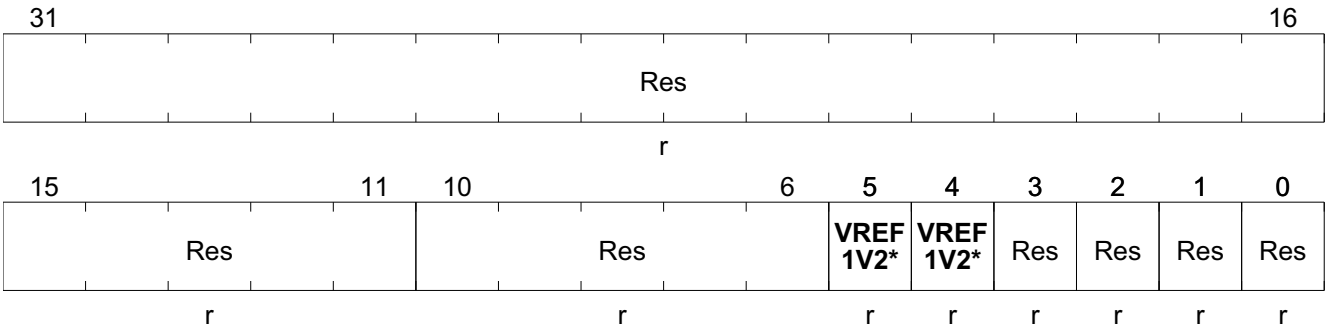
Table 377 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Supplement Modules Control and Status Register,			
MF_REF1_STS	Reference 1 Status Register	14_H	0000 00C1$_H$

The registers are addressed bytewise.

Reference 1 Status Register

MF_REF1_STS	Offset	Reset Value
Reference 1 Status Register	14_H	see Table 378



Field	Bits	Type	Description
Res	31:11	r	**Reserved** Always read as 0
Res	10:6	r	**Reserved** Always read as 1
VREF1V2_UPTHWARN_STS	5	r	**Status for Overvoltage Threshold Measurement of internal VAREF** 0_B **UPPER_TRIG_RESET**, write clears status 1_B **UPPER_TRIG_SET**, trigger status set
VREF1V2_LOTHWARN_STS	4	r	**Status for Undervoltage Threshold Measurement of internal VAREF** 0_B **UPPER_TRIG_RESET**, write clears status 1_B **UPPER_TRIG_SET**, trigger status set
Res	3	r	**Reserved** Always read as 0
Res	2	r	**Reserved** Always read as 0

Field	Bits	Type	Description
Res	1	r	**Reserved** Always read as 0
Res	0	r	**Reserved** Always read as 0

Table 378 RESET of MF_REF1_STS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$000000C1_H$	RESET_TYPE_3		

23 Measurement Core Module (incl. ADC2)

23.1 Features

- 9 individually programmable channels split into two groups of user configurable and non user configurable
- Individually programmable channel prioritization scheme for measurement unit
- Two independent filter stages with programmable low-pass and time filter characteristics for each channel
- Two channel configurations:
  - Programmable upper- and lower trigger thresholds comprising a fully programmable hysteresis
  - Two individually programmable trigger thresholds with limit hysteresis settings
- Status for all channel thresholds
- Operation down to reset threshold of entire system

23.2 Introduction

The basic function of this block is the digital postprocessing of several analog digitized measurement signals by means of filtering, level comparison and interrupt generation. The measurement postprocessing block is built of nine identical channel units attached to the outputs of the 9-channels 8-bit ADC (ADC2). It processes nine channels, where the channel sequence and prioritization is programmable within a wide range.

23.2.1 Block Diagram

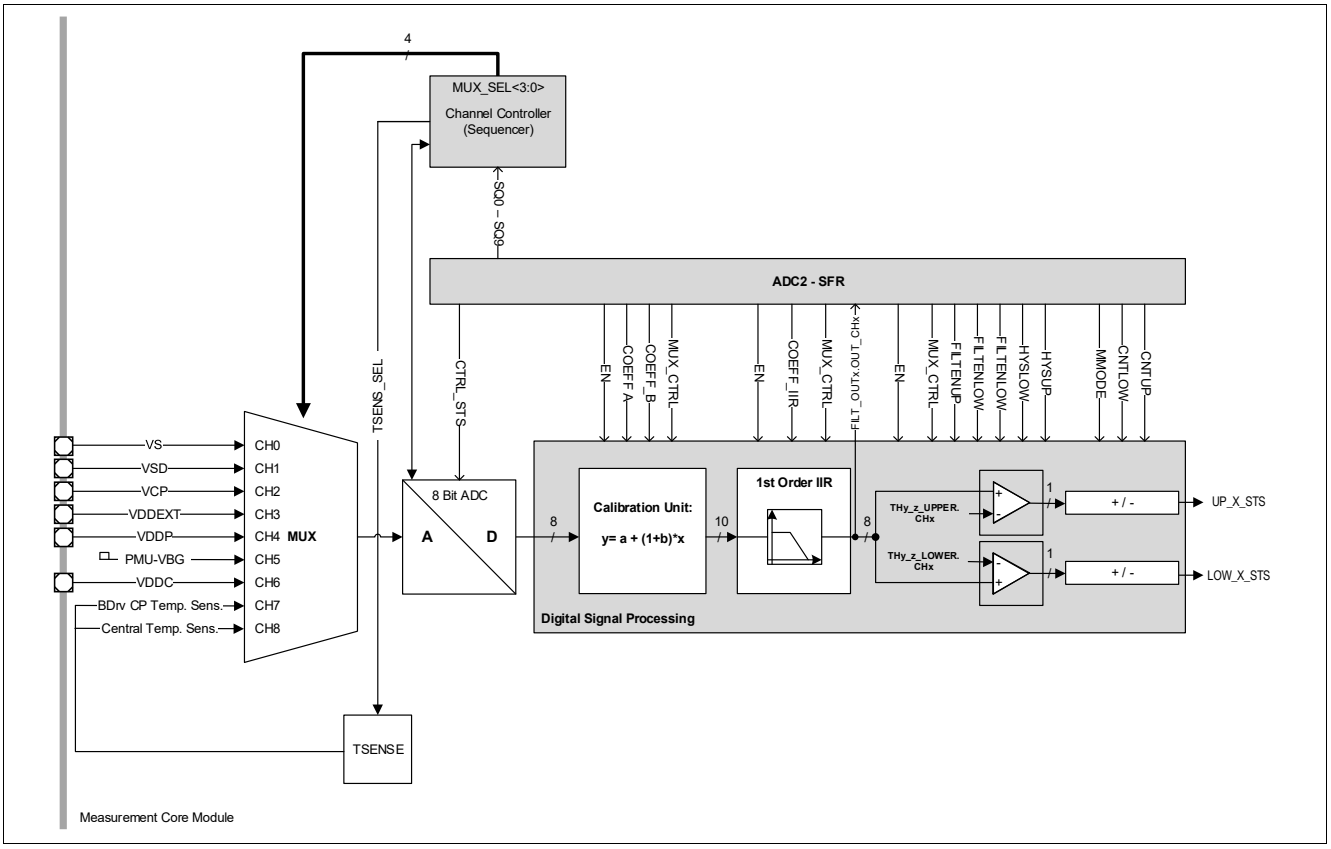


Figure 189 Module Block Diagram

23.2.2 Measurement Core Module Modes Overview

The basic function of this unit is the digital signal processing of several analog digitized measurement signals by means of filtering level comparison and interrupt generation. The Measurement Core module processes nine channels in a quasi parallel evaluation process.

As shown in the figure above, the ADC2 postprocessing consists of a channel controller (Sequencer), a 9-channel demultiplexer and the signal processing block, which filters and compares the sampled ADC2 values for each channel individually. The channel control block controls the multiplexer sequencing on the analog side before the ADC2 and on the digital domain after the ADC2. As described in the following section, the channel sequence can be controlled in a flexible way, which allows a certain degree of channel prioritization.

This capability can be used e.g. to set a higher priority to supply voltage channels compared to the other channel measurements. The Measurement Core Module offers additionally two different post-processing measurement modes for over-/undervoltage detection and for two-level threshold detection.

Usually the external register settings should only be changed during the start-up phase (**ADC2_CTRL2**).

“Software Mode”, Sequencer and Exceptional Interrupt Measurement is disabled, each measurement is triggered by software.

The IIR filter can be bypassed via **ADC2_FILT_UPLO_CTRL** for the data transferred to postprocessing only.

The threshold counter can be bypassed (counting only 1 measurement) via CNT_LO_CHx.

23.3 ADC2 - Core (8-bit ADC)

23.3.1 Functional Description

The different sequencer modes are controlled by SFR Register:

- “Normal Sequencer Mode” described in the Chapter **Channel Controller**.
- “Exceptional Interrupt Measurement” (EIM), upon hardware event, the channel programmed in **ADC2_CHx_EIM** is inserted after the current measurement is finished. Afterwards the current sequence will be continued with the next measurement from the current sequence.
- “Software Mode”, in Software Mode the control of the Channel Controller (Sequencer) is disabled, instead the conversions are fully controlled by software. EIM hardware events are ignored during Debug Suspend Mode, they are pending during Software Mode when entered by an empty sequence.

Software Mode:

- Software mode can be entered
  - by clearing one of the sequence registers SQn (e.g. SQ_{1_2}[9:0]) or
  - using Debug Suspend Mode
- In Software mode, the channel selection by the Sequencer is disabled. After the software mode is entered, the conversions are controlled via **ADC2_CTRL_STS**.
- The Software Mode is left
  - when the maximum time is reached (maximum time specified in **ADC2_MAX_TIME**) or
  - when the sequence which started the software mode is reprogrammed with at least one channel set in registers SQn (e.g. to SQ_{1_2}[9:0])
  - leaving Debug Suspend Mode

Software Mode:

In Software Mode, measurements are triggered by writing the **ADC2_CTRL_STS**.SOS bit. This bit is active as long as the conversion is in progress. The user polls the **ADC2_CTRL_STS**.EOC bit. Once this bit is ‘1’ the conversion is finished and the EOC bit is cleared on read (rh). After the EOC bit is cleared, a new conversion can be started with **ADC2_CTRL_STS**.SOS.

Debug Suspend Mode:

During Debug Suspend Mode the Sequencer is stopped once the current measurement is finished (after the next EOC event) and Software Mode is entered. As long as the Debug Suspend Mode is active no measurements are performed by the Sequencer. Once the Debug Suspend Mode is left, the Sequencer continues immediately with the next pending measurement.
Measurements can be still triggered in Debug Suspend Mode/Software Mode. The maximum time of Software Mode is disabled in Suspend Mode. EIM events are ignored during Debug Suspend Mode.

The ADC2 timing is controlled by SFR Register

- Sample time adjustment described in the register **ADC2_CTRL2**.

23.3.2 ADC2 Control Registers

The ADC2 is fully controllable by the below listed sfr Registers. The control must be enabled by setting all sequencer bits to zero. . To enable the sequencer again this corresponding bits in the sequencer register must be set to one again.

Table 379 shows the module base addresses.

Table 379 Register Address Space

Module	Base Address	End Address	Note
ADC2	$4801C000_H$	$4801DFFF_H$	ADC2 - ADC-SAR8B

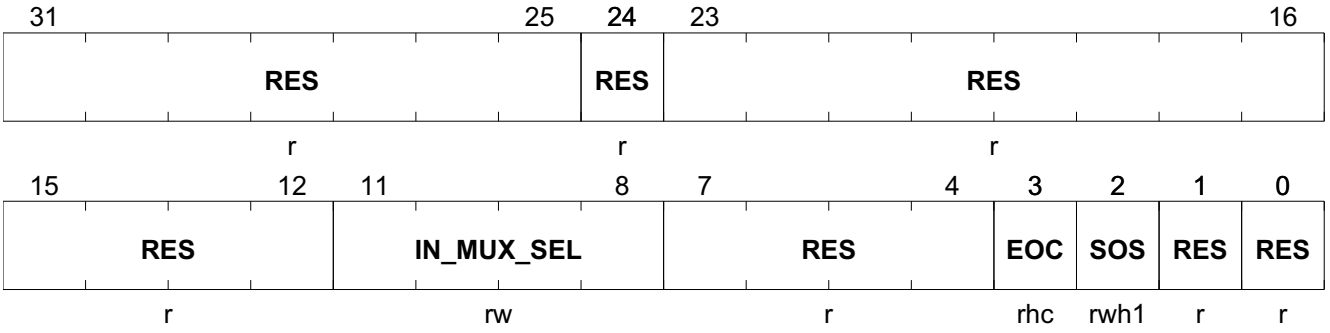
Table 380 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
ADC2 Control Registers,			
ADC2_CTRL_STS	ADC2 Control and Status Register	00_H	see **Table 381**
ADC2_STATUS	ADC2 HV Status Register	BC_H	see **Table 382**

The registers are addressed wordwise.

ADC2 Control Register

ADC2_CTRL_STS	Offset	Reset Value
ADC2 Control and Status Register	00_H	see Table 381



Field	Bits	Type	Description
RES	31:25	r	**Reserved** Always read as 0
RES	24	r	**Reserved** Always read as 0
RES	23:12	r	**Reserved** Always read as 0
IN_MUX_SEL	11:8	rw	**Channel for software mode** Other bit combinations are **reserved**, do not use. 0000_B **CH0_EN**, Channel 0 enable 0001_B **CH1_EN**, Channel 1 enable 0010_B **CH2_EN**, Channel 2 enable 0011_B **CH3_EN**, Channel 3 enable 0100_B **CH4_EN**, Channel 4 enable 0101_B **CH5_EN**, Channel 5 enable 0110_B **CH6_EN**, Channel 6 enable 0111_B **CH7_EN**, Channel 7 enable 1000_B **CH8_EN**, Channel 8 enable 1001_B **rfu**, reserved for future use 1111_B **rfu**, reserved for future use
RES	7:4	r	**Reserved** Always read as 0
EOC	3	rhc	**ADC2 End of Conversion in software mode** *Note: this flag is not only cleared by a read operation but also automatically by setting SOS* 0_B **Pending**, conversion still running 1_B **Finished**, conversion has finished

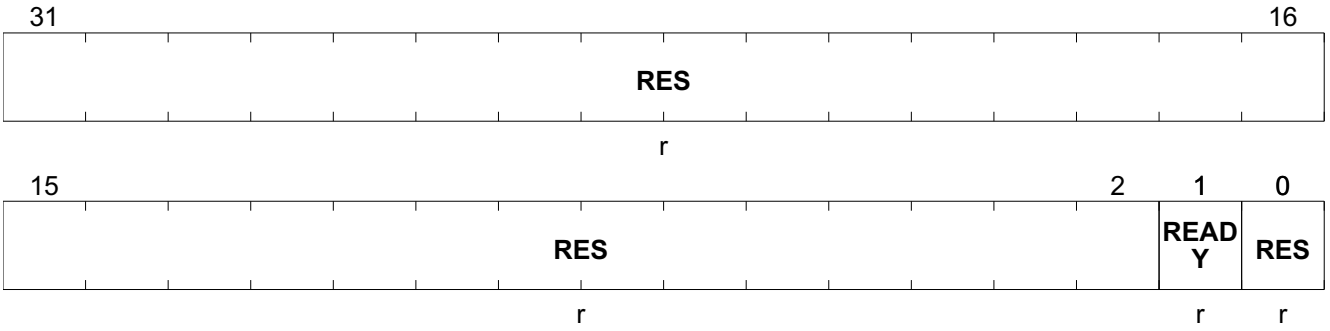
Field	Bits	Type	Description
SOS	2	rwh1	**ADC2 Start of Sampling/Conversion (software mode)** *Note: Bit is set by software to start sampling and conversion and it is cleared by hardware once the conversion is finished ADC2_SOC can be only written if the DPP is in software mode.* 0_B **Disable**, no conversion is started 1_B **Enable**, conversion is started
RES	1	r	**Reserved** Always read as 0
RES	0	r	**Reserved** Always read as 0

Table 381 RESET of ADC2_CTRL_STS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000001_H	RESET_TYPE_3		

ADC2 HV Status Register

ADC2_STATUS	Offset	Reset Value
ADC2 HV Status Register	BC_H	see Table 382



Field	Bits	Type	Description
RES	31:2	r	**Reserved** Always read as 0
READY	1	r	**HVADC Ready bit** 0_B **Not ready**, Module in power down or in init phase 1_B **Ready**, set automatically 5 ADC clock cycles after module is enabled
RES	0	r	**Reserved** Always read as 0

Table 382 RESET of ADC2_STATUS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

23.4 Channel Controller

23.4.1 Functional Description

The task of each channel controller is a prioritization of the individual measurement channels. The sequencing scheme is illustrated in the example of following table and can be programmed individually for measurement unit.

Table 383 Measurement channel sequence definition example (used as default sequence)

Measurement channel n	CH8	CH7	CH6	CH5	CH4	CH3	CH2	CH1	LSB CH0
Registers {SQ_{0_1}[8:0]}	0	1	1	0	1	1	1	1	1
Registers {SQ_{0_1}[24:16]}	1	0	0	1	0	0	1	1	0
Registers {SQ_{2_3}[8:0]}	0	1	1	0	1	1	1	1	0
Registers {SQ_{2_3}[24:16]}	1	0	0	1	0	0	1	1	1
Registers {SQ_{4_5}[8:0]}	0	1	1	0	1	1	1	1	0
Registers {SQ_{4_5}[24:16]}	1	0	0	1	0	0	1	1	0
Registers {SQ_{6_7}[8:0]}	0	1	1	0	1	1	1	1	1
Registers {SQ_{6_7}[24:16]}	1	0	0	1	0	0	1	1	0
Registers {SQ_{8_9}[8:0]}	0	1	1	0	1	1	1	1	0

The sequence registers SQ_n define the time sequence of the measurement channels by the following rules:

- The sequence registers define the measurement sequence and are evaluated from register 0 to 8 and for each register from MSB to LSB, which defines a max. overall measurement periodicity of 81 sampling and conversion cycles.
- If the individual bit in the sequence register is set to ’1’, the corresponding channel is measured.
- If the individual bit in the sequence register is not set, this measurement phase is skipped.

In the upper example, the resulting channel sequence is defined as:

CH7, CH6, CH4, CH3, CH2, CH1, CH0, CH8, CH5, CH2, CH1,....., CH7, CH6, CH4, CH3, CH2, CH1

In TLE985xQX channels 0 - 8 can be fully programmed. They are measured depending on the amount of ’1’ bits, written in the sequence registers. The following equations can be used to calculate the periodicity of the required channel measurement.

The overall measurement periodicity of all measurements in A/D conversion cycles is defined as:

$$\overline{N_{meas}} = \sum_{m=1}^{10}\left(\sum_{n=1}^{10} SQ_m[n]\right) \tag{23.1}$$

The average measurement periodicity of channel n in A/D conversion cycles is defined as

$$\overline{N_{meas,n}} = \frac{\left(\sum_{m=1}^{10} SQ_m[n]\right)}{\overline{T_{meas}}} \quad (23.2)$$

The timing of the analog MUX and the digital DEMUX is controlled by the channel controller accordingly. The analog MUX with sample and hold stage needs one clock cycle for channel switching and the ADC consumes, as default setting, 12 clock cycles for the sampling of the input voltage. The conversion time for a single channel measurement value is 10 clock cycles.

The minimum measurement periodicity, which can be achieved, by enabling only channel 1 in the sequence registers, depends on the MI_CLK frequency and is given by:

$$\overline{T_{meas\_Chl\_min}} = \frac{32}{f_{MI\_CLK}}$$

This following calculations include already the sampling time of ADC2. If all programmable channels are enabled, the maximum periodicity is calculated: (23.3)

$$\overline{T_{meas\_Chl\_max}} = \frac{320}{f_{MI\_CLK}} \quad (23.4)$$

For a MI_CLK frequency of 24 MHz, the channel 1 is measured with min. 4 µs. The maximum update time of channel 1 with 24 MHz clock frequency is 10 µs. As mentioned before, this is calculated with the assumption, that all channels are enabled and channel1 is enabled in every sequence register. As a prerequisite for this calculation we take **ADC2_CTRL2**.SAMPLE_TIME_int = 4.

23.4.2 Channel Controller Control Registers

The Channel Controller can be configured by the **SFR** Register listed in **Table 384**. The registers which cannot be written by the user have the attribute **rwpt**.

Table 384 Register Overview

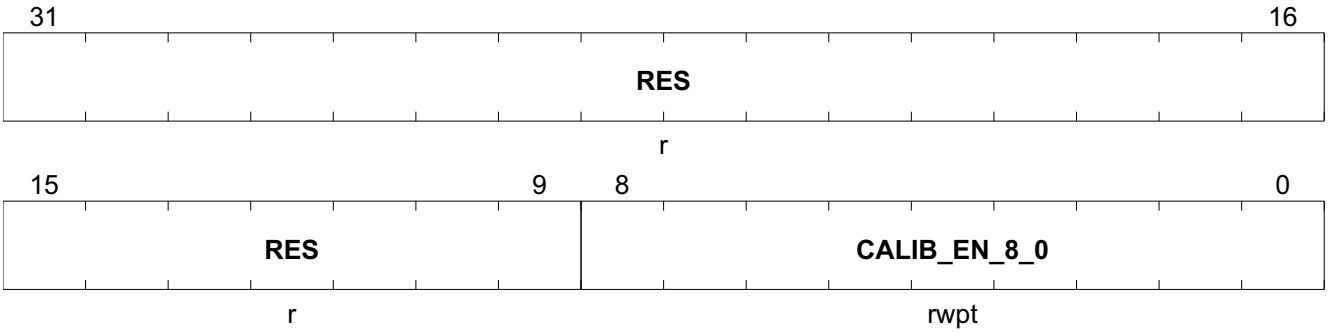
Register Short Name	Register Long Name	Offset Address	Reset Value
Channel Controller Control Registers,			
ADC2_SQ_FB	Sequencer Feedback Register	04_H	see **Table 393**
ADC2_CHx_EIM	Channel Settings Bits for Exceptional Interrupt Measurement	08_H	see **Table 394**
ADC2_MAX_TIME	Maximum Time for Software Mode	10_H	see **Table 395**
ADC2_CTRL1	Measurement Unit Control Register 1	14_H	see **Table 385**
ADC2_CTRL2	Measurement Unit Control Register 2	18_H	see **Table 386**
ADC2_CTRL4	Measurement Unit Control Register 4	$1C_H$	see **Table 387**
ADC2_SQ0_1	Measurement Channel Enable Bits for Sequence 0-1	20_H	see **Table 388**
ADC2_SQ4_5	Measurement Channel Enable Bits for Sequence 4 - 5	24_H	see **Table 390**
ADC2_SQ2_3	Measurement Channel Enable Bits for Sequence 2-3	28_H	see **Table 389**
ADC2_SQ6_7	Measurement Channel Enable Bits for Sequence 6 - 7	$2C_H$	see **Table 391**
ADC2_SQ8_9	Measurement Channel Enable Bits for Sequence 8	30_H	see **Table 392**

The registers are addressed wordwise.

Measurement Unit Control Register 1

This register is dedicated for controlling the calibration unit of the measurement core module. The respective channel calibration can be enabled or disabled by the bits listed below.

ADC2_CTRL1	Offset	Reset Value
Measurement Unit Control Register 1	**14_H**	**see Table 385**



Field	Bits	Type	Description
RES	31:9	r	**Reserved** Always read as 0
CALIB_EN_8_0	8:0	rwpt	**Calibration Enable for Channels 8 to 0** The following values can be ored: $0\ 0000\ 0001_B$**CH0_EN**, Channel 0 calibration enable $0\ 0000\ 0010_B$**CH1_EN**, Channel 1 calibration enable $0\ 0000\ 0100_B$**CH2_EN**, Channel 2 calibration enable $0\ 0000\ 1000_B$**CH3_EN**, Channel 3 calibration enable $0\ 0001\ 0000_B$**CH4_EN**, Channel 4 calibration enable $0\ 0010\ 0000_B$**CH5_EN**, Channel 5 calibration enable $0\ 0100\ 0000_B$**CH6_EN**, Channel 6 calibration enable $0\ 1000\ 0000_B$**CH7_EN**, Channel 7 calibration enable $1\ 0000\ 0000_B$**CH8_EN**, Channel 8 calibration enable

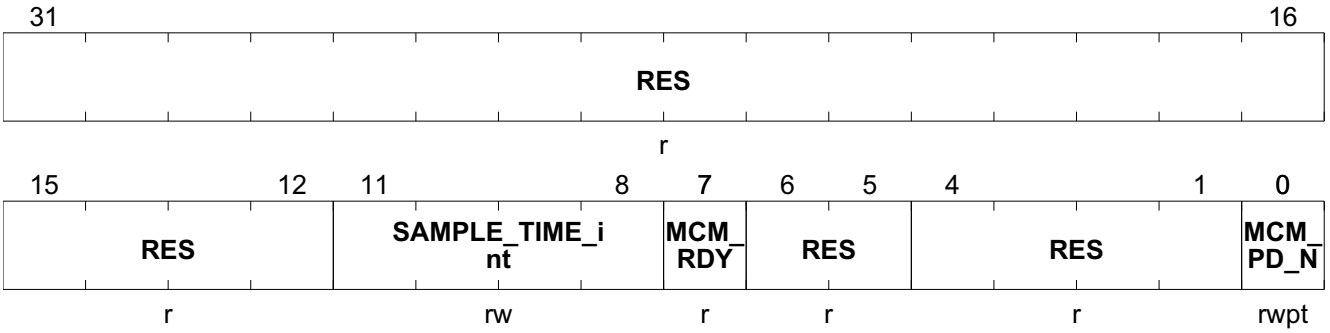
Table 385 RESET of ADC2_CTRL1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		
TRIM_2	$000001FF_H$	COMMON		

Measurement Unit Control Register 2

This register is used for controlling the calibration unit of channels 0-9 of the measurement core module. This register is protected for the purpose mentioned at the beginning of this chapter. Furthermore this register contains the sample time adjustment for ADC2. The default value is 12 clock cycles. Values above 12 clock cycles are not recommended, because they increase the overall response time of the measurement system.

ADC2_CTRL2	Offset	Reset Value
Measurement Unit Control Register 2	18_H	see Table 386



Field	Bits	Type	Description
RES	31:16	r	**Reserved** Always read as 0
RES	15:12	r	**Reserved** Always read as 0
SAMPLE_TIME_int	11:8	rw	**Sample time of ADC2** 0_H **MICLK4**, 4 MI_CLK clock periods 1_H **MICLK6**, 6 MI_CLK clock periods 2_H **MICLK8**, 8 MI_CLK clock periods 3_H **MICLK10**, 10 MI_CLK clock periods 4_H **MICLK12**, 12 MI_CLK clock periods (default) 5_H **MICLK14**, 14 MI_CLK clock periods 6_H **MICLK16**, 16 MI_CLK clock periods 7_H **MICLK18**, 18 MI_CLK clock periods 8_H **MICLK20**, 20 MI_CLK clock periods 9_H **MICLK22**, 22 MI_CLK clock periods A_H **n.u.**, not used B_H **n.u.**, not used C_H **n.u.**, not used D_H **n.u.**, not used E_H **n.u.**, not used F_H **n.u.**, not used
MCM_RDY	7	r	**Ready Signal for MCM[1] after Power On or Reset** 0_B **MCM Not Ready**, Measurement Core Module in startup phase 1_B **MCM Ready**, Measurement Core Module start-up phase finished
RES	6:5	r	**Reserved** Always read as 0
RES	4:1	r	**Reserved** Always read as 0
MCM_PD_N	0	rwpt	**Power Down Signal for MCM** 0_B **MCM Disabled**, Measurement Core Module disabled 1_B **MCM Enabled**, Measurement Core Module enabled

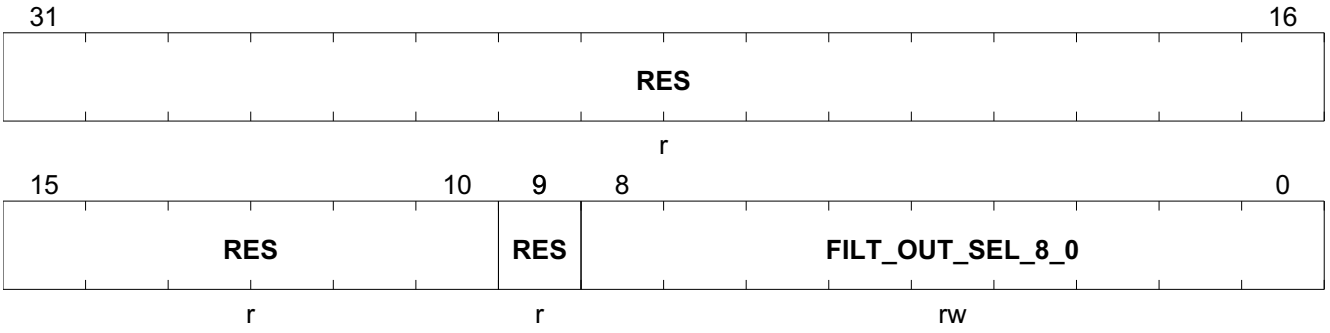
1) MCM = Measurement Core Module

Table 386 RESET of ADC2_CTRL2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000401_H	RESET_TYPE_4		
TRIM_2	00000401_H	RESET		

Measurement Unit Control Register 4

ADC2_CTRL4	Offset	Reset Value
Measurement Unit Control Register 4	$1C_H$	see Table 387



Field	Bits	Type	Description
RES	31:10	r	**Reserved** Always read as 0
RES	9	r	**Reserved** Always read as 0

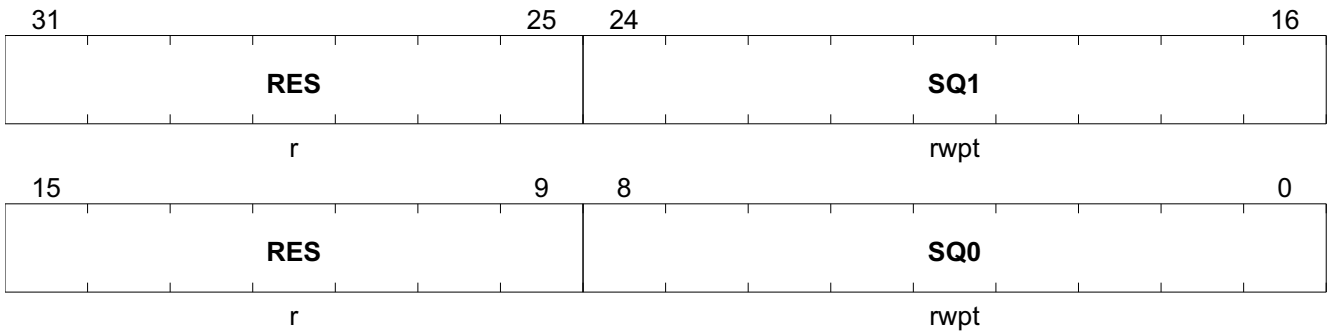
Field	Bits	Type	Description
FILT_OUT_SEL_8_0	8:0	rw	**Output Filter Selection for Channels 0 to 8** Each bit enables the IIR filter for the corresponding channel. 0 0000 0000_B**ADC2 Unfiltered Data can be monitored in the corresponding ADC2_FILT_OUTx Registers**, . 0 0000 0001_B**Channel 0 IIR Data enabled for ADC2_FILT_OUT0 Register**, . 0 0000 0010_B**Channel 1 IIR Data enabled for ADC2_FILT_OUT1 Register**, . 0 0000 0100_B**Channel 2 IIR Data enabled for ADC2_FILT_OUT2 Register**, . 0 0000 1000_B**Channel 3 IIR Data enabled for ADC2_FILT_OUT3 Register**, . 0 0001 0000_B**Channel 4 IIR Data enabled for ADC2_FILT_OUT4 Register**, . 0 0010 0000_B**Channel 5 IIR Data enabled for ADC2_FILT_OUT5 Register**, . 0 0100 0000_B**Channel 6 IIR Data enabled for ADC2_FILT_OUT6 Register**, . 0 1000 0000_B**Channel 7 IIR Data enabled for ADC2_FILT_OUT7 Register**, . 1 0000 0000_B**Channel 8 IIR Data enabled for ADC2_FILT_OUT8 Register**, . 1 1111 1111_B**Channels 8-0 IIR Data enabled for ADC2_FILT_OUTx Register**, .

Table 387 RESET of ADC2_CTRL4

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$000001FF_H$	RESET_TYPE_4		
TRIM_2	$000001FF_H$	RESET		

Measurement Channel Enable Bits for Sequence 0-1

ADC2_SQ0_1	Offset	Reset Value
Measurement Channel Enable Bits for Sequence 0-1	**20_H**	**see Table 388**



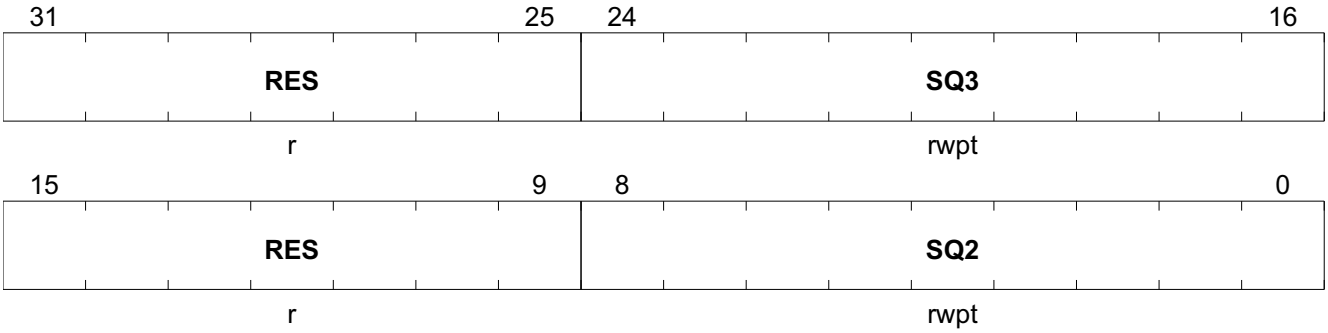
Field	Bits	Type	Description
RES	31:25	r	**Reserved** Always read as 0
SQ1	24:16	rwpt	**Sequence 1 channel enable** The following values can be ored: $0\ 0000\ 0001_B$**CH0_EN**, Channel 0 enable $0\ 0000\ 0010_B$**CH1_EN**, Channel 1 enable $0\ 0000\ 0100_B$**CH2_EN**, Channel 2 enable $0\ 0000\ 1000_B$**CH3_EN**, Channel 3 enable $0\ 0001\ 0000_B$**CH4_EN**, Channel 4 enable $0\ 0010\ 0000_B$**CH5_EN**, Channel 5 enable $0\ 0100\ 0000_B$**CH6_EN**, Channel 6 enable $0\ 1000\ 0000_B$**CH7_EN**, Channel 7 enable $1\ 0000\ 0000_B$**CH8_EN**, Channel 8 enable
RES	15:9	r	**Reserved** Always read as 0
SQ0	8:0	rwpt	**Sequence 0 channel enable** The following values can be ored: $0\ 0000\ 0001_B$**CH0_EN**, Channel 0 enable $0\ 0000\ 0010_B$**CH1_EN**, Channel 1 enable $0\ 0000\ 0100_B$**CH2_EN**, Channel 2 enable $0\ 0000\ 1000_B$**CH3_EN**, Channel 3 enable $0\ 0001\ 0000_B$**CH4_EN**, Channel 4 enable $0\ 0010\ 0000_B$**CH5_EN**, Channel 5 enable $0\ 0100\ 0000_B$**CH6_EN**, Channel 6 enable $0\ 1000\ 0000_B$**CH7_EN**, Channel 7 enable $1\ 0000\ 0000_B$**CH8_EN**, Channel 8 enable

Table 388 RESET of ADC2_SQ0_1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$012600DF_H$	RESET_TYPE_4		
TRIM_2	$012600DF_H$	RESET		

Measurement Channel Enable Bits for Sequence 2-3

ADC2_SQ2_3	Offset	Reset Value
Measurement Channel Enable Bits for Sequence 2-3	**28_H**	**see Table 389**



Field	Bits	Type	Description
RES	31:25	r	**Reserved** Always read as 0
SQ3	24:16	rwpt	**Sequence 3 channel enable** The following values can be ored: $0\ 0000\ 0001_B$**CH0_EN**, Channel 0 enable $0\ 0000\ 0010_B$**CH1_EN**, Channel 1 enable $0\ 0000\ 0100_B$**CH2_EN**, Channel 2 enable $0\ 0000\ 1000_B$**CH3_EN**, Channel 3 enable $0\ 0001\ 0000_B$**CH4_EN**, Channel 4 enable $0\ 0010\ 0000_B$**CH5_EN**, Channel 5 enable $0\ 0100\ 0000_B$**CH6_EN**, Channel 6 enable $0\ 1000\ 0000_B$**CH7_EN**, Channel 7 enable $1\ 0000\ 0000_B$**CH8_EN**, Channel 8 enable
RES	15:9	r	**Reserved** Always read as 0
SQ2	8:0	rwpt	**Sequence 2 channel enable** The following values can be ored: $0\ 0000\ 0001_B$**CH0_EN**, Channel 0 enable $0\ 0000\ 0010_B$**CH1_EN**, Channel 1 enable $0\ 0000\ 0100_B$**CH2_EN**, Channel 2 enable $0\ 0000\ 1000_B$**CH3_EN**, Channel 3 enable $0\ 0001\ 0000_B$**CH4_EN**, Channel 4 enable $0\ 0010\ 0000_B$**CH5_EN**, Channel 5 enable $0\ 0100\ 0000_B$**CH6_EN**, Channel 6 enable $0\ 1000\ 0000_B$**CH7_EN**, Channel 7 enable $1\ 0000\ 0000_B$**CH8_EN**, Channel 8 enable

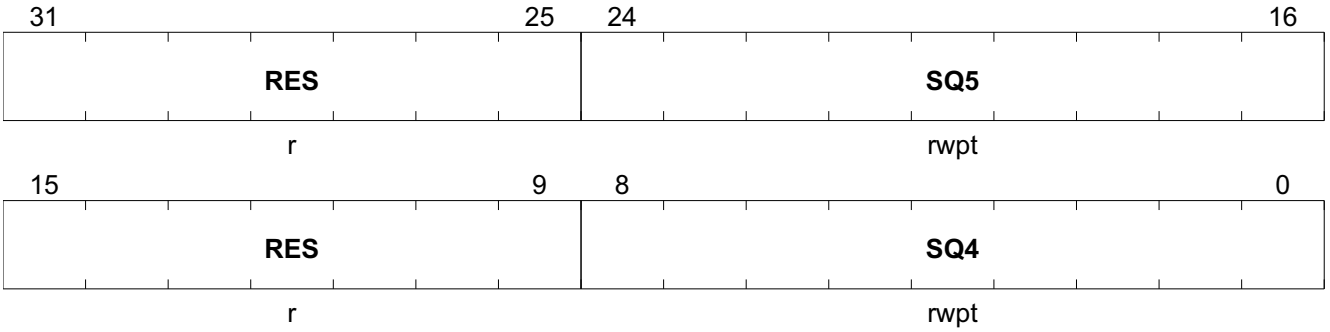
Table 389 RESET of ADC2_SQ2_3

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$012700DE_H$	RESET_TYPE_4		
TRIM_2	$012700DE_H$	RESET		

Measurement Core Module (incl. ADC2)

Measurement Channel Enable Bits for Sequence 4-5

ADC2_SQ4_5	Offset	Reset Value
Measurement Channel Enable Bits for Sequence 4 - 5	**24_H**	see Table 390



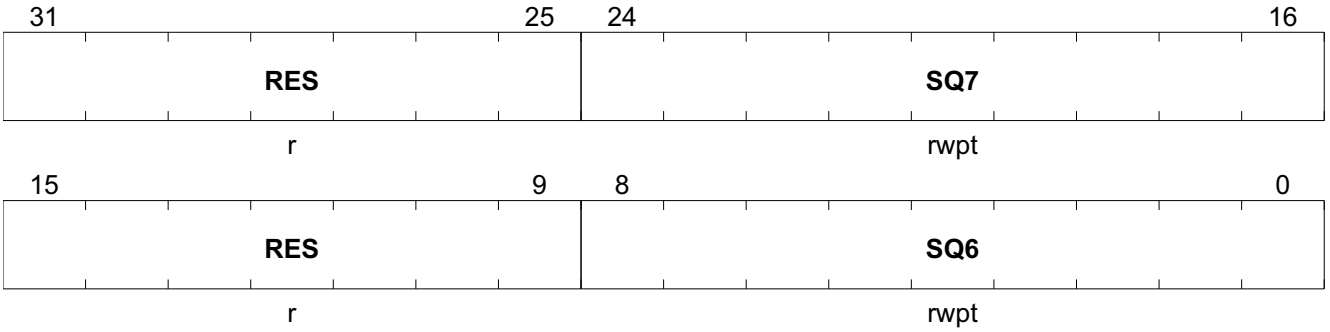
Field	Bits	Type	Description
RES	31:25	r	**Reserved** Always read as 0
SQ5	24:16	rwpt	**Sequence 5 channel enable** The following values can be ored: $0\ 0000\ 0001_B$**CH0_EN**, Channel 0 enable $0\ 0000\ 0010_B$**CH1_EN**, Channel 1 enable $0\ 0000\ 0100_B$**CH2_EN**, Channel 2 enable $0\ 0000\ 1000_B$**CH3_EN**, Channel 3 enable $0\ 0001\ 0000_B$**CH4_EN**, Channel 4 enable $0\ 0010\ 0000_B$**CH5_EN**, Channel 5 enable $0\ 0100\ 0000_B$**CH6_EN**, Channel 6 enable $0\ 1000\ 0000_B$**CH7_EN**, Channel 7 enable $1\ 0000\ 0000_B$**CH8_EN**, Channel 8 enable
RES	15:9	r	**Reserved** Always read as 0
SQ4	8:0	rwpt	**Sequence 4 channel enable** The following values can be ored: $0\ 0000\ 0001_B$**CH0_EN**, Channel 0 enable $0\ 0000\ 0010_B$**CH1_EN**, Channel 1 enable $0\ 0000\ 0100_B$**CH2_EN**, Channel 2 enable $0\ 0000\ 1000_B$**CH3_EN**, Channel 3 enable $0\ 0001\ 0000_B$**CH4_EN**, Channel 4 enable $0\ 0010\ 0000_B$**CH5_EN**, Channel 5 enable $0\ 0100\ 0000_B$**CH6_EN**, Channel 6 enable $0\ 1000\ 0000_B$**CH7_EN**, Channel 7 enable $1\ 0000\ 0000_B$**CH8_EN**, Channel 8 enable

Table 390 RESET of ADC2_SQ4_5

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$012600DE_H$	RESET_TYPE_4		
TRIM_2	$012600DE_H$	RESET		

Measurement Channel Enable Bits for Sequence 6-7

ADC2_SQ6_7	Offset	Reset Value
Measurement Channel Enable Bits for Sequence 6 - 7	$2C_H$	see Table 391



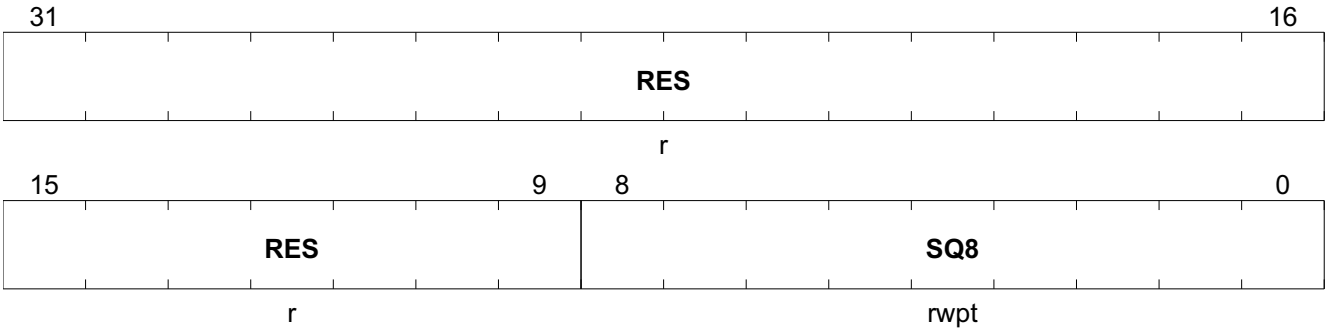
Field	Bits	Type	Description
RES	31:25	r	**Reserved** Always read as 0
SQ7	24:16	rwpt	**Sequence 7 channel enable** The following values can be ored: $0\ 0000\ 0001_B$**CH0_EN**, Channel 0 enable $0\ 0000\ 0010_B$**CH1_EN**, Channel 1 enable $0\ 0000\ 0100_B$**CH2_EN**, Channel 2 enable $0\ 0000\ 1000_B$**CH3_EN**, Channel 3 enable $0\ 0001\ 0000_B$**CH4_EN**, Channel 4 enable $0\ 0010\ 0000_B$**CH5_EN**, Channel 5 enable $0\ 0100\ 0000_B$**CH6_EN**, Channel 6 enable $0\ 1000\ 0000_B$**CH7_EN**, Channel 7 enable $1\ 0000\ 0000_B$**CH8_EN**, Channel 8 enable
RES	15:9	r	**Reserved** Always read as 0
SQ6	8:0	rwpt	**Sequence 6 channel enable** The following values can be ored: $0\ 0000\ 0001_B$**CH0_EN**, Channel 0 enable $0\ 0000\ 0010_B$**CH1_EN**, Channel 1 enable $0\ 0000\ 0100_B$**CH2_EN**, Channel 2 enable $0\ 0000\ 1000_B$**CH3_EN**, Channel 3 enable $0\ 0001\ 0000_B$**CH4_EN**, Channel 4 enable $0\ 0010\ 0000_B$**CH5_EN**, Channel 5 enable $0\ 0100\ 0000_B$**CH6_EN**, Channel 6 enable $0\ 1000\ 0000_B$**CH7_EN**, Channel 7 enable $1\ 0000\ 0000_B$**CH8_EN**, Channel 8 enable

Table 391 RESET of ADC2_SQ6_7

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$012600DF_H$	RESET_TYPE_4		
TRIM_2	$012600DF_H$	RESET		

Measurement Channel Enable Bits for Sequence 8

ADC2_SQ8_9	Offset	Reset Value
Measurement Channel Enable Bits for Sequence 8	30_H	see Table 392



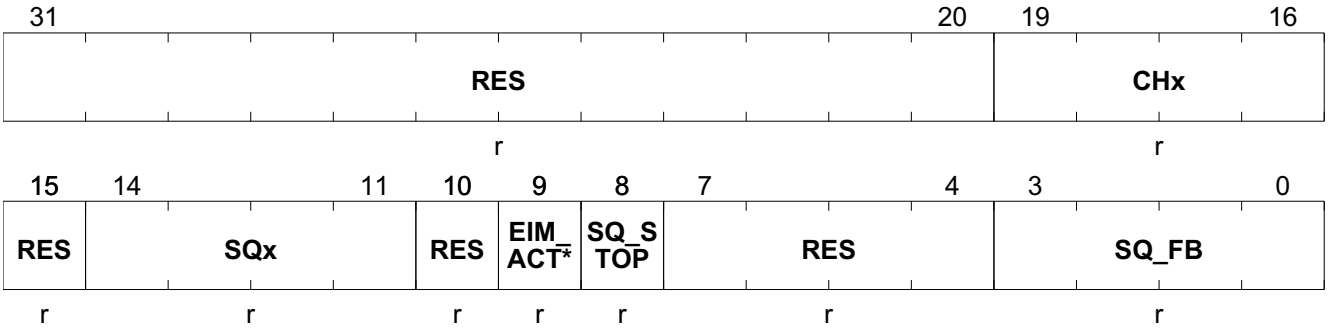
Field	Bits	Type	Description
RES	31:9	r	**Reserved** Always read as 0
SQ8	8:0	rwpt	**Sequence 8 channel enable** The following values can be ored: $0\ 0000\ 0001_B$**CH0_EN**, Channel 0 enable $0\ 0000\ 0010_B$**CH1_EN**, Channel 1 enable $0\ 0000\ 0100_B$**CH2_EN**, Channel 2 enable $0\ 0000\ 1000_B$**CH3_EN**, Channel 3 enable $0\ 0001\ 0000_B$**CH4_EN**, Channel 4 enable $0\ 0010\ 0000_B$**CH5_EN**, Channel 5 enable $0\ 0100\ 0000_B$**CH6_EN**, Channel 6 enable $0\ 1000\ 0000_B$**CH7_EN**, Channel 7 enable $1\ 0000\ 0000_B$**CH8_EN**, Channel 8 enable

Table 392 RESET of ADC2_SQ8_9

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$000000DE_H$	RESET_TYPE_4		
TRIM_2	$000000DE_H$	RESET		

Sequencer Feedback Register

ADC2_SQ_FB	Offset	Reset Value
Sequencer Feedback Register	04_H	see Table 393



Field	Bits	Type	Description
RES	31:20	r	**Reserved** Always read as 0
CHx	19:16	r	**Current active ADC2 Channel (in normal mode)** Other bit combinations are **reserved**, do not use. 0000_B **CH0**, Channel 0 0001_B **CH1**, Channel 1 0010_B **CH2**, Channel 2 0011_B **CH3**, Channel 3 0100_B **CH4**, Channel 4 0101_B **CH5**, Channel 5 0110_B **CH6**, Channel 6 0111_B **CH7**, Channel 7 1000_B **CH8**, Channel 8
RES	15	r	**Reserved** Always read as 0
SQx	14:11	r	**Current active ADC2 Sequence (in normal mode)** Other bit combinations are **reserved**, do not use. 0000_B **SQ0**, Sequence 0 0001_B **SQ1**, Sequence 1 0010_B **SQ2**, Sequence 2 0011_B **SQ3**, Sequence 3 0100_B **SQ4**, Sequence 4 0101_B **SQ5**, Sequence 5 0110_B **SQ6**, Sequence 6 0111_B **SQ7**, Sequence 7 1000_B **SQ8**, Sequence 8 1001_B **SQ9**, Startup sequence
RES	10	r	**Reserved** Always read as 0

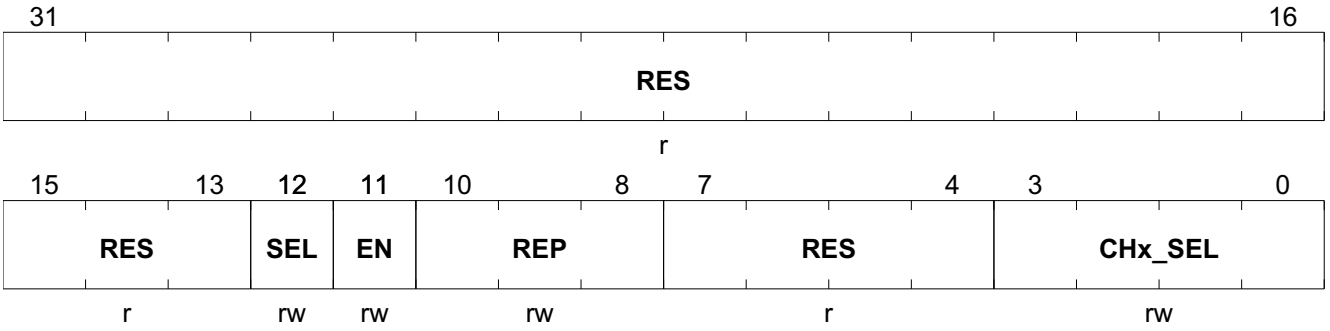
Field	Bits	Type	Description
EIM_ACTIVE	9	r	**ADC2 EIM active** 0_B **not active** , EIM not active 1_B **active**, EIM active
SQ_STOP	8	r	**ADC2 Sequencer Stop Signal for DPP** 0_B **DPP Running**, Postprocessing Sequencer in running mode 1_B **DPP Stopped**, Postprocessing Sequencer stopped / Software Mode entered
RES	7:4	r	**Reserved** Always read as 0
SQ_FB	3:0	r	**Current Sequence that caused software mode** Other bit combinations are **n.u.**, not used. *Note: When sw_maxtime = 0, the SQ_FB = 11 (CH_MASK) is not flagged, even when masked sequence is empty* 0000_B **SQ0**, Sequence 0 0001_B **SQ1**, Sequence 1 0010_B **SQ2**, Sequence 2 0011_B **SQ3**, Sequence 3 0100_B **SQ4**, Sequence 4 0101_B **SQ5**, Sequence 5 0110_B **SQ6**, Sequence 6 0111_B **SQ7**, Sequence 7 1000_B **SQ8**, Sequence 8 1011_B **CH_MASK**, Sequence was 0 only after masking; SWM not entered 1100_B **SUSPEND**, Debug Suspend Mode

Table 393 RESET of ADC2_SQ_FB

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000 0000 XXXX 0XXX X0XX 0000 $XXXX_B$	RESET_TYPE_3		

Channel Setting for Exceptional Interrupt Measurement

ADC2_CHx_EIM	Offset	Reset Value
Channel Settings Bits for Exceptional Interrupt Measurement	08_H	see Table 394



Field	Bits	Type	Description
RES	31:13	r	**Reserved** Always read as 0
SEL	12	rw	**Exceptional interrupt measurement (EIM) Trigger select** 0_B **GPT12PISEL.T3_GPT12_SEL**, Signal according to SCU_GPT12PISEL.T3_GPT12SEL setting 1_B **CP_clk**, Charge-pump clock
EN	11	rw	**Exceptional interrupt measurement (EIM) Trigger Event enable** 0_B **DISABLE**, start of EIM disabled 1_B **ENABLE**, start of IEM enabled
REP	10:8	rw	**Repeat count for exceptional interrupt measurement (EIM)** 000_B **1**, Measurement (minimum or continuous measurement as long as trigger signals stays high, 1 / continuous SOC generated for ADC8) 001_B **2**, Measurements (2 SOC generated for ADC8) 010_B **4**, Measurements (4 SOC generated for ADC8) 011_B **8**, Measurements (8 SOC generated for ADC8) 100_B **16**, Measurements (16 SOC generated for ADC8) 101_B **32**, Measurements (32 SOC generated for ADC8) 110_B **64**, Measurements (64 SOC generated for ADC8) 111_B **128**, Measurements (128 SOC generated for ADC8)
RES	7:4	r	**Reserved** Always read as 0

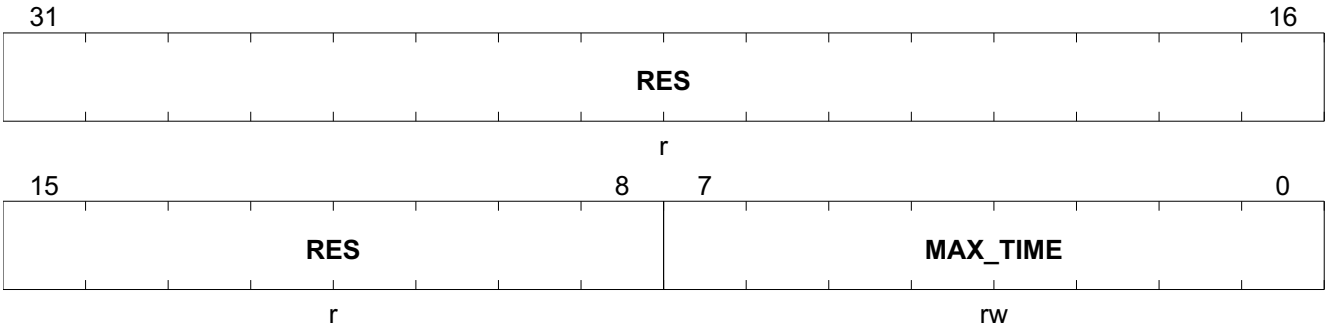
Field	Bits	Type	Description
CHx_SEL	3:0	rw	**Channel set for exceptional interrupt measurement (EIM)** Other bit combinations are **n.u.**, not used. 0000_B **CH0_EN**, Channel 0 enable 0001_B **CH1_EN**, Channel 1 enable 0010_B **CH2_EN**, Channel 2 enable 0011_B **CH3_EN**, Channel 3 enable 0100_B **CH4_EN**, Channel 4 enable 0101_B **CH5_EN**, Channel 5 enable 0110_B **CH6_EN**, Channel 6 enable 0111_B **CH7_EN**, Channel 7 enable 1000_B **CH8_EN**, Channel 8 enable 1001_B **rfu**, reserved for future use

Table 394 RESET of ADC2_CHx_EIM

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		
TRIM_2	00000000_H	RESET		

Maximum Time for Software Mode

ADC2_MAX_TIME	Offset	Reset Value
Maximum Time for Software Mode	10_H	see Table 395



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Always read as 0
MAX_TIME	7:0	rw	**Maximum Time in Software Mode** Maximum time in Software Mode with the unit of 1 clock cycle (typ. 50 ns). Software mode is active for (MAX_TIME + 2) * clock cycle 00_H **min**, Software mode is not entered 01_H **01**, Software mode is active for 3 clock cycles FF_H **max**, Software mode is active for 257 clock cycles (typ. 12.85 us)

Table 395 RESET of ADC2_MAX_TIME

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		
TRIM_2	00000000_H	RESET		

23.5 Calibration Unit

23.5.1 Functional Description

The calibration unit of the Measurement Core module is dedicated to cancel offset and gain errors out of the signal chain. The upcoming two chapters describe usage and setup of the calibration unit.

23.5.1.1 Method for determining the Calibration Parameters

As mentioned in the introduction of the calibration unit, the module can be used to correct gain and offset errors caused by non-idealities in the measurement chain. This non-idealities are caused by the corresponding measurement chain modules.

Those first order non-idealities are:

- Offset and Gain Error of ADC2.
- Offset and Gain Error of the Attenuator (especially voltage measurement).
- Offset and Gain Error of Reference Voltage.

All these factors are summed up in the overall Gain (factor **b**) and overall Offset (adder **a**) of the complete measurement chain. They are calculated from a two point test result and stored inside the NVM.

23.5.1.2 Setup of Calibration Unit

Each channel has its own calibration unit and thus also its dedicated Gain and Offset parameter. These parameters are stored in a 100TP page of the Flash Module. After each reset of RESET_TYPE_4 these coefficients are downloaded from NVM into the corresponding registers. The user may not take care about the configuration of these parameters. After this has been done, the values are used for the correction procedure. The figure below shows the formula performed by the calibration unit and the required **sfr**-Register to control its functionality in a generic way.

The parameters OFFS_CHx and GAIN_CHx are stored in a 5 bits resp. 8 bits, 2th complement format.

The function applied to calculate the calibrated ADC2 value is

ADC_cal_CHx = (1 + <GAIN_CHx>/256) * ADC_uncal_CHx + <OFFS_CHx>/2

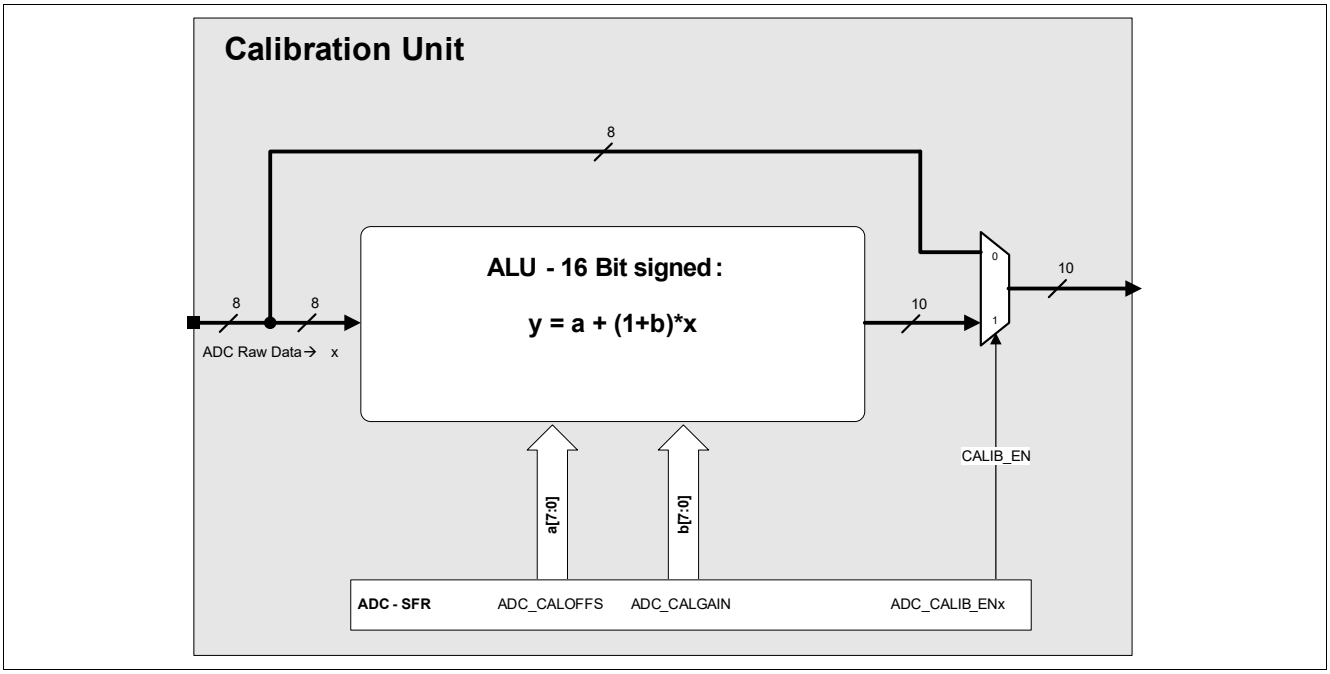


Figure 190 Structure of Calibration Unit

23.5.2 Calibration Unit Control Registers

The Calibration Unit can be configured by the **SFR** Registers shown below. The registers which cannot be written by the user have the attribute **rwpt**.

Table 396 Register Overview

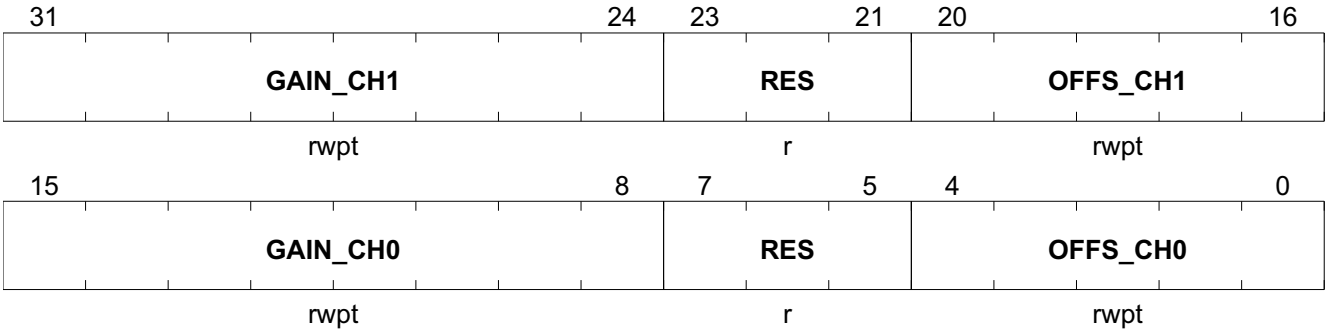
Register Short Name	Register Long Name	Offset Address	Reset Value
Calibration Unit Control Registers,			
ADC2_CAL_CH0_1	Calibration for Channel 0 and 1	34_H	see **Table 397**
ADC2_CAL_CH2_3	Calibration for Channel 2 and 3	38_H	see **Table 398**
ADC2_CAL_CH4_5	Calibration for Channel 4 and 5	$3C_H$	see **Table 399**
ADC2_CAL_CH6_7	Calibration for Channel 6 and 7	40_H	see **Table 400**
ADC2_CAL_CH8_9	Calibration for Channel 8	44_H	see **Table 401**

The registers are addressed wordwise.

ADC2 Calibration Value Channel 0 & 1

ADC2_CAL_CH0_1 **Offset** **Reset Value**

Calibration for Channel 0 & 1 **34_H** **see Table 397**



Field	Bits	Type	Description
GAIN_CH1	31:24	rwpt	**Gain Calibration for channel 1** For ADC output set CALIB_EN_1 = 0
RES	23:21	r	**Reserved** Always read as 0
OFFS_CH1	20:16	rwpt	**Offset Calibration for channel 1** For ADC output set CALIB_EN_1 = 0
GAIN_CH0	15:8	rwpt	**Gain Calibration for channel 0** For ADC output set CALIB_EN_0 = 0
RES	7:5	r	**Reserved** Always read as 0

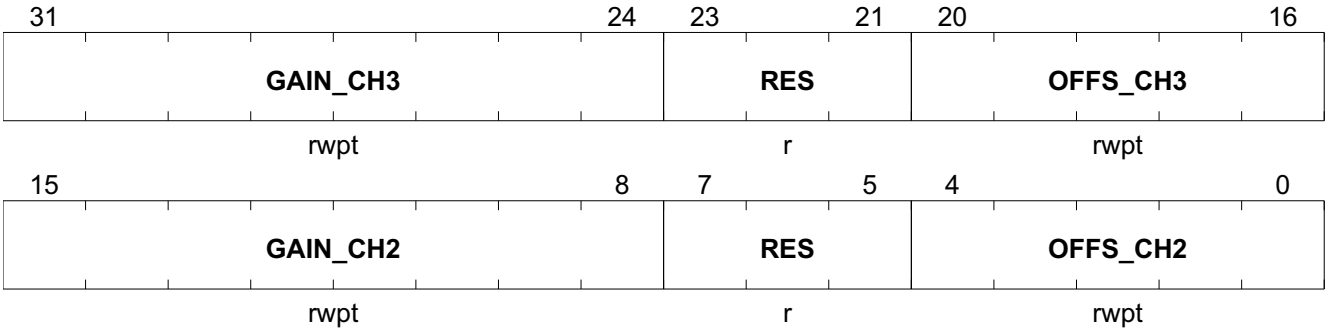
Field	Bits	Type	Description
OFFS_CH0	4:0	rwpt	**Offset Calibration for channel 0** For ADC output set CALIB_EN_0 = 0

Table 397 RESET of ADC2_CAL_CH0_1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		
TRIM_2	00000000_H	TRIM		

ADC2 Calibration Value Channel 2 & 3

ADC2_CAL_CH2_3	Offset	Reset Value
Calibration for Channel 2 & 3	38_H	see Table 398



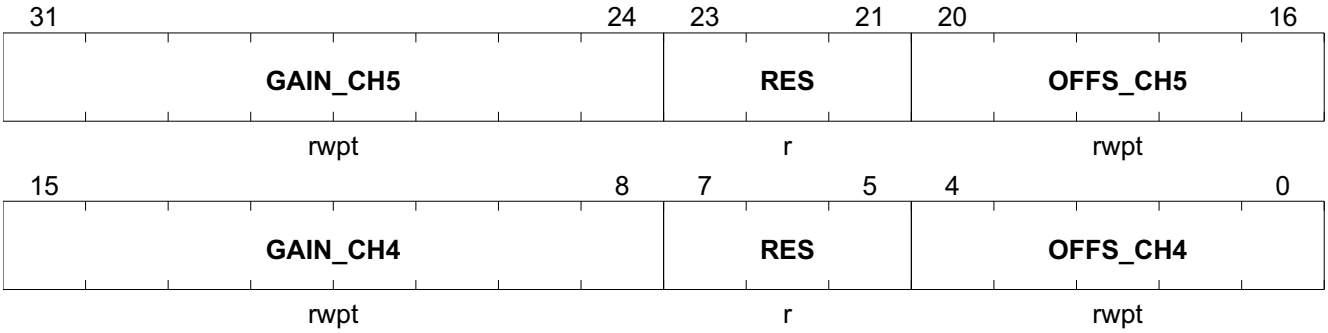
Field	Bits	Type	Description
GAIN_CH3	31:24	rwpt	**Gain Calibration for channel 3** For ADC output set CALIB_EN_3 = 0
RES	23:21	r	**Reserved** Always read as 0
OFFS_CH3	20:16	rwpt	**Offset Calibration for channel 3** For ADC output set CALIB_EN_3 = 0
GAIN_CH2	15:8	rwpt	**Gain Calibration for channel 2** For ADC output set CALIB_EN_2 = 0
RES	7:5	r	**Reserved** Always read as 0
OFFS_CH2	4:0	rwpt	**Offset Calibration for channel 2** For ADC output set CALIB_EN_2 = 0

Table 398 RESET of ADC2_CAL_CH2_3

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		
TRIM_2	00000000_H	TRIM		

ADC2 Calibration Value Channel 4 & 5

ADC2_CAL_CH4_5	Offset	Reset Value
Calibration for Channel 4 & 5	$3C_H$	see Table 399



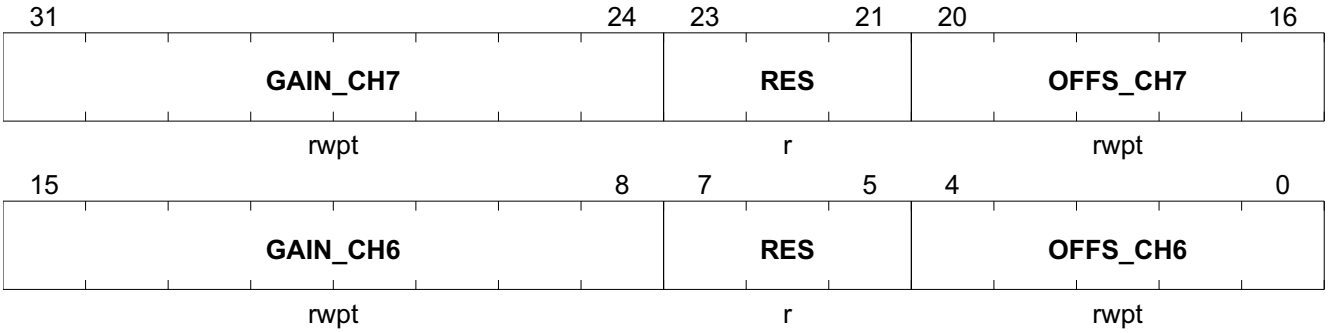
Field	Bits	Type	Description
GAIN_CH5	31:24	rwpt	**Gain Calibration for channel 5** For ADC output set CALIB_EN_5 = 0
RES	23:21	r	**Reserved** Always read as 0
OFFS_CH5	20:16	rwpt	**Offset Calibration for channel 5** For ADC output set CALIB_EN_5 = 0
GAIN_CH4	15:8	rwpt	**Gain Calibration for channel 4** For ADC output set CALIB_EN_4 = 0
RES	7:5	r	**Reserved** Always read as 0
OFFS_CH4	4:0	rwpt	**Offset Calibration for channel 4** For ADC output set CALIB_EN_4 = 0

Table 399 RESET of ADC2_CAL_CH4_5

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		
TRIM_2	00000000_H	TRIM		

ADC2 Calibration Value Channel 6 & 7

ADC2_CAL_CH6_7	Offset	Reset Value
Calibration for Channel 6 & 7	40_H	see Table 400



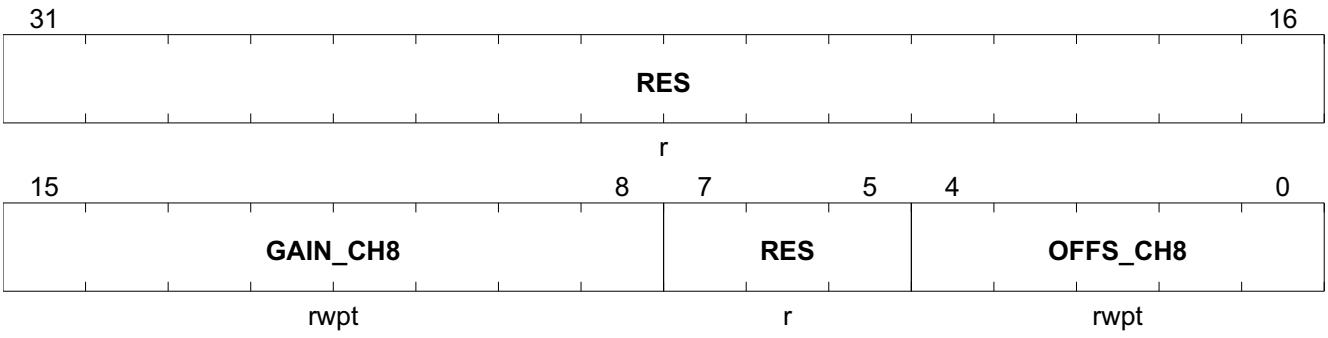
Field	Bits	Type	Description
GAIN_CH7	31:24	rwpt	**Gain Calibration for channel 7** For ADC output set CALIB_EN_7 = 0
RES	23:21	r	**Reserved** Always read as 0
OFFS_CH7	20:16	rwpt	**Offset Calibration for channel 7** For ADC output set CALIB_EN_7 = 0
GAIN_CH6	15:8	rwpt	**Gain Calibration for channel 6** For ADC output set CALIB_EN_6 = 0
RES	7:5	r	**Reserved** Always read as 0
OFFS_CH6	4:0	rwpt	**Offset Calibration for channel 6** For ADC output set CALIB_EN_6 = 0

Table 400 RESET of ADC2_CAL_CH6_7

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		
TRIM_2	00000000_H	TRIM		

ADC2 Calibration Value Channel 8

ADC2_CAL_CH8_9	Offset	Reset Value
Calibration for Channel 8	**44_H**	**see Table 401**



Field	Bits	Type	Description
RES	31:16	r	**Reserved** Always read as 0
GAIN_CH8	15:8	rwpt	**Gain Calibration for channel 8** For ADC output set CALIB_EN_8 = 0
RES	7:5	r	**Reserved** Always read as 0
OFFS_CH8	4:0	rwpt	**Offset Calibration for channel 8** For ADC output set CALIB_EN_8 = 0

Table 401 RESET of ADC2_CAL_CH8_9

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		
TRIM_2	00000000_H	TRIM		

23.6 IIR-Filter

23.6.1 Functional Description

To cancel high frequency noise out of the measured signal, every channel of the digital signal includes a first order IIR Filter. The structure of the IIR Filter is shown in the picture below.

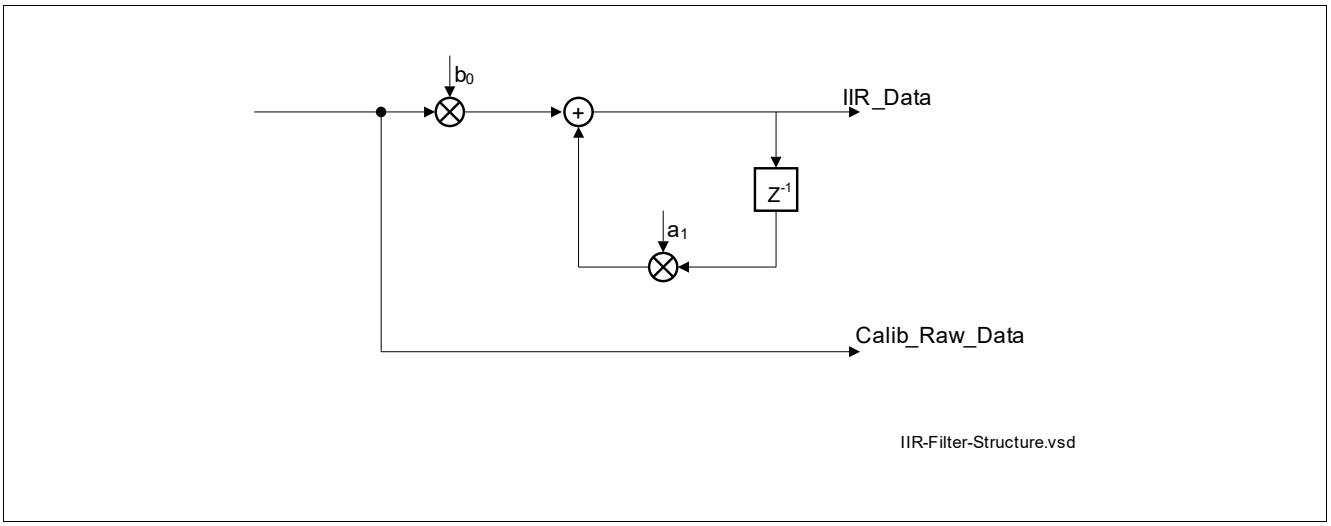


Figure 191 IIR-Filter Implementation Structure

$$H_{IIR}(z) = \frac{b_0}{(1 - a_1 * z^{-1})} \tag{23.5}$$

This filter allows an effective suppression of high-frequency components like noise or crosstalk caused by HF-components in order to avoid the generation of unwanted interrupts. The coefficient b can be expressed as:

$$b_0 = 1 - a_1 \tag{23.6}$$

With the coefficient b implemented in the IIR Filter transfer function, it looks like:

$$H_{IIR}(z) = \frac{1 - a_1}{(1 - a_1 * z^{-1})} \tag{23.7}$$

The IIR Filter transfer function is shown in the plot below.

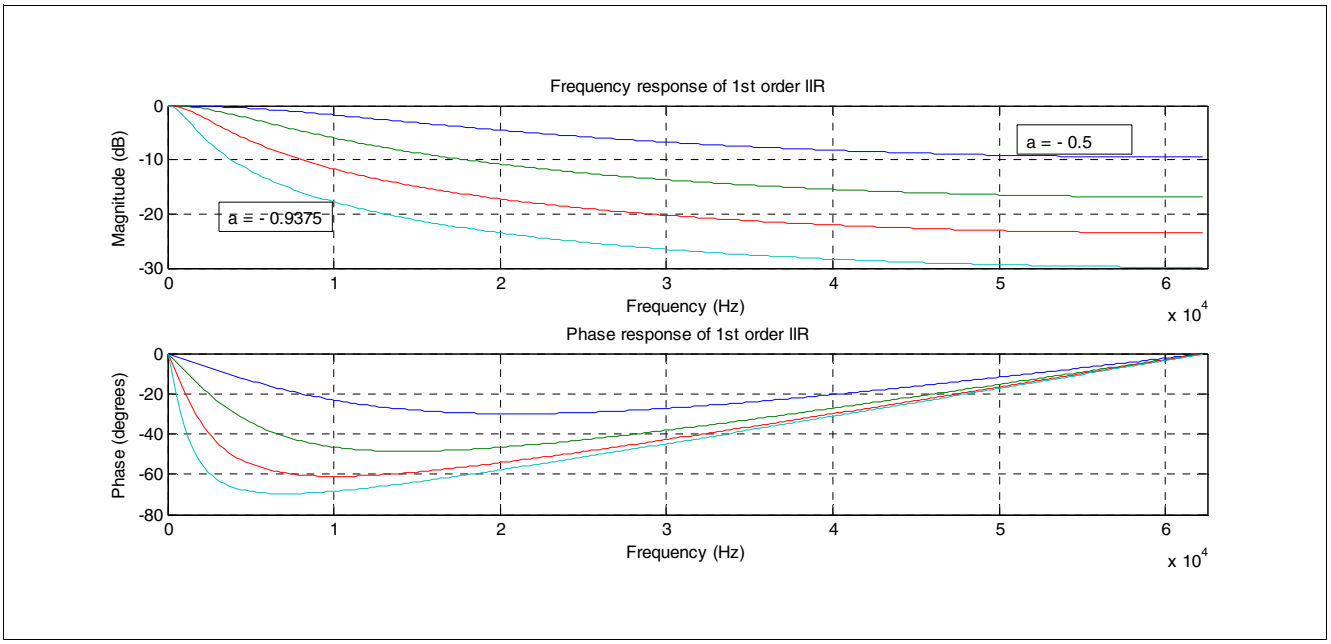


Figure 192 IIR filter transfer function for different filter length fl (1MHz corresponds to 1/2*channel sampling frequency)

23.6.1.1 Step Response

The IIR filter's step response time is shown in the figure below:

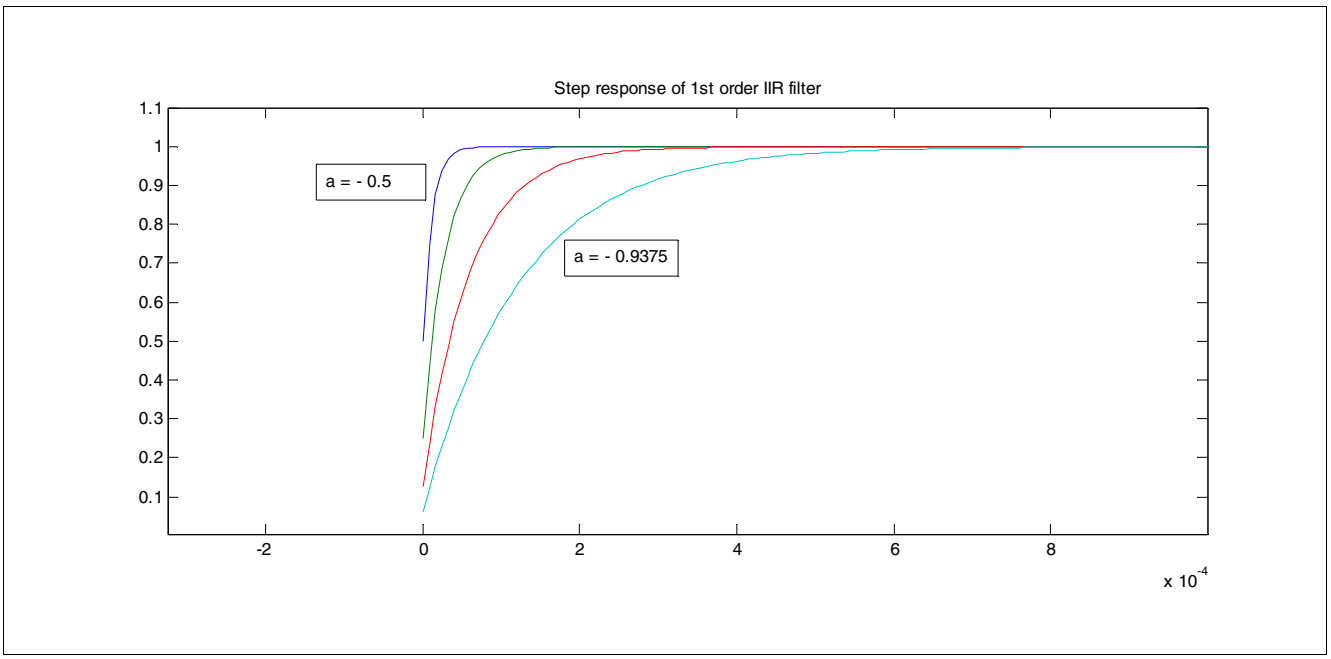


Figure 193 IIR Step Response Time

Table 402 summarizes the main filter characteristics.

Table 402 IIR filter characteristics

Filter coefficient	Group delay at=ω0
a	**τ[samples]**
-2^{-1}	2
-2^{-2}	4
-2^{-3}	8
-2^{-4}	16

23.6.2 IIR Filter Control Registers

The IIR Filter can also be configured by the **sfr** Register shown below. The registers which cannot be written by the user have the attribute **rwpt**.

The **ADC2_FILT_OUT0** to **ADC2_FILT_OUT8** registers are 10 bits wide, but the ADC delivers only a resolution of 8 bits. Bits 1:0 of ADC2_FILT_OUTx contain two bits frational part (2^{-1},2^{-2}) after calibration and filtering, increasing the resolution to 1/4 LSB. **Table 403** shows how the lower two bits are determined.

Table 403 ADC2_FILT_OUT Register Setting

ADC2_CTRL1.calib_en	ADC2_CTRL4.filt_out_sel	ADC2_FILT_OUT0.output[1:0]
0	0	"00"[1]
0	1	2 bit fractional part after filtering
1	0	2 bit fractional part after calibration
1	1	2 bit fractional part after calibration and filtering

1) ADC2_FILT_OUTx 9:2 contains the 8 bit ADC output value if calibration and filtering are disabled.

The result of the calibration unit is 10 bits (see **Figure 190**), the output is feed into the IIR filter (see **Figure 191**). The internal result of the IIR filter is 12 bits, the output is converted to 10 bit and feed into the postprocessing. The user can monitor the calculated values in the **ADC2_FILT_OUT0** to **ADC2_FILT_OUT8** registers and gets access to 10 bit wide result information.

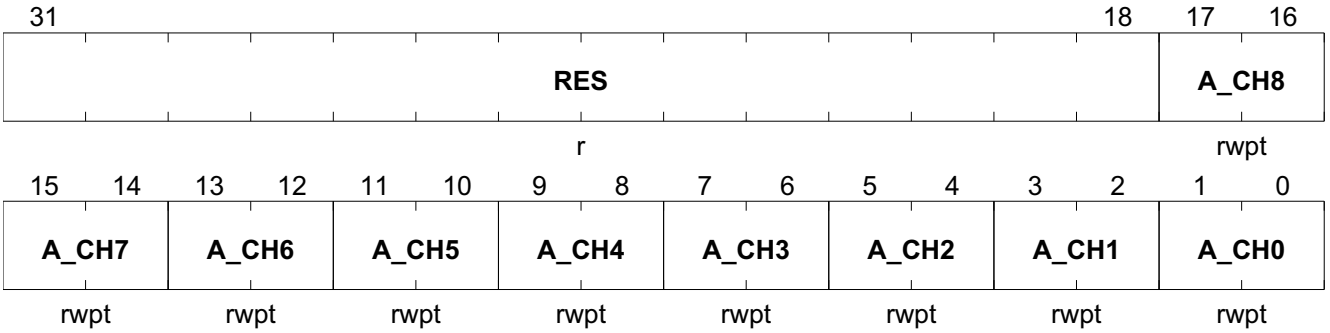
Table 404 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
IIR Filter Control Registers,			
ADC2_FILTCOEFF0_8	Filter Coefficients ADC Channel 0-8	48_H	see **Table 405**
ADC2_FILT_OUT0	ADC or Filter Output Channel 0	50_H	see **Table 406**
ADC2_FILT_OUT1	ADC or Filter Output Channel 1	54_H	see **Table 407**
ADC2_FILT_OUT2	ADC or Filter Output Channel 2	58_H	see **Table 408**
ADC2_FILT_OUT3	ADC or Filter Output Channel 3	$5C_H$	see **Table 409**
ADC2_FILT_OUT4	ADC or Filter Output Channel 4	60_H	see **Table 410**
ADC2_FILT_OUT5	ADC or Filter Output Channel 5	64_H	see **Table 411**
ADC2_FILT_OUT6	ADC or Filter Output Channel 6	68_H	see **Table 412**
ADC2_FILT_OUT7	ADC or Filter Output Channel 7	$6C_H$	see **Table 413**
ADC2_FILT_OUT8	ADC or Filter Output Channel 8	70_H	see **Table 414**

The registers are addressed wordwise.

Filter Coefficients ADC Channel 0-8

ADC2_FILTCOEFF0_8	Offset	Reset Value
Filter Coefficients ADC Channel 0-8	48_H	see Table 405



Field	Bits	Type	Description
RES	31:18	r	**Reserved** Always read as 0
A_CH8	17:16	rwpt	**Filter Coefficient A for ADC channel 8** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
A_CH7	15:14	rwpt	**Filter Coefficient A for ADC channel 7** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
A_CH6	13:12	rwpt	**Filter Coefficient A for ADC channel 6** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
A_CH5	11:10	rwpt	**Filter Coefficient A for ADC channel 5** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
A_CH4	9:8	rwpt	**Filter Coefficient A for ADC channel 4** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample

Field	Bits	Type	Description
A_CH3	7:6	rwpt	**Filter Coefficient A for ADC channel 3** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
A_CH2	5:4	rwpt	**Filter Coefficient A for ADC channel 2** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
A_CH1	3:2	rwpt	**Filter Coefficient A for ADC channel 1** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
A_CH0	1:0	rwpt	**Filter Coefficient A for ADC channel 0** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample

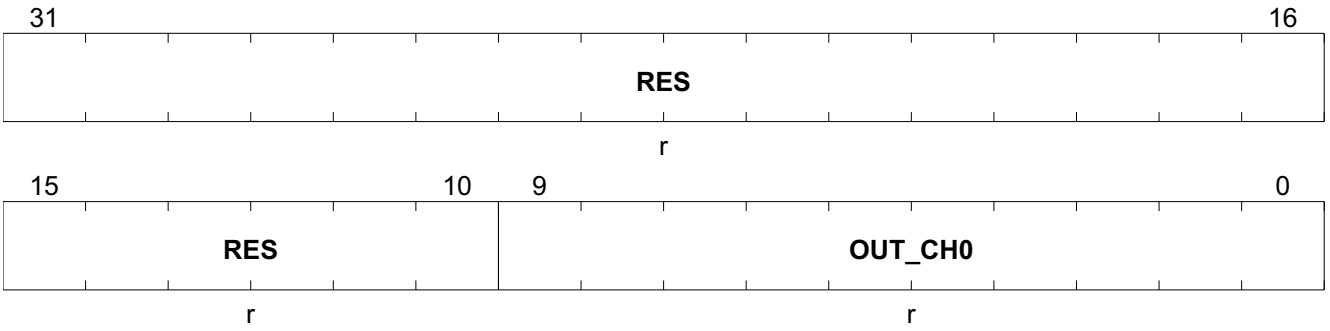
Table 405 RESET of ADC2_FILTCOEFF0_8

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00015555_H	RESET_TYPE_4		
TRIM_2	00015555_H	RESET		

ADC or Filter Output Channel 0

This register reflects the current value of channel 0 of the measurement chain, which is assigned to VBAT_SENSE measurement.

ADC2_FILT_OUT0	Offset	Reset Value
ADC or Filter Output Channel 0	50_H	see Table 406



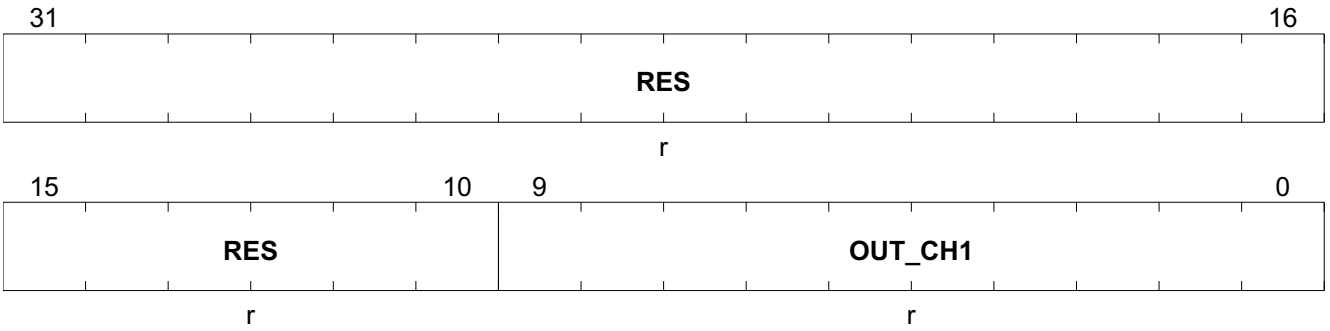
Field	Bits	Type	Description
RES	31:10	r	**Reserved** Always read as 0
OUT_CH0	9:0	r	**ADC or filter output value channel 0** For ADC output set ADC2_CTRL4.FILT_OUT_SEL_8_0<0>=0

Table 406 RESET of ADC2_FILT_OUT0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000 0000 0000 0000 00XX XXXX $XXXX_B$	RESET_TYPE_3		

ADC or Filter Output Channel 1

ADC2_FILT_OUT1	Offset	Reset Value
ADC or Filter Output Channel 1	54_H	see Table 407



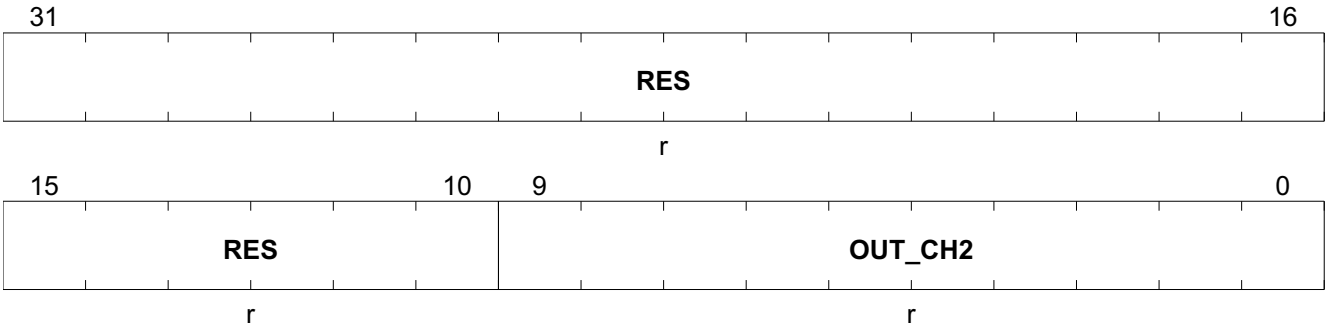
Field	Bits	Type	Description
RES	31:10	r	**Reserved** Always read as 0
OUT_CH1	9:0	r	**ADC or filter output value channel 1** For ADC output set ADC2_CTRL4.FILT_OUT_SEL_8_0<1>=0

Table 407 RESET of ADC2_FILT_OUT1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000 0000 0000 0000 00XX XXXX $XXXX_B$	RESET_TYPE_3		

ADC or Filter Output Channel 2

ADC2_FILT_OUT2	Offset	Reset Value
ADC or Filter Output Channel 2	58_H	see Table 408



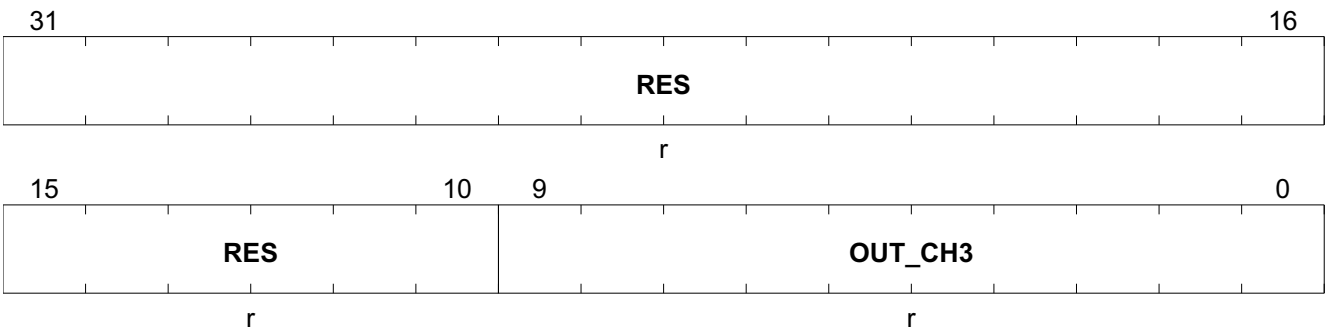
Field	Bits	Type	Description
RES	31:10	r	**Reserved** Always read as 0
OUT_CH2	9:0	r	**ADC or filter output value channel 2** For ADC output set ADC2_CTRL4.FILT_OUT_SEL_8_0<2>=0

Table 408 RESET of ADC2_FILT_OUT2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000 0000 0000 0000 00XX XXXX $XXXX_B$	RESET_TYPE_3		

ADC or Filter Output Channel 3

ADC2_FILT_OUT3	Offset	Reset Value
ADC or Filter Output Channel 3	$5C_H$	see Table 409



Field	Bits	Type	Description
RES	31:10	r	**Reserved** Always read as 0

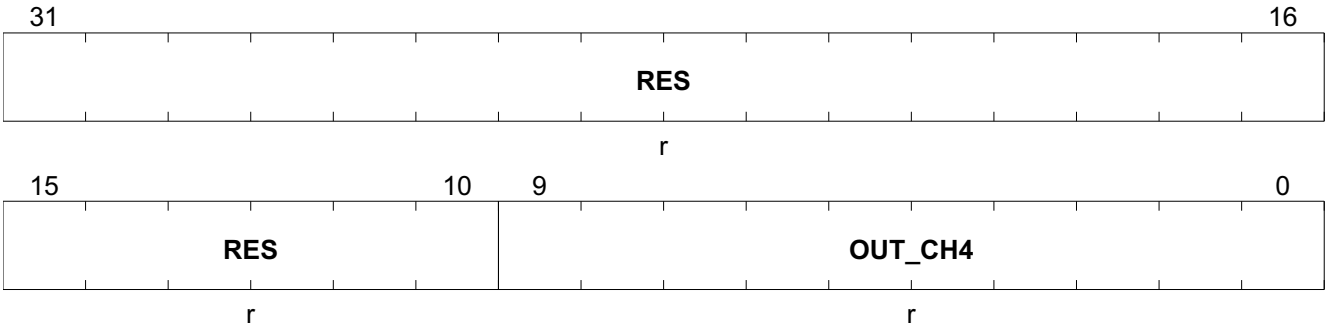
Field	Bits	Type	Description
OUT_CH3	9:0	r	**ADC or filter output value channel 3** For ADC output set ADC2_CTRL4.FILT_OUT_SEL_8_0<3>=0

Table 409 RESET of ADC2_FILT_OUT3

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000 0000 0000 0000 00XX XXXX $XXXX_B$	RESET_TYPE_3		

ADC or Filter Output Channel 4

ADC2_FILT_OUT4	Offset	Reset Value
ADC or Filter Output Channel 4	60_H	see Table 410



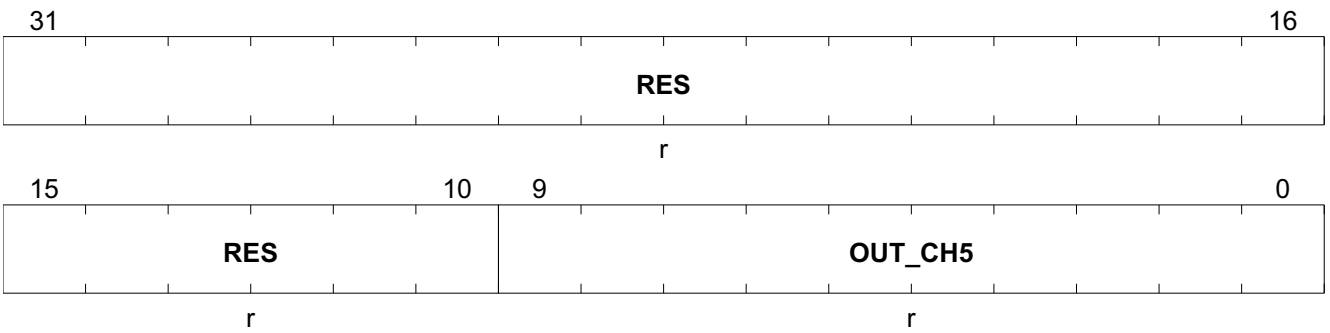
Field	Bits	Type	Description
RES	31:10	r	**Reserved** Always read as 0
OUT_CH4	9:0	r	**ADC or filter output value channel 4** For ADC output set ADC2_CTRL4.FILT_OUT_SEL_8_0<4>=0

Table 410 RESET of ADC2_FILT_OUT4

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000 0000 0000 0000 00XX XXXX $XXXX_B$	RESET_TYPE_3		

ADC or Filter Output Channel 5

ADC2_FILT_OUT5	Offset	Reset Value
ADC or Filter Output Channel 5	64_H	see Table 411



Field	Bits	Type	Description
RES	31:10	r	**Reserved** Always read as 0

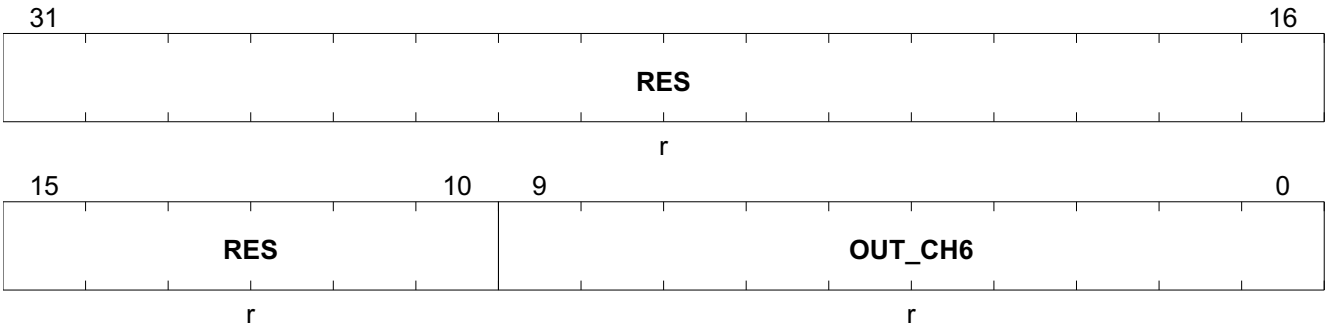
Field	Bits	Type	Description
OUT_CH5	9:0	r	**ADC or filter output value channel 5** For ADC output set ADC2_CTRL4.FILT_OUT_SEL_8_0<5>=0

Table 411 RESET of ADC2_FILT_OUT5

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000 0000 0000 0000 00XX XXXX $XXXX_B$	RESET_TYPE_3		

ADC or Filter Output Channel 6

ADC2_FILT_OUT6	Offset	Reset Value
ADC or Filter Output Channel 6	**68_H**	**see Table 412**



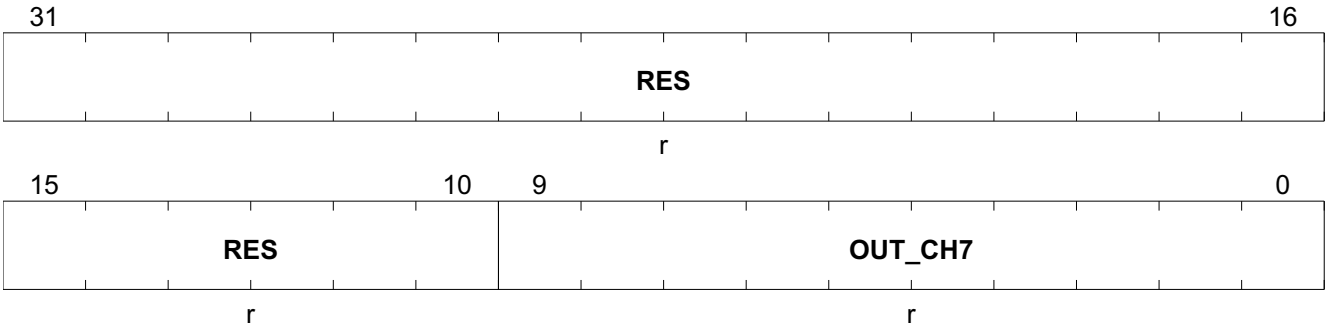
Field	Bits	Type	Description
RES	31:10	r	**Reserved** Always read as 0
OUT_CH6	9:0	r	**ADC or filter output value channel 6** For ADC output set ADC2_CTRL4.FILT_OUT_SEL_8_0<6>=0

Table 412 RESET of ADC2_FILT_OUT6

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000 0000 0000 0000 00XX XXXX $XXXX_B$	RESET_TYPE_3		

ADC or Filter Output Channel 7

ADC2_FILT_OUT7	Offset	Reset Value
ADC or Filter Output Channel 7	**$6C_H$**	**see Table 413**



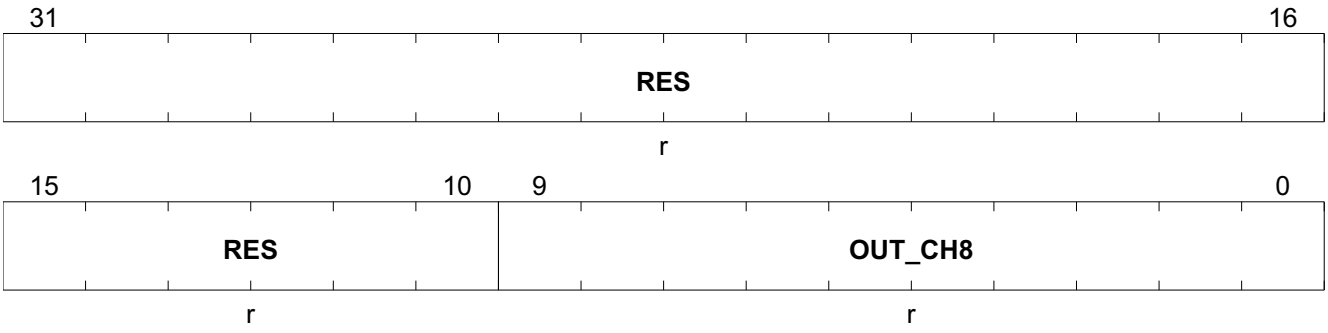
Field	Bits	Type	Description
RES	31:10	r	**Reserved** Always read as 0
OUT_CH7	9:0	r	**ADC or filter output value channel 7** For ADC output set ADC2_CTRL4.FILT_OUT_SEL_8_0<7>=0

Table 413 RESET of ADC2_FILT_OUT7

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000 0000 0000 0000 00XX XXXX $XXXX_B$	RESET_TYPE_3		

ADC or Filter Output Channel 8

ADC2_FILT_OUT8	Offset	Reset Value
ADC or Filter Output Channel 8	70_H	see Table 414



Field	Bits	Type	Description
RES	31:10	r	**Reserved** Always read as 0
OUT_CH8	9:0	r	**ADC or filter output value channel 8** For ADC output set ADC2_CTRL4.FILT_OUT_SEL_8_0<8>=0

Table 414 RESET of ADC2_FILT_OUT8

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000 0000 0000 0000 00XX XXXX $XXXX_B$	RESET_TYPE_3		

23.7 Signal Processing

23.7.1 Functional Description

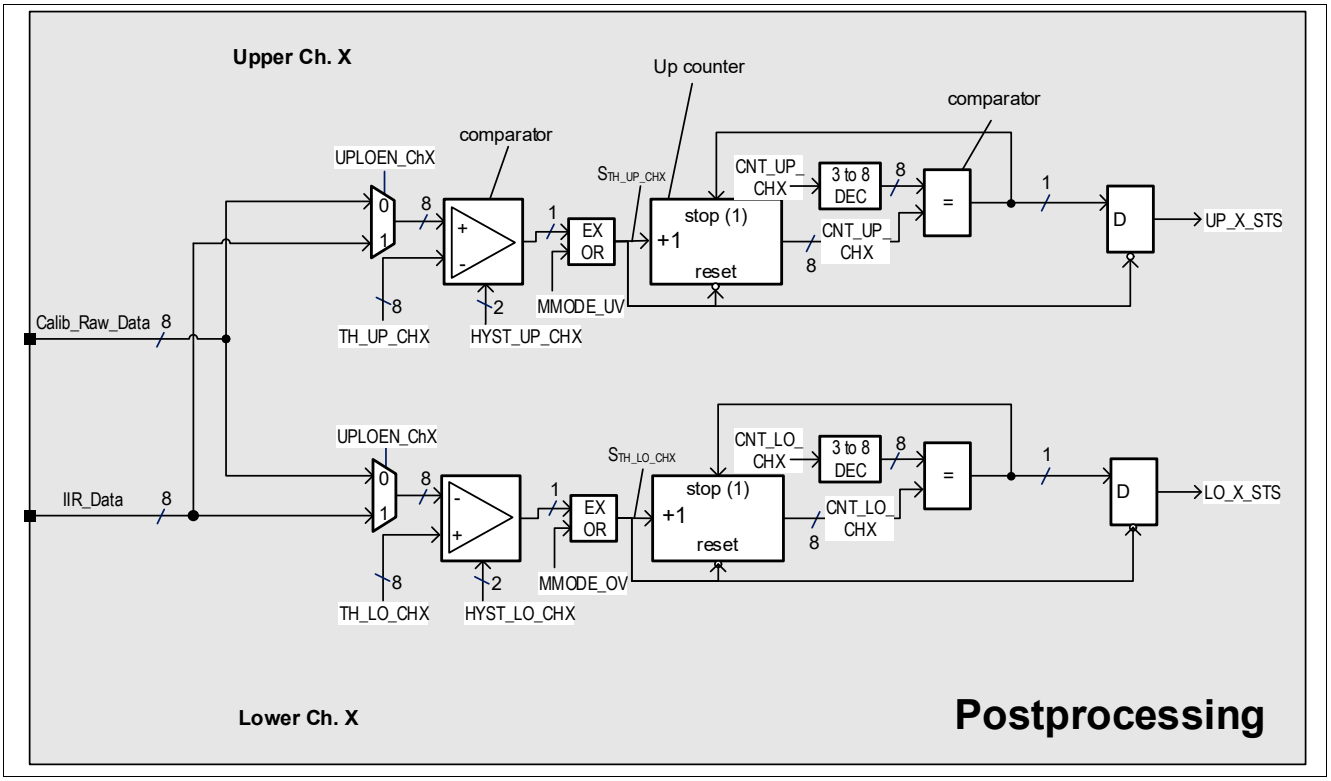


Figure 194 Postprocessing Channel Block Diagram for Voltage and Temperature Measurements

As shown in **Figure 194** an adjustable filter can be applied for the upper and the lower measurement channel, which averages 2, 4, 8 or 16 measurement values continuously. The filtered signal or the demultiplexed ADC output signal ADC_OUTX is compared with an upper threshold TH_UP_CHX and a lower threshold TH_LO_CHX. When the thresholds are exceeded, the comparator outputs get active. For all measurement modes a freely adjustable hysteresis can be defined which is defined with the HYST_UP_CHX and HYST_LO_CHX values.

In addition to the first filter stage, the second filters (counters) integrate the comparator output values S_{TH_UP/LO_CHX} until an individual upper and lower timing threshold $2^{CNT\_UP/LO\_CHX}$ is reached. When reaching the upper timing threshold $2^{CNT\_UP\_CHX}$, the upper counter increment is stalled and the status output CHX_UP_STS is set. For MMODE_OV = 1, the inverted lower comparator output signal $S_{TH\_LO\_CHX}$ is normalized again. When the output signal is above TH_LO_CHX, the lower counter is incremented until the max. threshold $2^{CNT\_LO\_CHX}$ is reached.

In general the IIR filter stage suppresses higher frequency noise efficiently and triggering with the upper and lower threshold TH_UP/LO_CHX are dependent on the measured values. Hence short high-level spikes might pass the thresholds. In opposite to the first stage, the nature of the second filter stage is more a time filter, which is less dependent on the measurement values but on event durations of S_{TH_LO/UP_CHX} as generated by the first comparator stage. Therefore the second stage has a lower noise suppression performance for higher frequencies and also adds a delay for the trigger time proportional to $2^{CNT\_LO/HI\_CHX}$.

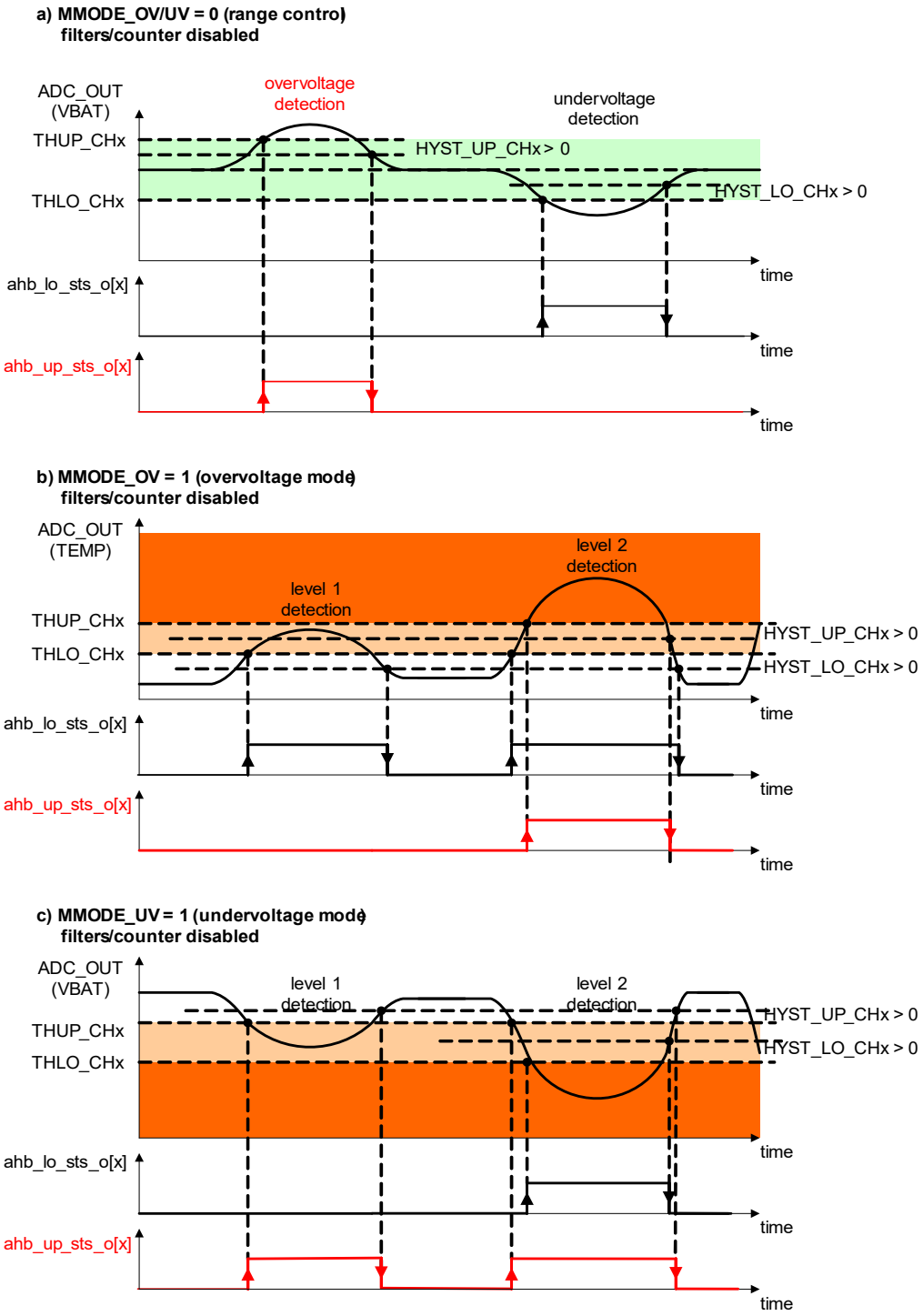


Figure 195 Measurement Examples of a Measurement Channel with Disabled Filters

Figure 195 shows three examples, an over- and undervoltage detection (e.g. VBAT_SENSE monitoring), a 2-step overvoltage and a 2-step undervoltage detection. The modes MMODE_OV/UV = 1 can be used as prewarning for the application software (e.g. close to overtemperature or supply undervoltage).

To ensure right functionality, registers of some channels are protected. This is shown in **Table 415**.

Table 415 Register protection for some channels

Measurement channel n	protection on LOWER registers	protection on UPPER registers
0 (VS)	YES	no
4 (VDDC)	YES	no
5 (VBG)	YES	YES
6 (VDDP)	YES	no
7 and 8 (temp. sensors)	no	YES

23.7.2 Postprocessing Control Registers

The Postprocessing Unit is fully controllable by the below listed sfr Registers.

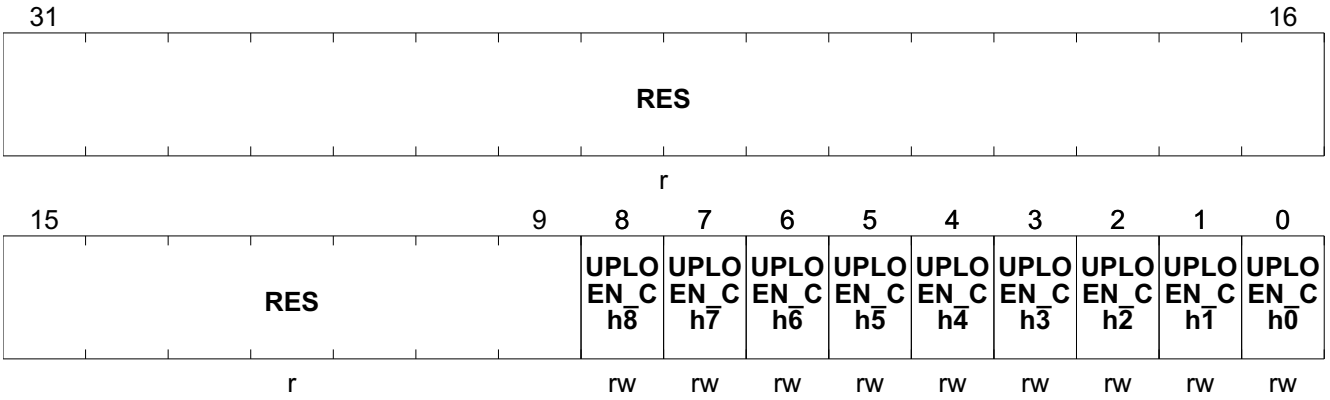
Table 416 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Postprocessing Control Registers,			
ADC2_FILT_UPLO_CTRL	Upper and Lower Threshold Filter Enable	78_H	see **Table 417**
ADC2_TH0_3_LOWER	Lower Comparator Trigger Level Channel 0 -3	80_H	see **Table 425**
ADC2_TH4_7_LOWER	Lower Comparator Trigger Level Channel 4 to 7	84_H	see **Table 426**
ADC2_TH8_11_LOWER	Lower Comparator Trigger Level Channel 8	88_H	see **Table 427**
ADC2_TH0_3_UPPER	Upper Comparator Trigger Level Channel 0-3	$8C_H$	see **Table 419**
ADC2_TH4_7_UPPER	Upper Comparator Trigger Level Channel 4 -7	90_H	see **Table 420**
ADC2_TH8_11_UPPER	Upper Comparator Trigger Level Channel 8	94_H	see **Table 421**
ADC2_CNT0_3_LOWER	Lower Counter Trigger Level Channel 0 - 3	98_H	see **Table 428**
ADC2_CNT4_7_LOWER	Lower Counter Trigger Level Channel 4 to 7	$9C_H$	see **Table 429**
ADC2_CNT8_11_LOWER	Lower Counter Trigger Level Channel 8	$A0_H$	see **Table 430**
ADC2_CNT0_3_UPPER	Upper Counter Trigger Level Channel 0 - 3	$A4_H$	see **Table 422**
ADC2_CNT4_7_UPPER	Upper Counter Trigger Level Channel 4 to 7	$A8_H$	see **Table 423**
ADC2_CNT8_11_UPPER	Upper Counter Trigger Level Channel 8	AC_H	see **Table 424**
ADC2_MMODE0_8	Measurement Mode of Ch 0-8	$B0_H$	see **Table 418**

The registers are addressed wordwise.

Upper and Lower Threshold Filter Enable

ADC2_FILT_UPLO_CTRL	Offset	Reset Value
Upper and Lower Threshold Filter Enable	78_H	see Table 417



Field	Bits	Type	Description
RES	31:9	r	**Reserved** Always read as 0
UPLOEN_Ch8	8	rw	**Upper and lower threshold IIR filter enable ch 8** 0_B **Disable**, 1_B **Enable**,
UPLOEN_Ch7	7	rw	**Upper and lower threshold IIR filter enable ch 7** 0_B **Disable**, 1_B **Enable**,
UPLOEN_Ch6	6	rw	**Upper and lower threshold IIR filter enable ch 6** 0_B **Disable**, 1_B **Enable**,
UPLOEN_Ch5	5	rw	**Upper and lower threshold IIR filter enable ch 5** 0_B **Disable**, 1_B **Enable**,
UPLOEN_Ch4	4	rw	**Upper and lower threshold IIR filter enable ch 4** 0_B **Disable**, 1_B **Enable**,
UPLOEN_Ch3	3	rw	**Upper and lower threshold IIR filter enable ch 3** 0_B **Disable**, 1_B **Enable**,
UPLOEN_Ch2	2	rw	**Upper and lower threshold IIR filter enable ch 2** 0_B **Disable**, 1_B **Enable**,
UPLOEN_Ch1	1	rw	**Upper and lower threshold IIR filter enable ch 1** 0_B **Disable**, 1_B **Enable**,

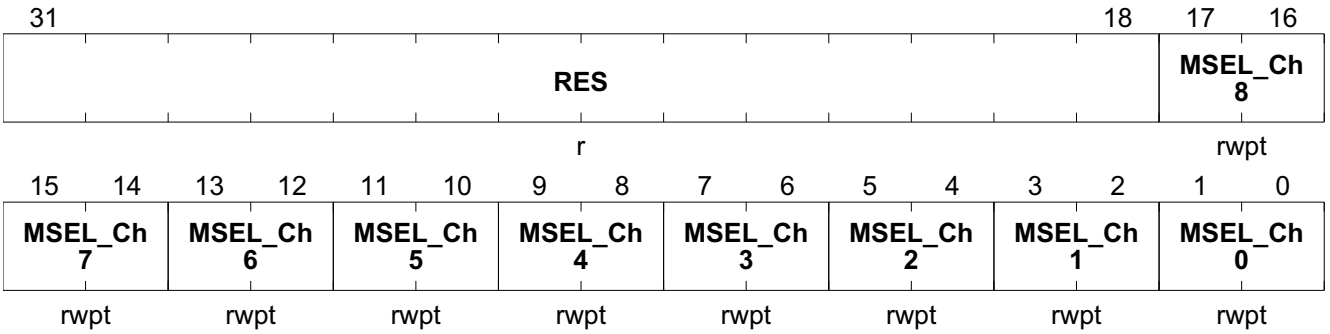
Field	Bits	Type	Description
UPLOEN_Ch0	0	rw	**Upper and lower threshold IIR filter enable ch 0** 0_B **Disable**, 1_B **Enable**,

Table 417 RESET of ADC2_FILT_UPLO_CTRL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$000001FF_H$	RESET_TYPE_4		
TRIM_2	$000001FF_H$	RESET		

Overvoltage Measurement Mode of Channel 0-8

ADC2_MMODE0_8	Offset	Reset Value
Measurement Mode of Ch 0-8	$B0_H$	see Table 418



Field	Bits	Type	Description
RES	31:18	r	**Reserved** Always read as 0
MSEL_Ch8	17:16	rwpt	**Measurement mode ch 8** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved
MSEL_Ch7	15:14	rwpt	**Measurement mode ch 7** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved

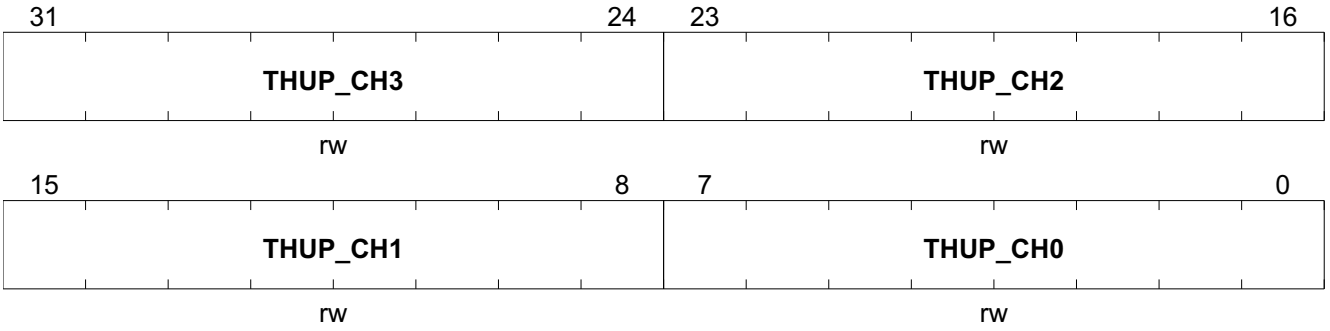
Field	Bits	Type	Description
MSEL_Ch6	13:12	rwpt	**Measurement mode ch 6** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved
MSEL_Ch5	11:10	rwpt	**Measurement mode ch 5** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved
MSEL_Ch4	9:8	rwpt	**Measurement mode ch 4** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved
MSEL_Ch3	7:6	rwpt	**Measurement mode ch 3** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved
MSEL_Ch2	5:4	rwpt	**Measurement mode ch 2** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved
MSEL_Ch1	3:2	rwpt	**Measurement mode ch 1** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved
MSEL_Ch0	1:0	rwpt	**Measurement mode ch 0** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved

Table 418 RESET of ADC2_MMODE0_8

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00028000_H	RESET_TYPE_4		
TRIM_2	00028000_H	RESET		

Upper Comparator Trigger Level Channel 0 - 3

ADC2_TH0_3_UPPER	Offset	Reset Value
Upper Comparator Trigger Level Channel 0-3	$8C_H$	see Table 419



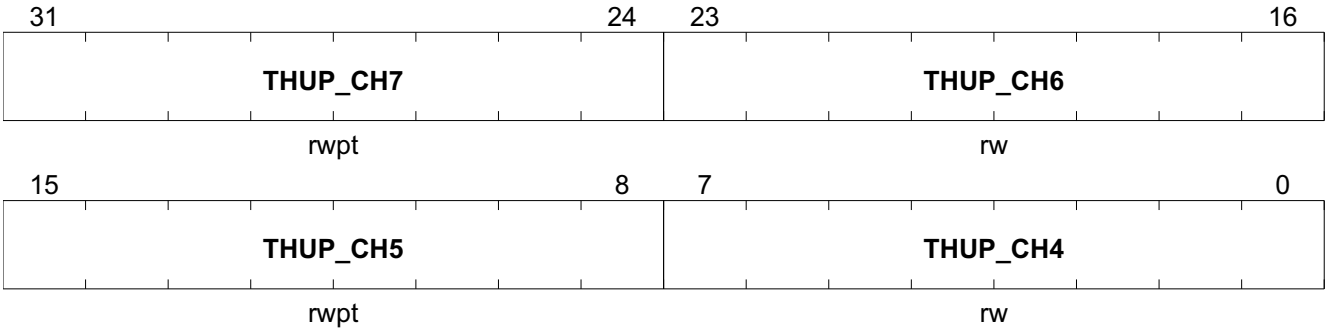
Field	Bits	Type	Description
THUP_CH3	31:24	rw	**Channel 3 upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value
THUP_CH2	23:16	rw	**Channel 2 upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value
THUP_CH1	15:8	rw	**Channel 1 upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value
THUP_CH0	7:0	rw	**Channel 0 upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value

Table 419 RESET of ADC2_TH0_3_UPPER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$DBE2EAEA_H$	RESET_TYPE_4		
TRIM_2	$DBE5F2EA_H$	COMMON		

Upper Comparator Trigger Level Channel 4 - 7

ADC2_TH4_7_UPPER	Offset	Reset Value
Upper Comparator Trigger Level Channel 4 - 7	90_H	see Table 420



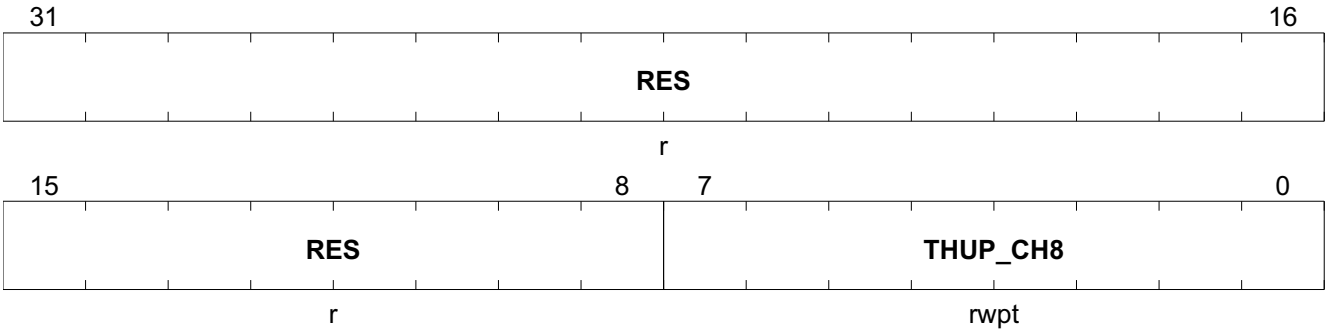
Field	Bits	Type	Description
THUP_CH7	31:24	rwpt	**Channel 7 upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value
THUP_CH6	23:16	rw	**Channel 6 upper trigger level** 00_H **0**, min threshold value FF_H **255**, max. threshold value
THUP_CH5	15:8	rwpt	**Channel 5 upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value
THUP_CH4	7:0	rw	**Channel 4 upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value

Table 420 RESET of ADC2_TH4_7_UPPER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$E0FDAFD7_H$	RESET_TYPE_4		
TRIM_2	$E0FDB2D7_H$	COMMON		

Upper Comparator Trigger Level Channel 8

ADC2_TH8_11_UPPER	Offset	Reset Value
Upper Comparator Trigger Level Channel 8	94_H	see Table 421



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Always read as 0
THUP_CH8	7:0	rwpt	**Channel 8 upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value

Table 421 RESET of ADC2_TH8_11_UPPER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$000000E0_H$	RESET_TYPE_4		
TRIM_2	$000000E0_H$	RESET		

Upper Counter Trigger Level Channel 0 - 3

ADC2_CNT0_3_UPPER	Offset	Reset Value
Upper Counter Trigger Level Channel 0 - 3	$A4_H$	see Table 422

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RES			HYST_UP_CH3		RES	CNT_UP_CH3		RES			HYST_UP_CH2		RES	CNT_UP_CH2	
r			rw		r	rw		r			rw		r	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RES			HYST_UP_CH1		RES	CNT_UP_CH1		RES			HYST_UP_CH0		RES	CNT_UP_CH0	
r			rw		r	rw		r			rw		r	rw	

Field	Bits	Type	Description
RES	31:29	r	**Reserved** Always read as 0
HYST_UP_CH3	28:27	rw	**Channel 3 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	26	r	**Reserved** Always read as 0
CNT_UP_CH3	25:24	rw	**Upper timer trigger threshold channel 3** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	23:21	r	**Reserved** Always read as 0
HYST_UP_CH2	20:19	rw	**Channel 2 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	18	r	**Reserved** Always read as 0
CNT_UP_CH2	17:16	rw	**Upper timer trigger threshold channel 2** 0_H **1** , 1 measurement 1_H **2** , 2 measurements 2_H **4** , 4 measurements 3_H **7** , 7 measurements
RES	15:13	r	**Reserved** Always read as 0

Field	Bits	Type	Description
HYST_UP_CH1	12:11	rw	**Channel 1 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	10	r	**Reserved** Always read as 0
CNT_UP_CH1	9:8	rw	**Upper timer trigger threshold channel 1** 0_H **1** , 1 measurement 1_H **2** , 2 measurements 2_H **4** , 4 measurements 3_H **7** , 7 measurements
RES	7:5	r	**Reserved** Always read as 0
HYST_UP_CH0	4:3	rw	**Channel 0 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	2	r	**Reserved** Always read as 0
CNT_UP_CH0	1:0	rw	**Upper timer trigger threshold channel 0** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements

Table 422 RESET of ADC2_CNT0_3_UPPER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	09090909_H	RESET_TYPE_4		
TRIM_2	09090909_H	RESET		

Upper Counter Trigger Level Channel 4 to 7

ADC2_CNT4_7_UPPER	Offset	Reset Value
Upper Counter Trigger Level Channel 4 to 7	$A8_H$	see Table 423

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RES			HYST_UP_CH7		RES	CNT_UP_CH7		RES			HYST_UP_CH6		RES	CNT_UP_CH6	
r			rwpt		r	rwpt		r			rw		r	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RES			HYST_UP_CH5		RES	CNT_UP_CH5		RES			HYST_UP_CH4		RES	CNT_UP_CH4	
r			rwpt		r	rwpt		r			rw		r	rw	

Field	Bits	Type	Description
RES	31:29	r	**Reserved** Always read as 0
HYST_UP_CH7	28:27	rwpt	**Channel 7 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	26	r	**Reserved** Always read as 0
CNT_UP_CH7	25:24	rwpt	**Upper timer trigger threshold channel 7** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	23:21	r	**Reserved** Always read as 0
HYST_UP_CH6	20:19	rw	**Channel 6 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	18	r	**Reserved** Always read as 0
CNT_UP_CH6	17:16	rw	**Upper timer trigger threshold channel 6** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	15:13	r	**Reserved** Always read as 0

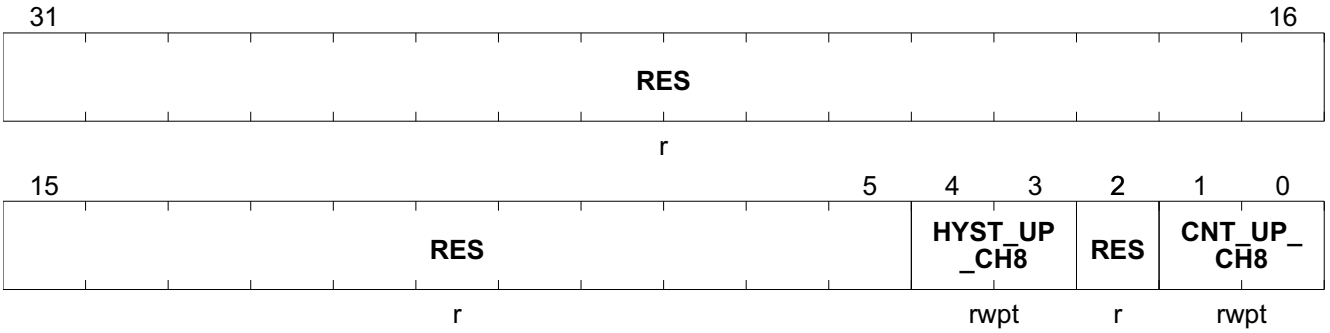
Field	Bits	Type	Description
HYST_UP_CH5	12:11	rwpt	**Channel 5 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	10	r	**Reserved** Always read as 0
CNT_UP_CH5	9:8	rwpt	**Upper timer trigger threshold channel 5** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	7:5	r	**Reserved** Always read as 0
HYST_UP_CH4	4:3	rw	**Channel 4 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	2	r	**Reserved** Always read as 0
CNT_UP_CH4	1:0	rw	**Upper timer trigger threshold channel 4** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements

Table 423 RESET of ADC2_CNT4_7_UPPER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0B090909_H$	RESET_TYPE_4		
TRIM_2	$0B090909_H$	RESET		

Upper Counter Trigger Level Channel 8

ADC2_CNT8_11_UPPER	Offset	Reset Value
Upper Counter Trigger Level Channel 8	AC_H	see Table 424



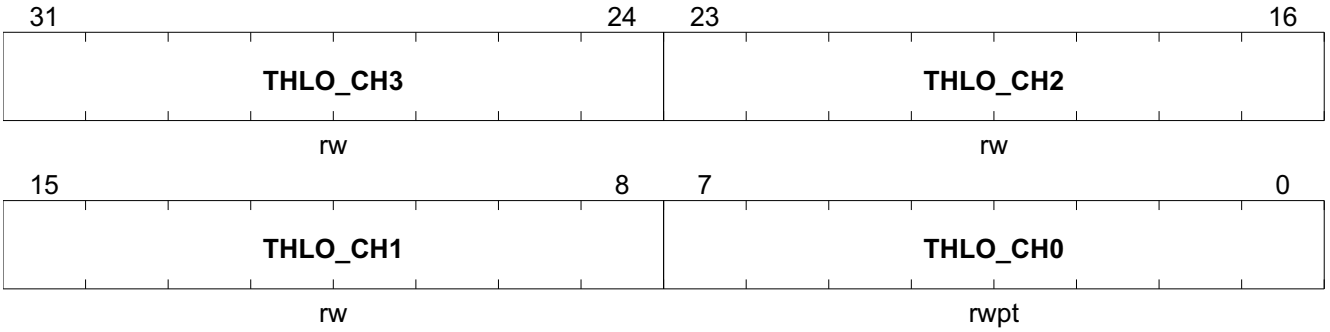
Field	Bits	Type	Description
RES	31:5	r	**Reserved** Always read as 0
HYST_UP_CH8	4:3	rwpt	**Channel 8 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	2	r	**Reserved** Always read as 0
CNT_UP_CH8	1:0	rwpt	**Upper timer trigger threshold channel 8** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements

Table 424 RESET of ADC2_CNT8_11_UPPER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000000B_H$	RESET_TYPE_4		
TRIM_2	$0000000B_H$	RESET		

Lower Comparator Trigger Level Channel 0 - 3

ADC2_TH0_3_LOWER	Offset	Reset Value
Lower Comparator Trigger Level Channel 0 - 3	80_H	see Table 425



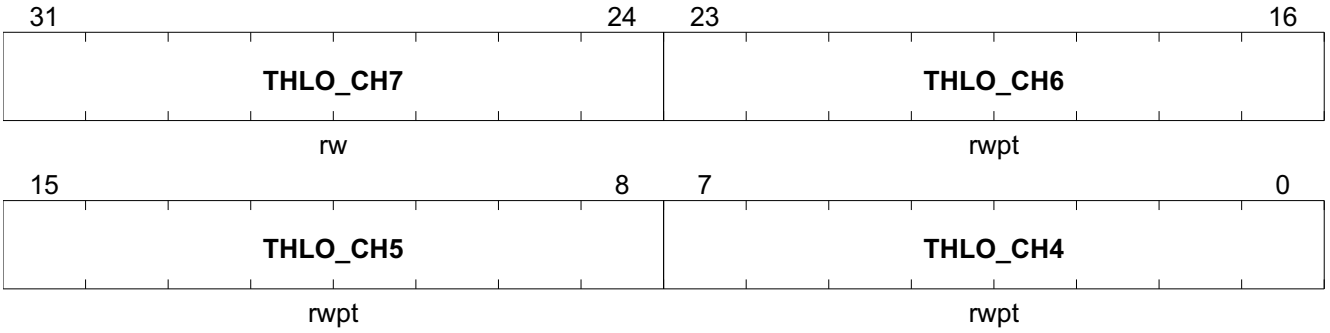
Field	Bits	Type	Description
THLO_CH3	31:24	rw	**Channel 3 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value
THLO_CH2	23:16	rw	**Channel 2 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value
THLO_CH1	15:8	rw	**Channel 1 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value
THLO_CH0	7:0	rwpt	**Channel 0 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value

Table 425 RESET of ADC2_TH0_3_LOWER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$C32E2B2C_H$	RESET_TYPE_4		
TRIM_2	$C32E2B2C_H$	RESET		

Lower Comparator Trigger Level Channel 4 to 7

ADC2_TH4_7_LOWER	Offset	Reset Value
Lower Comparator Trigger Level Channel 4 to 7	84_H	see Table 426



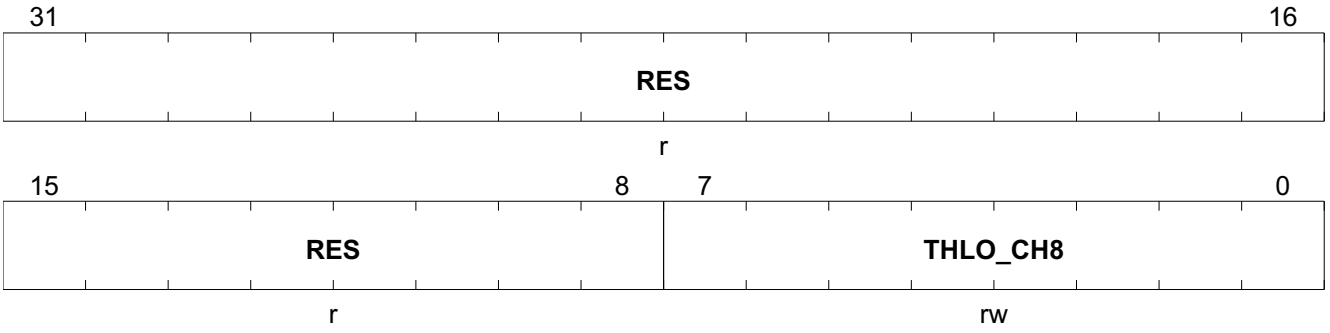
Field	Bits	Type	Description
THLO_CH7	31:24	rw	**Channel 7 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value
THLO_CH6	23:16	rwpt	**Channel 6 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value
THLO_CH5	15:8	rwpt	**Channel 5 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value
THLO_CH4	7:0	rwpt	**Channel 4 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value

Table 426 RESET of ADC2_TH4_7_LOWER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$C8D38FC7_H$	RESET_TYPE_4		
TRIM_2	$C8D38CC7_H$	COMMON		

Lower Comparator Trigger Level Channel 8

ADC2_TH8_11_LOWER	Offset	Reset Value
Lower Comparator Trigger Level Channel 8	88_H	see Table 427



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Always read as 0
THLO_CH8	7:0	rw	**Channel 8 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value

Table 427 RESET of ADC2_TH8_11_LOWER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$000000C8_H$	RESET_TYPE_4		
TRIM_2	$000000C8_H$	RESET		

Lower Counter Trigger Level Channel 0 - 3

ADC2_CNT0_3_LOWER	Offset	Reset Value
Lower Counter Trigger Level Channel 0 - 3	98_H	see Table 428

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RES			HYST_LO_CH3		RES	CNT_LO_CH3		RES			HYST_LO_CH2		RES	CNT_LO_CH2	
r			rw		r	rw		r			rw		r	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RES			HYST_LO_CH1		RES	CNT_LO_CH1		RES			HYST_LO_CH0		RES	CNT_LO_CH0	
r			rw		r	rw		r			rwpt		r	rwpt	

Field	Bits	Type	Description
RES	31:29	r	**Reserved** Always read as 0
HYST_LO_CH3	28:27	rw	**Channel 3 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	26	r	**Reserved** Always read as 0
CNT_LO_CH3	25:24	rw	**Lower timer trigger threshold channel 3** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	23:21	r	**Reserved** Always read as 0
HYST_LO_CH2	20:19	rw	**Channel 2 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	18	r	**Reserved** Always read as 0
CNT_LO_CH2	17:16	rw	**Lower timer trigger threshold channel 2** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	15:13	r	**Reserved** Always read as 0

Field	Bits	Type	Description
HYST_LO_CH1	12:11	rw	**Channel 1 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	10	r	**Reserved** Always read as 0
CNT_LO_CH1	9:8	rw	**Lower timer trigger threshold channel 1** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	7:5	r	**Reserved** Always read as 0
HYST_LO_CH0	4:3	rwpt	**Channel 0 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	2	r	**Reserved** Always read as 0
CNT_LO_CH0	1:0	rwpt	**Lower timer trigger threshold channel 0** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements

Table 428 RESET of ADC2_CNT0_3_LOWER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	09090909_H	RESET_TYPE_4		
TRIM_2	09090909_H	RESET		

Lower Counter Trigger Level Channel 4 to 7

ADC2_CNT4_7_LOWER	Offset	Reset Value
Lower Counter Trigger Level Channel 4 to 7	$9C_H$	see Table 429

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RES			HYST_LO_CH7		RES	CNT_LO_CH7		RES			HYST_LO_CH6		RES	CNT_LO_CH6	
r			rw		r	rw		r			rwpt		r	rwpt	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RES			HYST_LO_CH5		RES	CNT_LO_CH5		RES			HYST_LO_CH4		RES	CNT_LO_CH4	
r			rwpt		r	rwpt		r			rwpt		r	rwpt	

Field	Bits	Type	Description
RES	31:29	r	**Reserved** Always read as 0
HYST_LO_CH7	28:27	rw	**Channel 7 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	26	r	**Reserved** Always read as 0
CNT_LO_CH7	25:24	rw	**Lower timer trigger threshold channel 7** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	23:21	r	**Reserved** Always read as 0
HYST_LO_CH6	20:19	rwpt	**Channel 6 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	18	r	**Reserved** Always read as 0
CNT_LO_CH6	17:16	rwpt	**Lower timer trigger threshold channel 6** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	15:13	r	**Reserved** Always read as 0

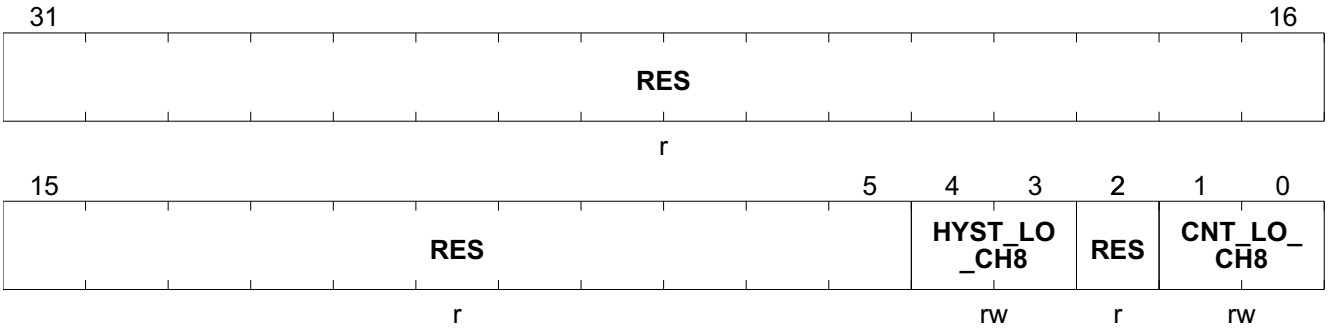
Field	Bits	Type	Description
HYST_LO_CH5	12:11	rwpt	**Channel 5 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	10	r	**Reserved** Always read as 0
CNT_LO_CH5	9:8	rwpt	**Lower timer trigger threshold channel 5** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	7:5	r	**Reserved** Always read as 0
HYST_LO_CH4	4:3	rwpt	**Channel 4 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	2	r	**Reserved** Always read as 0
CNT_LO_CH4	1:0	rwpt	**Lower timer trigger threshold channel 4** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements

Table 429 RESET of ADC2_CNT4_7_LOWER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0B090909_H$	RESET_TYPE_4		
TRIM_2	$0B090909_H$	RESET		

Lower Counter Trigger Level Channel 8

ADC2_CNT8_11_LOWER	Offset	Reset Value
Lower Counter Trigger Level Channel 8	$A0_H$	see Table 430



Field	Bits	Type	Description
RES	31:5	r	**Reserved** Always read as 0
HYST_LO_CH8	4:3	rw	**Channel 8 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	2	r	**Reserved** Always read as 0
CNT_LO_CH8	1:0	rw	**Lower timer trigger threshold channel 8** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements

Table 430 RESET of ADC2_CNT8_11_LOWER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000000B_H$	RESET_TYPE_4		
TRIM_2	$0000000B_H$	RESET		

23.8 Start-up Behavior after Reset

After the end of a reset phase the measurement sources and the post-processing units need some time for settling. In order to avoid undesired triggering of interrupts until the measurement signal acquisition is in a steady state, the status signals are forced to zero during the start-up phase.

The end of the start-up phase is indicated by the ready signal MI_RDY.

Measurement Core start-up procedure: the startup time of the complete signal chain are 2200 EoC cycles. The IIR-filter coefficient is set to C=2^-1 (fastest response time).

During the startup phase, the DPP will use SQ=1_1111_1111, regardless of the sequence registers configuration.

23.9 Postprocessing Default Values

The following table shows the assigned measurements of the particular channels and the reset default values which read from FW during power-up.

Table 431 Channel allocation and postprocessing default settings (effective after reset)

Channel / MMODE [1]	Analog	Digital [2]	Hyste-resis [3]	IIR - Filter [4]	Counters [5]	Functional Description
Ch. 0 / 0_H **VS**	5.3 V	$2C_H$	1_H (4)	1_H (4)	1_H (2)	Battery supply voltage input, lower
	28.3 V	EA_H	1_H (4)		1_H (2)	Battery supply voltage input, upper
Ch. 1/ 0_H **VSD**	5.1 V	$2B_H$	1_H (4)	1_H (4)	1_H (2)	VSD voltage, lower
	29.3 V	$F2_H$	1_H (4)		1_H (2)	VSD voltage, upper
Ch. 2/ 0_H **VCP**	9.2 V	$2E_H$	1_H (4)	1_H (4)	1_H (2)	VCP voltage, lower
	46.2 V	$E5_H$	1_H (4)		1_H (2)	VCP voltage, upper
Ch. 3/ 0_H **VDDEXT**	4.7 V	$C3_H$	1_H (4)	1_H (4)	1_H (2)	VDDEXT, lower
	5.3 V	DB_H	1_H (4)		1_H (2)	VDDEXT, upper
Ch. 4/ 0_H **VDDP**	4.8 V	$C7_H$	1_H (4)	1_H (4)	1_H (2)	+5V, Port supply voltage, lower
	5.2 V	$D7_H$	1_H (4)		1_H (2)	+5V, Port supply voltage, upper
Ch. 5/ 0_H **VBG**	0.88 V	$8C_H$	1_H (4)	1_H (4)	1_H (2)	VBG, lower
	1.12 V	$B2_H$	1_H (4)		1_H (2)	VBG, upper
Ch. 6/ 0_H **VDDC**	1.33 V	$D3_H$	1_H (4)	1_H (4)	1_H (2)	Core supply voltage, lower
	1.59 V	FD_H	1_H (4)		1_H (2)	Core supply voltage, upper
Ch. 7/ 2_H **TEMP_CP**	0.94 V	$C8_H$	1_H (4)	1_H (4)	3_H (8)	temperature sensor: lower hysteresis threshold value corresponding to approx. 135°C
	1.06 V	$E0_H$	1_H (4)		3_H (8)	overtemperature threshold corresponding to 185°C.
Ch. 8/ 2_H **TEMP_SYS**	0.94 V	$C8_H$	1_H (4)	1_H (4)	3_H (8)	temperature sensor: lower hysteresis threshold value corresponding to approx. 135°C
	1.06 V	$E0_H$	1_H (4)		3_H (8)	overtemperature threshold corresponding to 185°C.

1) MMODE of each channel is defined by sfr reset values: 0_H range control, 1_H under-voltage mode, 2_H overvoltage mode.

2) register: THLO_CHx / THUP_CHx

3) register: HYST_LO_CHx / HYST_UP_CHx; selectable decimal values [0, 4, 8, 16]
4) register: ADC2_FILTCOEFF0_8; selectable decimal values [2, 4, 8, 16]
5) register: CNT_LO_CHx / CNT_UP_CHx; selectable decimal values [$2^0 2^1 ... 2^3$

24 Analog Digital Converter ADC10B (ADC1)

24.1 Features

The basic function of this block is the digital postprocessing of several analog digitized measurement signals by means of filtering, level comparison and interrupt generation. The measurement postprocessing block is built of the same number of identical channel units attached to the outputs of the 10-bit ADC. It processes allchannels, where the channel sequence and prioritization is programmable within a wide range.

Functional Features

- 10 Bit SAR ADC with conversion time of 17 clock cycles
- programmable clock divider for sequencer and ADC
- 12 individually programmable channels:
  - 6 HV Channels: VS, VBAT_SENSE, MON1...MON4
  - 5 LV Channels: P2.0, P2.1, P2.2, P2.3, P2.7
  - 1 Current-Sense Amplifier Channel
- all channels are fully calibrated; calibration can be enabled/disabled by user
- individually programmable channel prioritization scheme for digital postprocessing (dpp)
- two independent filter stages with programmable low-pass and time filter characteristics for each channel
- two channel configurations:
  - programmable upper- and lower trigger thresholds comprising a fully programmable hysteresis
  - two individually programmable trigger thresholds with limit hysteresis settings
- individually programmable upper threshold and lower threshold interrupts and status for all channel thresholds
- one additional differential channel (MON1-MON2) with postprocessing and interrupt generation
- ADC voltage reference completely integrated

Note: In case the MONx should be evaluated by the ADC1, it is recommended to add 6.8nF capacitors close to the MONx pin of the device, in order to build an external RC filter to limit the bandwidth of the input signal.

24.2 Introduction

The basic function of this unit, is the digital signal processing of several analog digitized measurement signals by means of filtering, level comparison and interrupt generation. The Measurement Core module processes all channels in a quasi parallel process.

24.2.1 Block Diagram

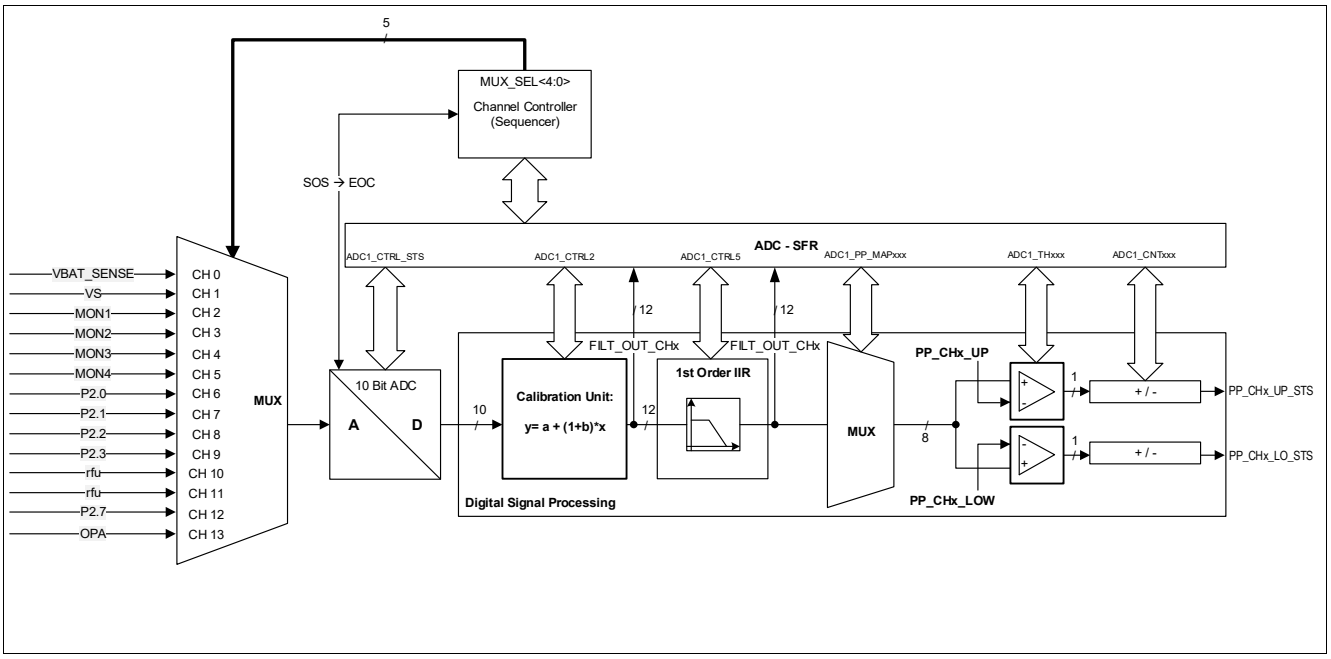


Figure 196 Module Block Diagram

As shown in the figure above, the ADC postprocessing consists of a channel controller (Sequencer), a demultiplexer and the signal processing block, which filters and compares the sampled ADC values for each channel individually. The channel control block controls the multiplexer sequencing on the analog side before the ADC and on the digital domain after the ADC. As described in the following section, the channel sequence can be controlled in a flexible way, which allows a certain degree of channel prioritization.

This capability can be used e.g. to set a higher priority to supply voltage channels compared to the other channel measurements. The Measurement Core Module offers additionally two different post-processing measurement modes for over-/undervoltage detection and for two-level threshold detection.

24.2.2 ADC1 Modes Overview

Usually the external register settings should only be changed during the start-up phase.

The channel controller (Sequencer) runs in one of the following mode:

“Normal Sequencer Mode”, channels are selected out of 14 Sequence register which contain individual enables for each of the 14 channels.

“Exceptional Interrupt Measurement”, a high priority channel is inserted into the current sequence. The current actual measurement is not destroyed

“Exceptional Sequence Measurement”, upon a hardware event event, a complete sequence is inserted after the current measurement is finished. The current sequence is interrupted by the exception sequence.

“Software Mode”, Sequencer and Exceptional Interrupt and Sequence Measurement is disabled, each measurement is triggered by software.

The IIR filter can be bypassed via **ADC1_FILT_UPLO_CTRL**

The threshold counter can be bypassed (counting only 1 measurement) via CNT_LO_PPx

24.3 ADC1 - Core (10-Bit ADC)

24.3.1 Functional Description

The different sequencer modes are controlled by SFR Register:

- “Normal Sequencer Mode” described in the Chapter **Channel Controller**.
- “Exceptional Interrupt Measurement” (EIM), upon a hardware event , the channel programmed in **ADC1_CHx_EIM** is inserted after the current measurement is finished. Afterwards the current sequence will be continued with the next measurement from the current sequence. Up to max. 128 consecutive measurements are possible.
- “Exceptional Sequence Measurement“ (ESM), upon a hardware event , the sequence programmed in **ADC1_CHx_ESM** is inserted after the current measurement is finished. After the sequence (up to 12 measurements) exception is finished the next measurement from the interrupted sequence is selected. After the Exceptional Sequence Measurement is finished an interrupt is issued.
- “Software Mode”, in Software Mode the control of the Channel Controller (Sequencer) is disabled, instead the conversions are fully controlled by software. During Software Mode, entered via SQx = 0, EIM and ESM hardware events are set to pending. During Software Mode, entered via debug or SW Mode SFR Register, ESM and EIM events are ignored.

Software Mode:

- Software mode can be entered in different ways
  - by writing one of the sequence registers SQn (e.g. to SQ_1[13:0]) to zero or setting the register **ADC1_CTRL3**.SW_MODE
  - by writing the Exceptional Sequence Measurement (**ADC1_CHx_ESM**) to zero and enable the Exceptional Sequence Measurement.
  - using Debug Suspend Mode
- In Software mode, the channel selection by the Sequencer is disabled. The entry of software mode is acknowledged in the **ADC1_SQ_FB**. After the software mode is entered, the conversion are controlled via **ADC1_CTRL_STS**.
- The Software Mode is left
  - when the maximum time is reached (maximum time specified in **ADC1_MAX_TIME**) or
  - when the sequence which started the software mode is reprogrammed with at least one channel set in its registers SQn (e.g. to SQ_1) not equal to zero
  - when the Exceptional Sequence Measurement (**ADC1_CHx_ESM**) is reprogrammed with at least one channel set
  - leaving Debug Suspend Mode

Software Mode:

The default mode of the DPP1 is the sequencer mode. To change from this default mode to Software mode the corresponding flag ADC1.ADC1_CTRL2.SW_MODE. In Software Mode measurements are triggered by writing the **ADC1_CTRL_STS**.SOS bit. This bit is active as long as the conversion is in progress. The user polls the **ADC1_CTRL_STS**.EOC bit. Once this bit is ‘1’ the conversion is finished and the EOC bit is cleared on read (rhc). After the EOC bit is cleared a new conversion can be started **ADC1_CTRL_STS**.SOS.

Debug Suspend Mode:

During Debug Suspend Mode the Sequencer is stopped once the current measurement is finished (after the next EOC event) and Software Mode is entered. As long as the Debug Suspend Mode is active no measurements are performed by the Sequencer. Once the Debug Suspend Mode is left, the Sequencer continues immediately with the next pending measurement.
Measurements can be still triggered in Debug Suspend Mode/Software Mode. The maximum time of Software Mode is disabled in Suspend Mode.EIM and ESM events are ignored during Debug Suspend Mode.

The ADC timing is controlled by SFR Register

- Sample time adjustment described in the register **ADC1_CTRL3**.

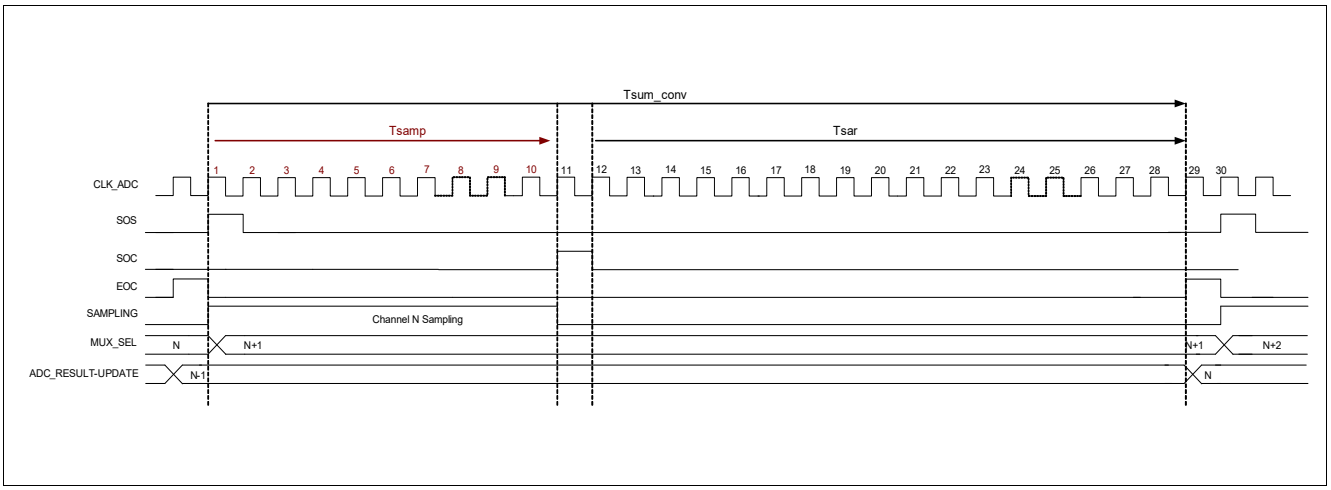


Figure 197 10 Bit ADC Timing - Single conversion

24.3.2 ADC1 Control and Status Registers

The ADC1 is fully controllable by the below listed special function registers in software mode.

Table 432 shows the module base addresses.

Table 432 Register Address Space

Module	Base Address	End Address	Note
ADC1	40004000_H	$40007FFF_H$	ADC1 - ADC-SAR10B

Table 433 Register Overview

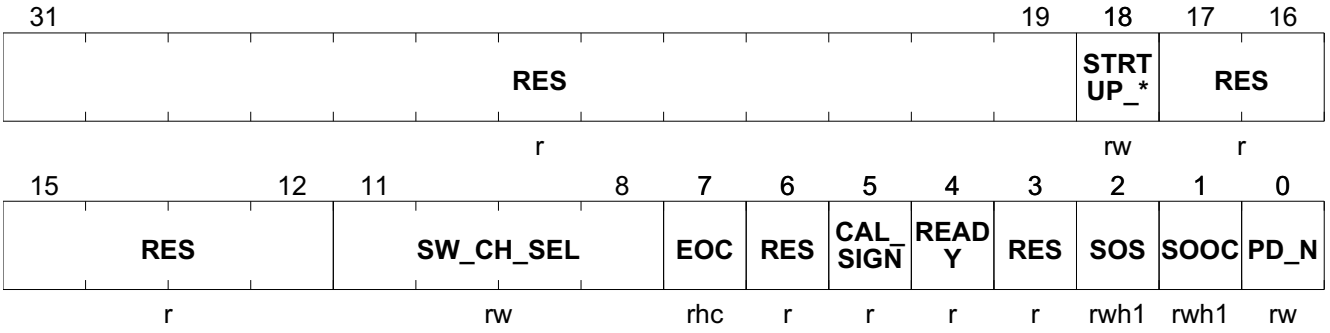
Register Short Name	Register Long Name	Offset Address	Reset Value
ADC1 Control and Status Registers,			
ADC1_CTRL_STS	ADC1 Control and Status Register	00_H	see **Table 434**
ADC1_OFFSETCALIB	ADC1 Offset Calibration Register	$3C_H$	see **Table 435**
ADC1_STATUS	ADC1 Status Register	BC_H	see **Table 436**

The registers are addressed wordwise.

Analog Digital Converter ADC10B (ADC1)

ADC1 Control and Status Register

ADC1_CTRL_STS	Offset	Reset Value
ADC1 Control and Status Register	00_H	see Table 434



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
STRTUP_DIS	18	rw	**DPP1 Startup Disable** 0_B **Startup Enable**, DPP1 Startup enabled 1_B **Startup Disable**, DPP1 Startup disable
RES	17:12	r	**Reserved** Always read as 0
SW_CH_SEL	11:8	rw	**Channel for software mode** 0000_B **CH0_EN**, Channel 0 enable 0001_B **CH1_EN**, Channel 1 enable 0010_B **CH2_EN**, Channel 2 enable 0011_B **CH3_EN**, Channel 3 enable 0100_B **CH4_EN**, Channel 4 enable 0101_B **CH5_EN**, Channel 5 enable 0110_B **CH6_EN**, Channel 6 enable 0111_B **CH7_EN**, Channel 7 enable 1000_B **CH8_EN**, Channel 8 enable 1001_B **CH9_EN**, Channel 9 enable 1010_B **rfu**, reserved for future use 1011_B **rfu**, reserved for future use 1100_B **CH12_EN**, Channel 12 enable 1101_B **CH13_EN**, Channel 13 enable 1110_B **rfu**, reserved for future use 1111_B **rfu**, reserved for future use
EOC	7	rhc	**ADC1 End of Conversion (software mode)** *Note: this flag is not only cleared by a read operation but also automatically by setting ADC1_SOS* 0_B **Pending**, conversion still running 1_B **Finished**, conversion has finished

Field	Bits	Type	Description
RES	6	r	**Reserved** Always read as 0
CAL_SIGN	5	r	**Output of Comparator to Steer Gain / Offset calibration**
READY	4	r	**HVADC Ready bit** 0_B **Not ready**, Module in power down or in init phase 1_B **Ready**, set automatically 5 ADC clock cycles after module is enabled
RES	3	r	**Reserved** Always read as 0
SOS	2	rwh1	**ADC1 Start of Sampling/Conversion (software mode)** *Note: Bit is set by software to start sampling and conversion and it is cleared by hardware once the conversion is finished. ADC1_SOS can be only written if the DPP is in software mode.* 0_B **Disable**, no conversion is started 1_B **Enable**, conversion is started
SOOC	1	rwh1	**ADC1 Start of Offset Calibration (software mode)** *Note: Bit is set by software to start calibration and it is cleared by hardware once the calibration is finished ADC1_SOOC can be only written if the DPP is in software mode.* 0_B **Disable**, no offset calibration is started 1_B **Enable**, offset calibration is started
PD_N	0	rw	**ADC1 Power Down Signal** 0_B **POWER DOWN**, ADC1 is powered down 1_B **ACTIVE**, ADC1 is switched on

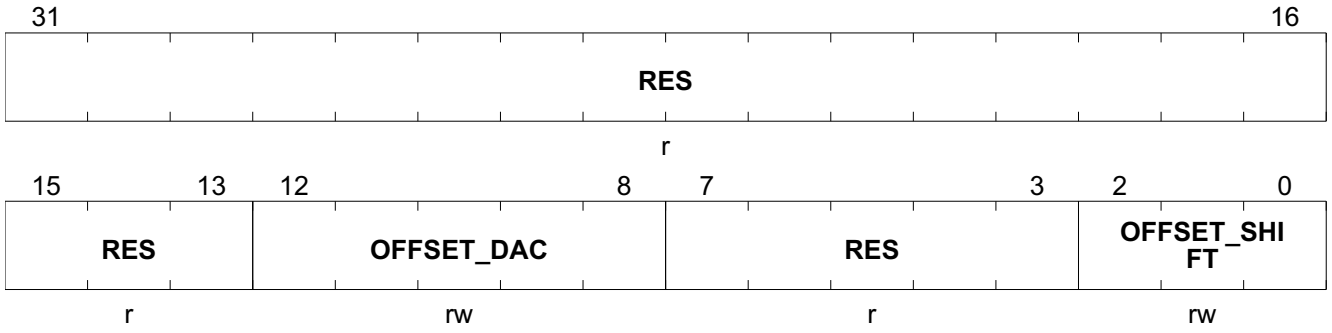
Table 434 RESET of ADC1_CTRL_STS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Analog Digital Converter ADC10B (ADC1)

ADC1 Offset Calibration Register

ADC1_OFFSETCALIB	Offset	Reset Value
ADC1 Offset Calibration Register	$3C_H$	see Table 435



Field	Bits	Type	Description
RES	31:13	r	**Reserved** Always read as 0
OFFSET_DAC	12:8	rw	**Set the Value of the Offset Calibration DAC**
RES	7:3	r	**Reserved** Always read as 0
OFFSET_SHIFT	2:0	rw	**Set the Value of the Offset Shift DAC**

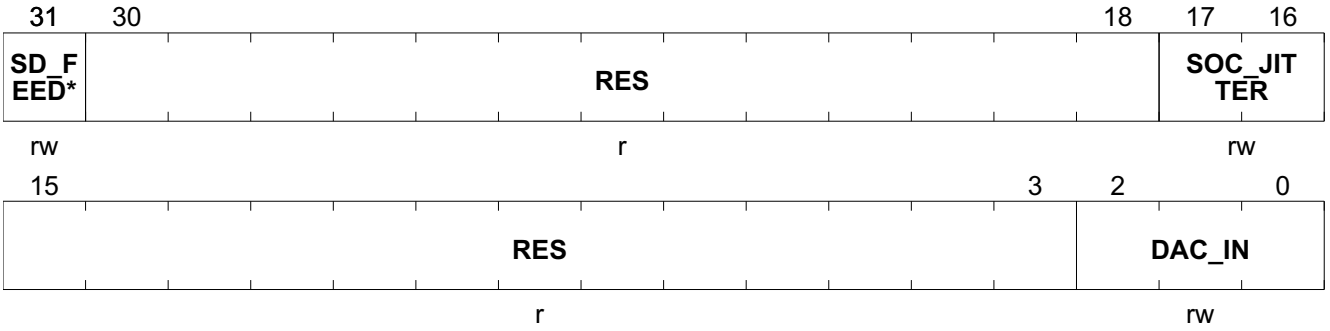
Table 435 RESET of ADC1_OFFSETCALIB

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	TRIM		

Analog Digital Converter ADC10B (ADC1)

ADC1 Status Register

ADC1_STATUS	Offset	Reset Value
ADC1 Status Register	BC_H	see Table 436



Field	Bits	Type	Description
SD_FEEDB_ON	31	rw	**Sigma Delta Feedback Loop** 0_B **Disable**, 1_B **Enable**,
RES	30:18	r	**Reserved** Always read as 0
SOC_JITTER	17:16	rw	**Programs Soc Clock Jitter** 00_B **0n**, 01_B **3.5n**, 10_B **5.5n**, 11_B **8n**,
RES	15:3	r	**Reserved** Always read as 0
DAC_IN	2:0	rw	**Programs the 2-bit DAC for functional test** Number of added LSBs.

Table 436 RESET of ADC1_STATUS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	0000 0000_H	RESET_TYPE_4		

24.4 ADC - Trigger Unit

The DPP Unit provides also a trigger block. This trigger block provides the following functionality:

- “Exceptional Interrupt Measurement” (EIM), upon hardware event , the channel programmed in **ADC1_CHx_EIM** is inserted after the current measurement is finished. Afterwards the current sequence will be continued with the next measurement from the current sequence.
- “Exceptional Sequence Measurement“ (ESM), upon hardware event , the sequence programmed in **ADC1_CHx_ESM** is inserted after the current measurement is finished. After the sequence (up to 14 measurements) exception is finished the next measurement from the interrupted sequence is selected. After the Exceptional Sequence Measurement is finished an interrupt is issued.
- “Software Mode”, in Software Mode the control of the Channel Controller (Sequencer) is disabled, instead the conversions are fully controlled by software. During Software Mode, entered via SQx = 0, EIM and ESM hardware events are set to pending. During Software Mode, entered via debug or SW Mode SFR Register, ESM and EIM events are ignored.

24.5 Channel Controller

24.5.1 Functional Description

The task of each channel controller is a prioritization of the individual measurement channels. The sequencing scheme is illustrated in the example of following table and can be programmed individually for measurement unit.

Table 437 Measurement channel sequence definition example

Measurement channel n	MSB CH13	CH12	CH11	CH10	CH9	CH8	CH7	CH6	CH5	CH4	CH3	CH2	CH1	LSB CH0
Registers SQ_{0_1}[13:0]	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Registers SQ_{0_1}[29:16]	0	1	0	0	0	0	0	0	1	1	1	1	0	0
Registers SQ_{2_3}[13:0]	0	1	1	0	0	0	0	0	0	0	0	0	0	0
Registers SQ_{2_3}[29:16]	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Registers SQ_{4_5}[13:0]	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Registers SQ_{4_5}[29:16]	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Registers SQ_{6_7}[13:0]	0	0	1	1	1	1	1	1	1	1	1	1	1	1
Registers SQ_{6_7}[29:16]	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Registers SQ_{8_9}[13:0]	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Registers SQ_{8_9}[29:16]	0	0	1	1	1	1	1	1	1	1	1	1	1	1
Registers SQ_{10_11}[13:0]	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Registers SQ_{10_11}[29:16]	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Registers SQ_{12_13}[13:0]	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Registers SQ_{12_13}[29:16]	1	1	0	0	0	0	0	0	0	0	0	0	0	0

The sequence registers SQ_n define the time sequence of the measurement channels by the following rules:

- The sequence registers define the measurement sequence and are evaluated from register 0 to 13 and for each register from MSB to LSB, which defines a max. overall measurement periodicity of 196 sampling and conversion cycles.
- If the individual bit in the sequence register is set to ’1’, the corresponding channel is measured.
- If the individual bit in the sequence register is not set, this measurement phase is skipped.

In the upper example, the resulting channel sequence is defined as:

CH13, CH12, CH11, CH10, CH9, CH8, CH7, CH6, CH5,....., CH1, CH0, CH12, CH5, CH4, CH3,...

In TLE985x Channels 0 - 13 can be fully programmed. The channels 0-13 are measured depending on the amount of ’1’ bits, written in the sequence registers. The following equations can be used to calculate the periodicity of the required channel measurement.

The overall measurement periodicity of all measurements in A/D conversion cycles is defined as:

$$\overline{N_{meas}} = \sum_{m=0}^{13} \left(\sum_{n=0}^{13} SQ_m[n] \right) \tag{24.1}$$

which results in 196 A/D conversion cycles. The average measurement periodicity of channel n in A/D conversion cycles is defined as

$$\overline{N_{meas,n}} = \frac{\left(\sum_{m=0}^{13} SQ_m[n] \right)}{\overline{T_{meas}}} \tag{24.2}$$

The timing of the analog MUX and the digital DEMUX is controlled by the channel controller accordingly. The analog MUX with sample and hold stage needs one clock cycle for channel switching and the ADC consumes, as default setting, 12 clock cycles for the sampling of the input voltage. The conversion time for a single channel measurement value is 17 clock cycles.

The minimum measurement periodicity, which can be achieved, by enabling only channel 1 in the sequence registers, depends on the ADC_CLK frequency and is given by:

$$\overline{T_{meas\_CH1\_min}} = \frac{29}{f_{adc\_clk}} \tag{24.3}$$

This following calculations include already the sampling time of ADC. If all programmable channels are enabled, the maximum periodicity is calculated:

$$\overline{T_{meas\_CH1\_min}} = \frac{377}{f_{adc\_clk}} \tag{24.4}$$

For a ADC_CLK frequency of 24 MHz, the channel 1 is measured with min. 1.2 µs. The maximum update time of channel 1 with 24 MHz clock frequency is 16 µs. As mentioned before, this is calculated with the assumption, that all channels are enabled and channel1 is enabled in every sequence register. As a prerequisite for this calculation we take **ADC1_CTRL3** = 4 (sample period = 12 adc clock cycles).

Table 438 ADC1 channel mapping

Sequencer Channel Number	Analog Input
CH0	VBAT_SENSE
CH1	VS
CH2	MON1
CH3	MON2
CH4	MON3

Table 438 ADC1 channel mapping

CH5	MON4
CH6	P2.0
CH7	P2.1
CH8	P2.2
CH9	P2.3
CH10	rfu
CH11	rfu
CH12	P2.7
CH13	OPA

24.5.2 Channel Controller Control Registers

The Channel Controller can fully be configured by the **SFR** Register listed in **Table 439**.

Table 439 Register Overview

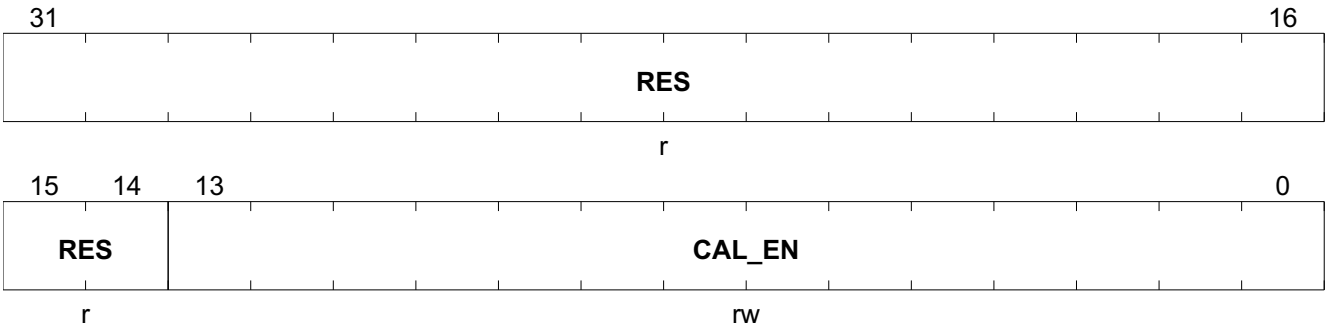
Register Short Name	Register Long Name	Offset Address	Reset Value
Channel Controller Control Registers,			
ADC1_SQ_FB	Sequencer Feedback Register	04_H	see **Table 450**
ADC1_CHx_EIM	Channel Setting Bits for Exceptional Interrupt Measurement	08_H	see **Table 451**
ADC1_CHx_ESM	Channel Setting Bits for Exceptional Sequence Measurement	$0C_H$	see **Table 452**
ADC1_MAX_TIME	Maximum Time for Software Mode	10_H	see **Table 453**
ADC1_CTRL2	Measurement Unit 1 Control Register 2	14_H	see **Table 440**
ADC1_CTRL3	Measurement Unit 1 Control Register 3	18_H	see **Table 441**
ADC1_CTRL5	Measurement Unit 1 Control Register 5	$1C_H$	see **Table 442**
ADC1_SQ0_1	Measurement Unit 1 Channel Enable Bits for Cycle 0-1	20_H	see **Table 443**
ADC1_SQ2_3	Measurement Unit 1 Channel Enable Bits for Cycle 2-3	24_H	see **Table 444**
ADC1_SQ4_5	Measurement Unit 1 Channel Enable Bits for Cycle 4-5	28_H	see **Table 445**
ADC1_SQ6_7	Measurement Unit 1 Channel Enable Bits for Cycle 6-7	$2C_H$	see **Table 446**
ADC1_SQ8_9	Measurement Unit 1 Channel Enable Bits for Cycle 8-9	30_H	see **Table 447**
ADC1_SQ10_11	Measurement Unit 1 Channel Enable Bits for Cycle 10-11	34_H	see **Table 448**
ADC1_SQ12_13	Measurement Unit 1 Channel Enable Bits for Cycle 12-13	130_H	see **Table 449**

The registers are addressed wordwise.

Measurement Unit 1 Control Register 2

This register is dedicated for controlling the calibration unit of the measurement core module. The respective channel calibration can be enabled or disabled by the bits listed below.

ADC1_CTRL2	Offset	Reset Value
Measurement Unit 1 Control Register 2	14_H	see Table 440



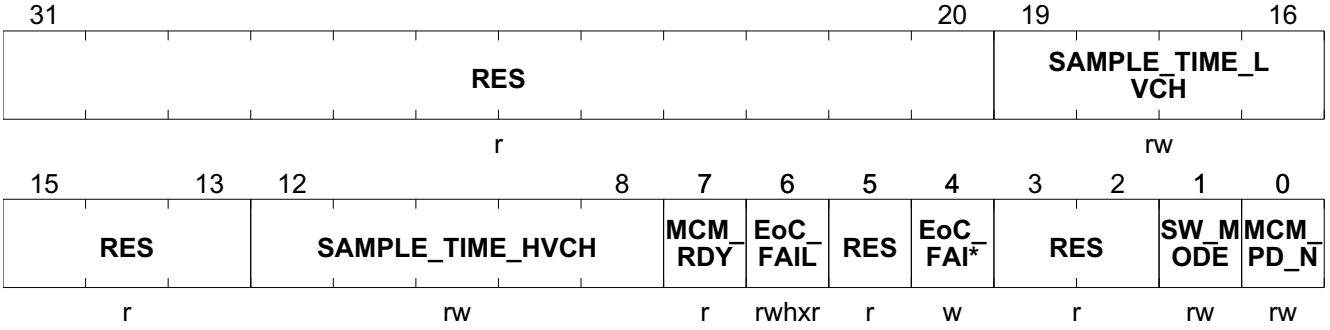
Field	Bits	Type	Description
RES	31:14	r	**Reserved** Always read as 0
CAL_EN	13:0	rw	**Calibration Enable for Channels 0 to 13** The following values can be ored: 0001_H **CH0_EN**, Channel 0 calibration enable 0002_H **CH1_EN**, Channel 1 calibration enable 0004_H **CH2_EN**, Channel 2 calibration enable 0008_H **CH3_EN**, Channel 3 calibration enable 0010_H **CH4_EN**, Channel 4 calibration enable 0020_H **CH5_EN**, Channel 5 calibration enable 0040_H **CH6_EN**, Channel 6 calibration enable 0080_H **CH7_EN**, Channel 7 calibration enable 0100_H **CH8_EN**, Channel 8 calibration enable 0200_H **CH9_EN**, Channel 9 calibration enable 0400_H **CH10_EN**, Channel 10 calibration enable 0800_H **CH11_EN**, Channel 11 calibration enable 1000_H **CH12_EN**, Channel 12 calibration enable 2000_H **CH13_EN**, Channel 13 calibration enable

Table 440 RESET of ADC1_CTRL2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_2	$00003FFF_H$	COMMON		

Measurement Unit 1 Control Register 3

ADC1_CTRL3	Offset	Reset Value
Measurement Unit 1 Control Register 3	18_H	see Table 441



Field	Bits	Type	Description
RES	31:20	r	**Reserved** Always read as 0
SAMPLE_TIME_LVCH	19:16	rw	**Sample time of ADC1** *Note: the asolute sampling time of a Low Voltage Channel should not be choosen lower than 200 ns. Otherwise it is not ensured that the settling time of the input signal is long enough.* 0_H **ADCCLK4**, 4 ADC_CLK clock periods 1_H **ADCCLK6**, 6 ADC_CLK clock periods 2_H **ADCCLK8**, 8 ADC_CLK clock periods(default) 3_H **ADCCLK10**, 10 ADC_CLK clock periods 4_H **ADCCLK12**, 12 ADC_CLK clock periods 5_H **ADCCLK14**, 14 ADC_CLK clock periods 6_H **ADCCLK16**, 16 ADC_CLK clock periods 7_H **ADCCLK18**, 18 ADC_CLK clock periods 8_H **ADCCLK20**, 20 ADC_CLK clock periods 9_H **ADCCLK22**, 22 ADC_CLK clock periods A_H **ADCCLK20**, 20 ADC_CLK clock periods B_H **ADCCLK20**, 20 ADC_CLK clock periods C_H **ADCCLK20**, 20 ADC_CLK clock periods D_H **ADCCLK20**, 20 ADC_CLK clock periods E_H **ADCCLK20**, 20 ADC_CLK clock periods F_H **ADCCLK20**, 20 ADC_CLK clock periods
RES	15:13	r	**Reserved** Always read as 0

Field	Bits	Type	Description
SAMPLE_TIME_HVCH	12:8	rw	**Sample time of ADC1** *Note: the absolute sampling time of a High Voltage Channel should not be choosen lower than 600 ns. Otherwise it is not ensured that the settling time of the input signal is long enough.* 00_H **ADCCLK4**, 4 ADC_CLK clock periods 01_H **ADCCLK6**, 6 ADC_CLK clock periods 02_H **ADCCLK8**, 8 ADC_CLK clock periods 03_H **ADCCLK10**, 10 ADC_CLK clock periods 04_H **ADCCLK12**, 12 ADC_CLK clock periods 05_H **ADCCLK14**, 14 ADC_CLK clock periods 06_H **ADCCLK16**, 16 ADC_CLK clock periods 07_H **ADCCLK18**, 18 ADC_CLK clock periods 08_H **ADCCLK20**, 20 ADC_CLK clock periods 09_H **ADCCLK22**, 22 ADC_CLK clock periods $0A_H$ **ADCCLK24**, 24 ADC_CLK clock periods (default) $0B_H$ **ADCCLK26**, 26 ADC_CLK clock periods $0C_H$ **ADCCLK28**, 28 ADC_CLK clock periods $0D_H$ **ADCCLK30**, 30 ADC_CLK clock periods $0E_H$ **ADCCLK32**, 32 ADC_CLK clock periods $0F_H$ **ADCCLK34**, 34 ADC_CLK clock periods 10_H **ADCCLK36**, 36 ADC_CLK clock periods 11_H **ADCCLK38**, 38 ADC_CLK clock periods 12_H **ADCCLK40**, 40 ADC_CLK clock periods 13_H **ADCCLK42**, 42 ADC_CLK clock periods 14_H **ADCCLK44**, 44 ADC_CLK clock periods 15_H **ADCCLK46**, 46 ADC_CLK clock periods 16_H **ADCCLK48**, 48 ADC_CLK clock periods 17_H **ADCCLK50**, 50 ADC_CLK clock periods 18_H **ADCCLK52**, 52 ADC_CLK clock periods 19_H **ADCCLK54**, 54 ADC_CLK clock periods $1A_H$ **ADCCLK56**, 56 ADC_CLK clock periods $1B_H$ **ADCCLK58**, 58 ADC_CLK clock periods $1C_H$ **ADCCLK60**, 60 ADC_CLK clock periods $1D_H$ **ADCCLK62**, 62 ADC_CLK clock periods $1E_H$ **n.u.**, not used $1F_H$ **n.u.**, not used
MCM_RDY	7	r	**Ready Signal for MCM[1] after Power On or Reset** 0_B **MCM Not Ready**, Measurement Core Module in startup phase 1_B **MCM Ready**, Measurement Core Module start-up phase finished

Field	Bits	Type	Description
EoC_FAIL	6	rwhxr	**Fail of ADC End of Conversion Signal** 0_B **ADC EoC available**, End of Conversion Signal was sent properly by ADC 1_B **ADC EoC not available**, End of Conversion Signal was not sent properly by ADC
RES	5	r	**Reserved** Always read as 0
EoC_FAIL_CLR	4	w	**Fail of ADC End of Conversion Signal Clear** 0_B **ADC EoC Fail not clear**, no clear of EoC_FAIL flag 1_B **ADC EoC Fail clear**, Clear of EoC_FAIL flag
RES	3:2	r	**Reserved** Always read as 0
SW_MODE	1	rw	**Flag to enter SW Mode** 0_B **Software Mode Disable**, Sequencer running 1_B **Software Mode Enabled**, Sequencer stopped
MCM_PD_N	0	rw	**Power Down Signal for MCM** 0_B **MCM Disabled**, Measurement Core Module Disabled 1_B **MCM Enabled**, Measurement Core Module Enabled

1) MCM = Measurement Core Module

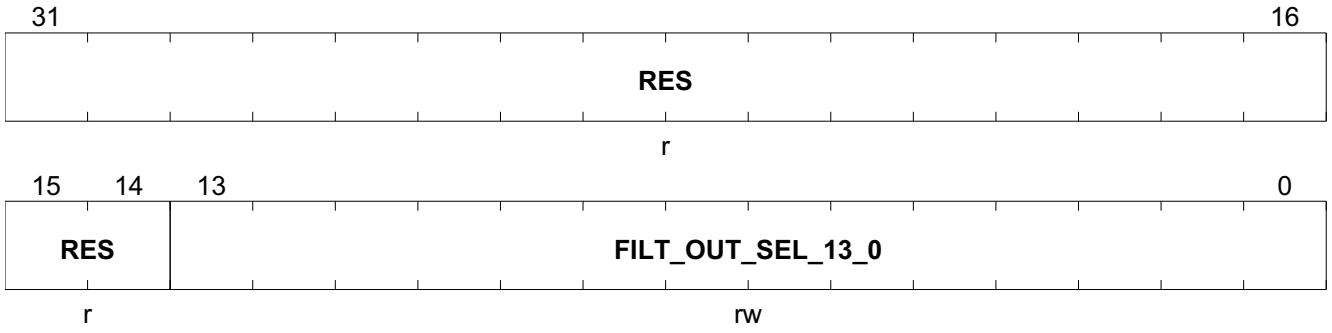
Table 441 RESET of ADC1_CTRL3

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	0002 $0A01_H$	RESET_TYPE_4		
TRIM_100_TP	0002 $0A01_H$	RESET		

Analog Digital Converter ADC10B (ADC1)

Measurement Unit 1 Control Register 5

ADC1_CTRL5	Offset	Reset Value
Measurement Unit 1 Control Register 5	$1C_H$	see Table 442



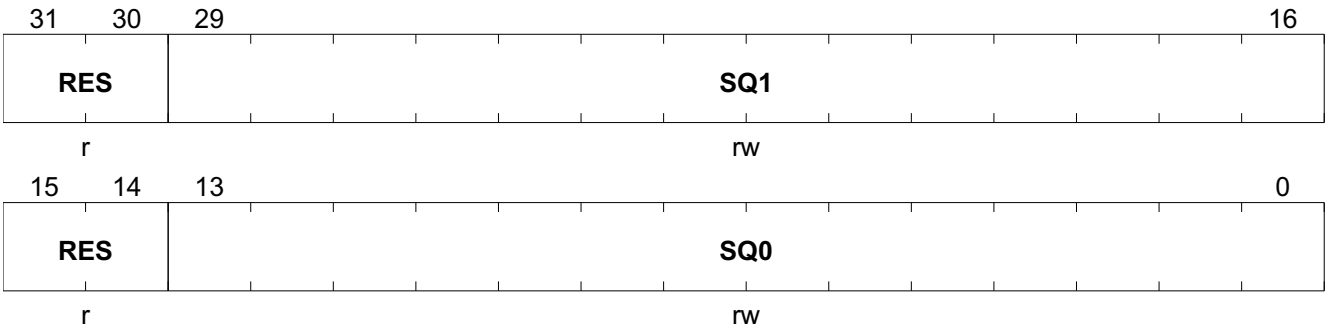
Field	Bits	Type	Description
RES	31:14	r	**Reserved** Always read as 0
FILT_OUT_SEL_13_0	13:0	rw	**Output Filter Selection for Channels 0 to 13** Each bit enables the IIR filter for the corresponding channel. $0000\ 0000\ 0000_B$ **ADC1 Unfiltered Data can be monitored in the corresponding FILT_OUTx Registers**, 0001_H **Channel 0 IIR Data enabled for FILT_OUT0 Register**, 0002_H **Channel 1 IIR Data enabled for FILT_OUT1 Register**, 0004_H **Channel 2 IIR Data enabled for FILT_OUT2 Register**, 0008_H **Channel 3 IIR Data enabled for FILT_OUT3 Register**, 0010_H **Channel 4 IIR Data enabled for FILT_OUT4 Register**, 0020_H **Channel 5 IIR Data enabled for FILT_OUT5 Register**, 0040_H **Channel 6 IIR Data enabled for FILT_OUT6 Register**, 0080_H **Channel 7 IIR Data enabled for FILT_OUT7 Register**, 0100_H **Channel 8 IIR Data enabled for FILT_OUT8 Register**, 0200_H **Channel 9 IIR Data enabled for FILT_OUT9 Register**, 0400_H **Channel 10 IIR Data enabled for FILT_OUT10 Register**, 0800_H **Channel 11 IIR Data enabled for FILT_OUT11 Register**, 1000_H **Channel 12 IIR Data enabled for FILT_OUT12 Register**, 2000_H **Channel 13 IIR Data enabled for FILT_OUT13 Register**, $3FFF_H$ **For Channels 13-0 IIR Data is enabled for FILT_OUTx Registers**,

Table 442 RESET of ADC1_CTRL5

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		

Measurement Unit 1 Channel Enable Bits for Cycle 0 - 1

ADC1_SQ0_1	Offset	Reset Value
Measurement Unit 1 Channel Enable Bits for Cycle 0-1	20_H	see Table 443



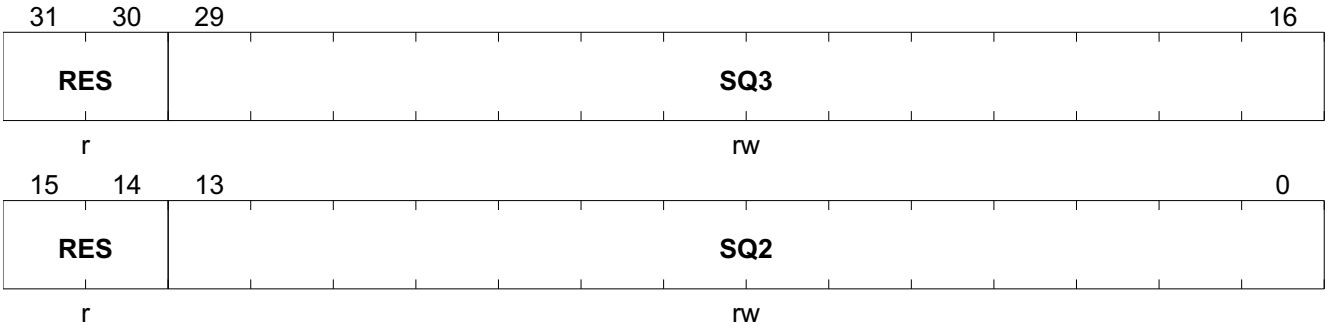
Field	Bits	Type	Description
RES	31:30	r	**Reserved** Always read as 0
SQ1	29:16	rw	**Sequence 1 channel enable** Each bit enables the corresponding channel in the sequence 1.
RES	15:14	r	**Reserved** Always read as 0
SQ0	13:0	rw	**Sequence 0 channel enable** Each bit enables the corresponding channel in the sequence 0.

Table 443 RESET of ADC1_SQ0_1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	RESET		

Measurement Unit 1 Channel Enable Bits for Cycle 2 - 3

ADC1_SQ2_3	Offset	Reset Value
Measurement Unit 1 Channel Enable Bits for Cycle 2-3	24_H	see Table 444



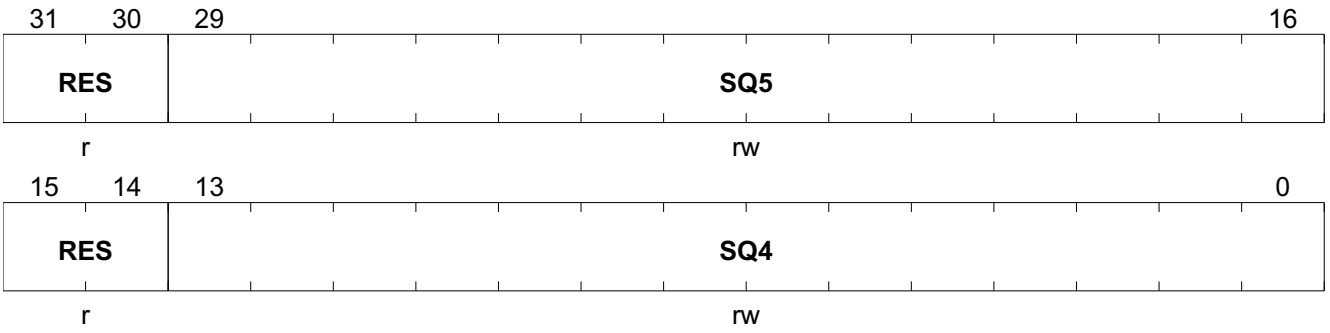
Field	Bits	Type	Description
RES	31:30	r	**Reserved** Always read as 0
SQ3	29:16	rw	**Sequence 3 channel enable** Each bit enables the corresponding channel in the sequence 3.
RES	15:14	r	**Reserved** Always read as 0
SQ2	13:0	rw	**Sequence 2 channel enable** Each bit enables the corresponding channel in the sequence 2.

Table 444 RESET of ADC1_SQ2_3

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	0000 0000_H	RESET_TYPE_4		
TRIM_100_TP	0000 0000_H	RESET		

Measurement Unit 1 Channel Enable Bits for Cycle 4-5

ADC1_SQ4_5	Offset	Reset Value
Measurement Unit 1 Channel Enable Bits for Cycle 4-5	28_H	see Table 445



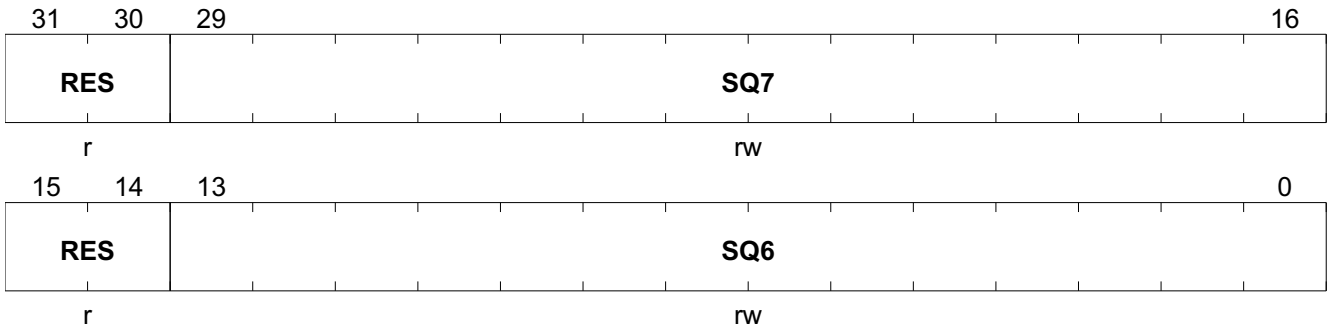
Field	Bits	Type	Description
RES	31:30	r	**Reserved** Always read as 0
SQ5	29:16	rw	**Sequence 5 channel enable** Each bit enables the corresponding channel in the sequence 5.
RES	15:14	r	**Reserved** Always read as 0
SQ4	13:0	rw	**Sequence 4 channel enable** Each bit enables the corresponding channel in the sequence 4.

Table 445 RESET of ADC1_SQ4_5

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	0000 0000_H	RESET_TYPE_4		
TRIM_100_TP	0000 0000_H	RESET		

Measurement Unit 1 Channel Enable Bits for Cycle 6-7

ADC1_SQ6_7	Offset	Reset Value
Measurement Unit 1 Channel Enable Bits for Cycle 6-7	$2C_H$	see Table 446



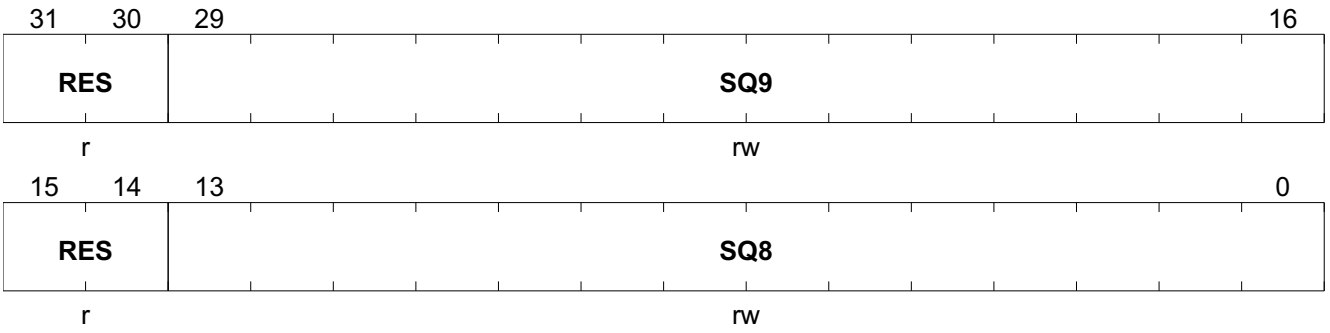
Field	Bits	Type	Description
RES	31:30	r	**Reserved** Always read as 0
SQ7	29:16	rw	**Sequence 7 channel enable** Each bit enables the corresponding channel in the sequence 7.
RES	15:14	r	**Reserved** Always read as 0
SQ6	13:0	rw	**Sequence 6 channel enable** Each bit enables the corresponding channel in the sequence 6.

Table 446 RESET of ADC1_SQ6_7

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	RESET		

Measurement Unit 1 Channel Enable Bits for Cycle 8-9

ADC1_SQ8_9	Offset	Reset Value
Measurement Unit 1 Channel Enable Bits for Cycle 8-9	30_H	see Table 447



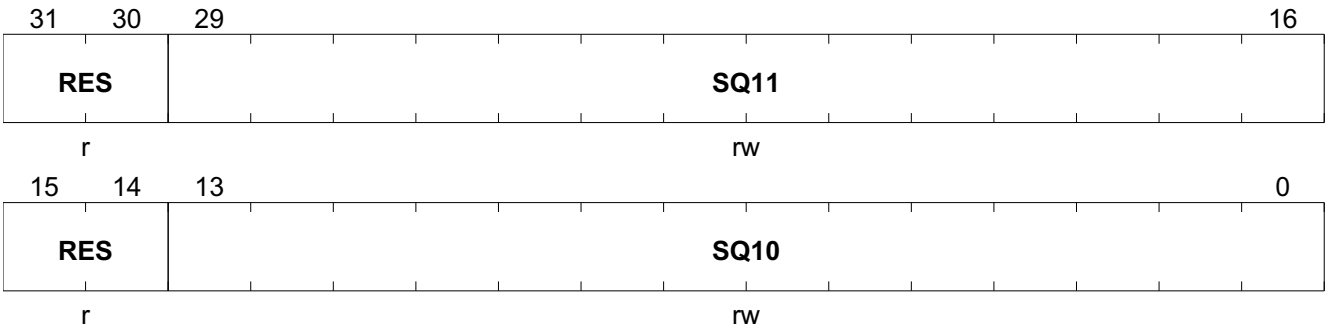
Field	Bits	Type	Description
RES	31:30	r	**Reserved** Always read as 0
SQ9	29:16	rw	**Sequence 9 channel enable** Each bit enables the corresponding channel in the sequence 9.
RES	15:14	r	**Reserved** Always read as 0
SQ8	13:0	rw	**Sequence 8 channel enable** Each bit enables the corresponding channel in the sequence 8.

Table 447 RESET of ADC1_SQ8_9

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	0000 0000_H	RESET_TYPE_4		
TRIM_100_TP	0000 0000_H	RESET		

Measurement Unit 1 Channel Enable Bits for Cycle 10-11

ADC1_SQ10_11	Offset	Reset Value
Measurement Unit 1 Channel Enable Bits for Cycle 10-11	34_H	see Table 448



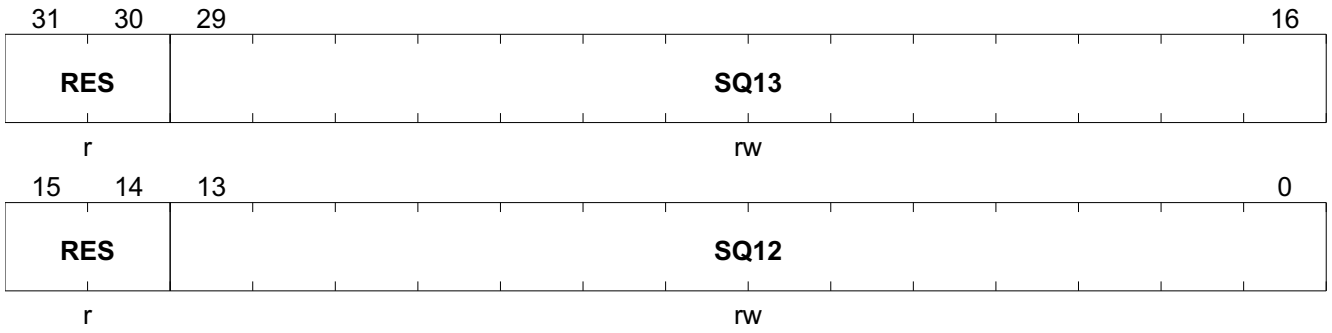
Field	Bits	Type	Description
RES	31:30	r	**Reserved** Always read as 0
SQ11	29:16	rw	**Sequence 11 channel enable** Each bit enables the corresponding channel in the sequence 11.
RES	15:14	r	**Reserved** Always read as 0
SQ10	13:0	rw	**Sequence 10 channel enable** Each bit enables the corresponding channel in the sequence 10.

Table 448 RESET of ADC1_SQ10_11

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	RESET		

Measurement Unit 1 Channel Enable Bits for Cycle 12-13

ADC1_SQ12_13	Offset	Reset Value
Measurement Unit 1 Channel Enable Bits for Cycle 12-13	130_H	see Table 449



Field	Bits	Type	Description
RES	31:30	r	**Reserved** Always read as 0
SQ13	29:16	rw	**Sequence 13 channel enable** Each bit enables the corresponding channel in the sequence 13.
RES	15:14	r	**Reserved** Always read as 0
SQ12	13:0	rw	**Sequence 12 channel enable** Each bit enables the corresponding channel in the sequence 12.

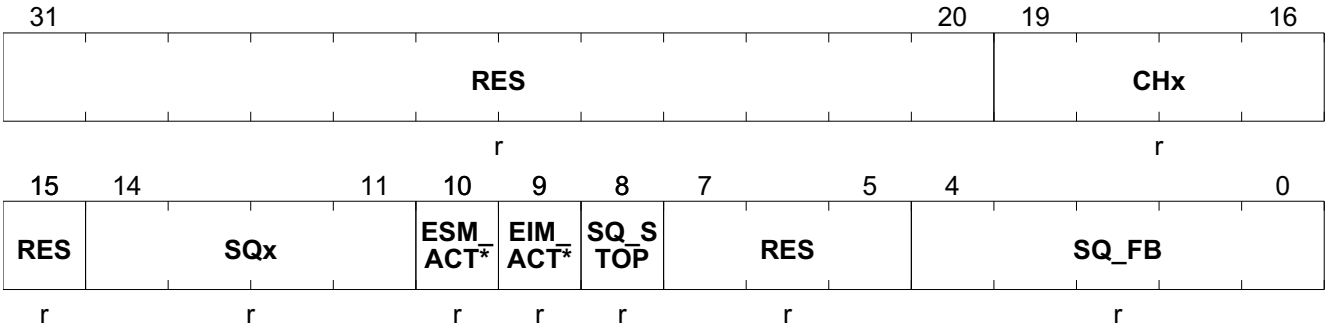
Table 449 RESET of ADC1_SQ12_13

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	RESET		

Analog Digital Converter ADC10B (ADC1)

Sequencer Feedback Register

ADC1_SQ_FB	Offset	Reset Value
Sequencer Feedback Register	04_H	see Table 450



Field	Bits	Type	Description
RES	31:20	r	**Reserved** Always read as 0
CHx	19:16	r	**Current ADC1 Channel** Other bit combinations are **reserved**, do not use. 0000_B **CH0**, Channel 0 enable 0001_B **CH1**, Channel 1 enable 0010_B **CH2**, Channel 2 enable 0011_B **CH3**, Channel 3 enable 0100_B **CH4**, Channel 4 enable 0101_B **CH5**, Channel 5 enable 0110_B **CH6**, Channel 6 enable 0111_B **CH7**, Channel 7 enable 1000_B **CH8**, Channel 8 enable 1001_B **CH9**, Channel 9 enable 1010_B **CH10**, Channel 10 enable 1011_B **CH11**, Channel 11 enable 1100_B **CH12**, Channel 12 enable 1101_B **CH13**, Channel 13 enable
RES	15	r	**Reserved** Always read as 0

Field	Bits	Type	Description
SQx	14:11	r	**Current Active ADC1 Sequence** Other bit combinations are **reserved**, do not use. 0000_B **SQ0**, Sequence 0 enable 0001_B **SQ1**, Sequence 1 enable 0010_B **SQ2**, Sequence 2 enable 0011_B **SQ3**, Sequence 3 enable 0100_B **SQ4**, Sequence 4 enable 0101_B **SQ5**, Sequence 5 enable 0110_B **SQ6**, Sequence 6 enable 0111_B **SQ7**, Sequence 7 enable 1000_B **SQ8**, Sequence 8 enable 1001_B **SQ9**, Sequence 9 enable 1010_B **SQ10**, Sequence 10 enable 1011_B **SQ11**, Sequence 11 enable 1100_B **SQ12**, Sequence 12 enable 1101_B **SQ13**, Sequence 13 enable
ESM_ACTIVE	10	r	**ADC1 ESM active** *Note: this bit indicates an active or a pending sequence measurement; a pending measurement is signalled when EIM or Software Mode is selected (modes with higher priority).* 0_B **not active** , ESM not active 1_B **active**, ESM active
EIM_ACTIVE	9	r	**ADC1 EIM active** *Note: this bit indicates an active or a pending exception measurement; a pending measurement is signalled when Software Mode is selected (mode with higher priority).* 0_B **not active** , EIM not active 1_B **active**, EIM active
SQ_STOP	8	r	**ADC1 Sequencer Stop Signal for DPP** 0_B **DPP Running**, Postprocessing Sequencer in running mode 1_B **DPP Stopped**, Postprocessing Sequencer stopped / Software Mode entered
RES	7:5	r	**Reserved** Always read as 0

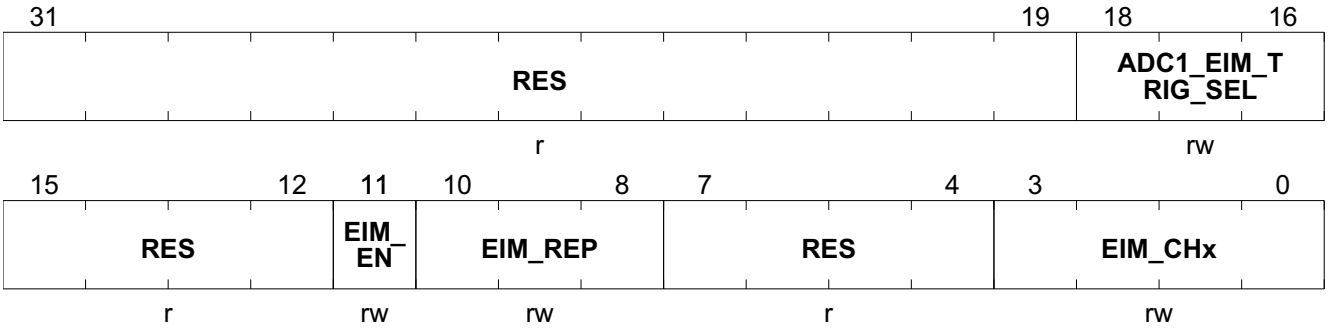
Field	Bits	Type	Description
SQ_FB	4:0	r	**Current Sequence that caused software mode** $0\ 0000_B$ **SQ0**, Sequence 0 enable $0\ 0001_B$ **SQ1**, Sequence 1 enable $0\ 0010_B$ **SQ2**, Sequence 2 enable $0\ 0011_B$ **SQ3**, Sequence 3 enable $0\ 0100_B$ **SQ4**, Sequence 4 enable $0\ 0101_B$ **SQ5**, Sequence 5 enable $0\ 0110_B$ **SQ6**, Sequence 6 enable $0\ 0111_B$ **SQ7**, Sequence 7 enable $0\ 1000_B$ **SQ8**, Sequence 8 enable $0\ 1001_B$ **SQ9**, Sequence 9 enable $0\ 1010_B$ **SQ10**, Sequence 10 enable $0\ 1011_B$ **SQ11**, Sequence 11 enable $0\ 1100_B$ **SQ12**, Sequence 12 enable $0\ 1101_B$ **SQ13**, Sequence 13 enable $0\ 1110_B$ **rfu**, $1\ 1010_B$ **ESM**, ESM $1\ 1011_B$ **rfu**, $1\ 1100_B$ **SUSPEND**, SW Mode per Flag

Table 450 RESET of ADC1_SQ_FB

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$00XX\ XX0X_H$	RESET_TYPE_4		Exact reset value is:0000 0000 000X XXXX 0XXX XXXX 0000 XXXX(binary)

Channel Setting Bits for Exceptional Interrupt Measurement

ADC1_CHx_EIM	Offset	Reset Value
Channel Setting Bits for Exceptional Interrupt Measurement	08_H	see Table 451



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
ADC1_EIM_TRIG_SEL	18:16	rw	**Trigger selection for exceptional interrupt measurement (EIM)** 000_B **NONE**, 001_B **COUT63**, 010_B **GPT12_T6OUT**, 011_B **GPT12_T3OUT**, 100_B **T2**, t2_adc_trigger 101_B **T21**, t21_adc_trigger 110_B **RES**, reserved 111_B **RES**, reserved
RES	15:12	r	**Reserved** Always read as 0
EIM_EN	11	rw	**Exceptional interrupt measurement (EIM) Trigger Event enable** Always read as 0 0_B **DISABLE**, start of EIM disabled 1_B **ENABLE**, start of EIM enabled
EIM_REP	10:8	rw	**Repeat count for exceptional interrupt measurement (EIM)** 000_B **CM**, Continous Mode 001_B **2**, Measurements (2 SOC generated for ADC10) 010_B **4**, Measurements (4 SOC generated for ADC10) 011_B **8**, Measurements (8 SOC generated for ADC10) 100_B **16**, Measurements (16 SOC generated for ADC10) 101_B **32**, Measurements (32 SOC generated for ADC10) 110_B **64**, Measurements (64 SOC generated for ADC10) 111_B **128**, Measurements (128 SOC generated for ADC10)
RES	7:4	r	**Reserved** Always read as 0

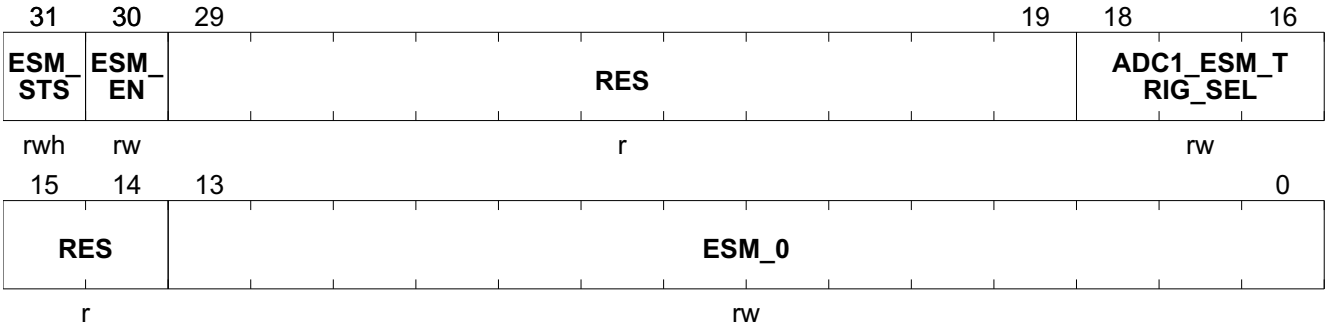
Field	Bits	Type	Description
EIM_CHx	3:0	rw	**Channel set for exceptional interrupt measurement (EIM)** 0000_B **CH0_EN**, Channel 0 enable 0001_B **CH1_EN**, Channel 1 enable 0010_B **CH2_EN**, Channel 2 enable 0011_B **CH3_EN**, Channel 3 enable 0100_B **CH4_EN**, Channel 4 enable 0101_B **CH5_EN**, Channel 5 enable 0110_B **CH6_EN**, Channel 6 enable 0111_B **CH7_EN**, Channel 7 enable 1000_B **CH8_EN**, Channel 8 enable 1001_B **CH9_EN**, Channel 9 enable 1010_B **rfu**, reserved for future use 1011_B **rfu**, reserved for future use 1100_B **CH12_EN**, Channel 12 enable 1101_B **CH13_EN**, Channel 13 enable 1110_B **rfu**, reserved for future use 1111_B **rfu**, reserved for future use

Table 451 RESET of ADC1_CHx_EIM

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	RESET		

Channel Setting Bits for Exceptional Sequence Measurement

ADC1_CHx_ESM	Offset	Reset Value
Channel Setting Bits for Exceptional Sequence Measurement	$0C_H$	see Table 452



Field	Bits	Type	Description
ESM_STS	31	rwh	**Exceptional Sequence Measurement is finished** 0_B **not active**, Exceptional Sequence Measurement not done 1_B **done**, Exceptional Sequence Measurement done
ESM_EN	30	rw	**Enable for Exceptional Sequence Measurement Trigger Event** 0_B **Disable**, start of ESM disabled 1_B **Enable**, start of ESM enabled
RES	29:19	r	**Reserved** Always read as 0
ADC1_ESM_TRIG_SEL	18:16	rw	**Trigger selection for exceptional interrupt measurement (ESM)** 000_B **NONE**, 001_B **COUT63**, 010_B **GPT12_T6OUT**, 011_B **GPT12_T3OUT**, 100_B **T2**, t2_adc_trigger 101_B **T21**, t21_adc_trigger 110_B **RES**, reserved 111_B **RES**, reserved
RES	15:14	r	**Reserved** Always read as 0

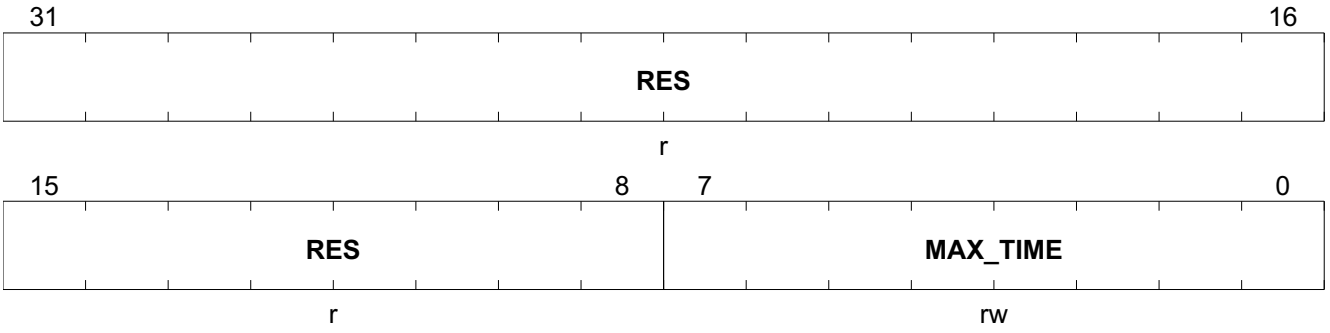
Field	Bits	Type	Description
ESM_0	13:0	rw	**Channel Sequence for Exceptional Sequence Measurement (ESM)** The following values can be ored: 0001_H **CH0_EN**, Channel 0 enable 0002_H **CH1_EN**, Channel 1 enable 0004_H **CH2_EN**, Channel 2 enable 0008_H **CH3_EN**, Channel 3 enable 0010_H **CH4_EN**, Channel 4 enable 0020_H **CH5_EN**, Channel 5 enable 0040_H **CH6_EN**, Channel 6 enable 0080_H **CH7_EN**, Channel 7 enable 0100_H **CH8_EN**, Channel 8 enable 0200_H **CH9_EN**, Channel 9 enable 0400_H **CH10_EN**, Channel 10 enable 0800_H **CH11_EN**, Channel 11 enable 1000_H **CH12_EN**, Channel 12 enable 2000_H **CH13_EN**, Channel 13 enable

Table 452 RESET of ADC1_CHx_ESM

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	RESET		

Maximum Time for Software Mode

ADC1_MAX_TIME	Offset	Reset Value
Maximum Time for Software Mode	10_H	see Table 453



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Always read as 0
MAX_TIME	7:0	rw	**Maximum Time in Software Mode** Maximum time in Software Mode with the unit of 50 ns. Software mode is active for ADC1_MAX_TIME * 50 ns *Note: MAX_TIME to be set >= 8 to enter SW Mode* 00_H **min**, Software mode is immediately left FF_H **max**, Software mode is active for 12.75 us

Table 453 RESET of ADC1_MAX_TIME

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	0000 0000_H	RESET_TYPE_4		
TRIM_100_TP	0000 0000_H	RESET		

24.6 Calibration Unit

24.6.1 Functional Description

The calibration unit of the Measurement Core module is dedicated to cancel offset and gain errors out of the signal chain. The upcoming two chapter describe usage and setup of the calibration unit.

24.6.1.1 Method for determining the Calibration Parameters

As mentioned in the introduction of the calibration unit, the module can be used to correct gain and offset errors caused by non-idealities in the measurement chain. These non-idealities are caused by the corresponding measurement chain modules.

Those first order non-idealities are:

- Offset and Gain Error of ADC1.
- Offset and Gain Error of the Attenuator (especially voltage measurement).
- Offset and Gain Error of Reference Voltage.

All these factors are summed up in the overall Gain (factor **b**) and overall Offset (adder **a**) of the complete measurement chain. They are calculated from a two point test result and stored inside the NVM.

24.6.1.2 Setup of Calibration Unit

Each channel has its own calibration unit and thus also its dedicated Gain and Offset parameter. These parameters are stored in a 100TP page of the Flash Module. After each reset of RESET_TYPE_4 these coefficients are downloaded from NVM into the corresponding registers. The user may not take care about the configuration of these parameters. After this has been done, the values are used for the correction procedure. The figure below shows the formula performed by the calibration unit and the required **sfr**-Register to control its functionality.

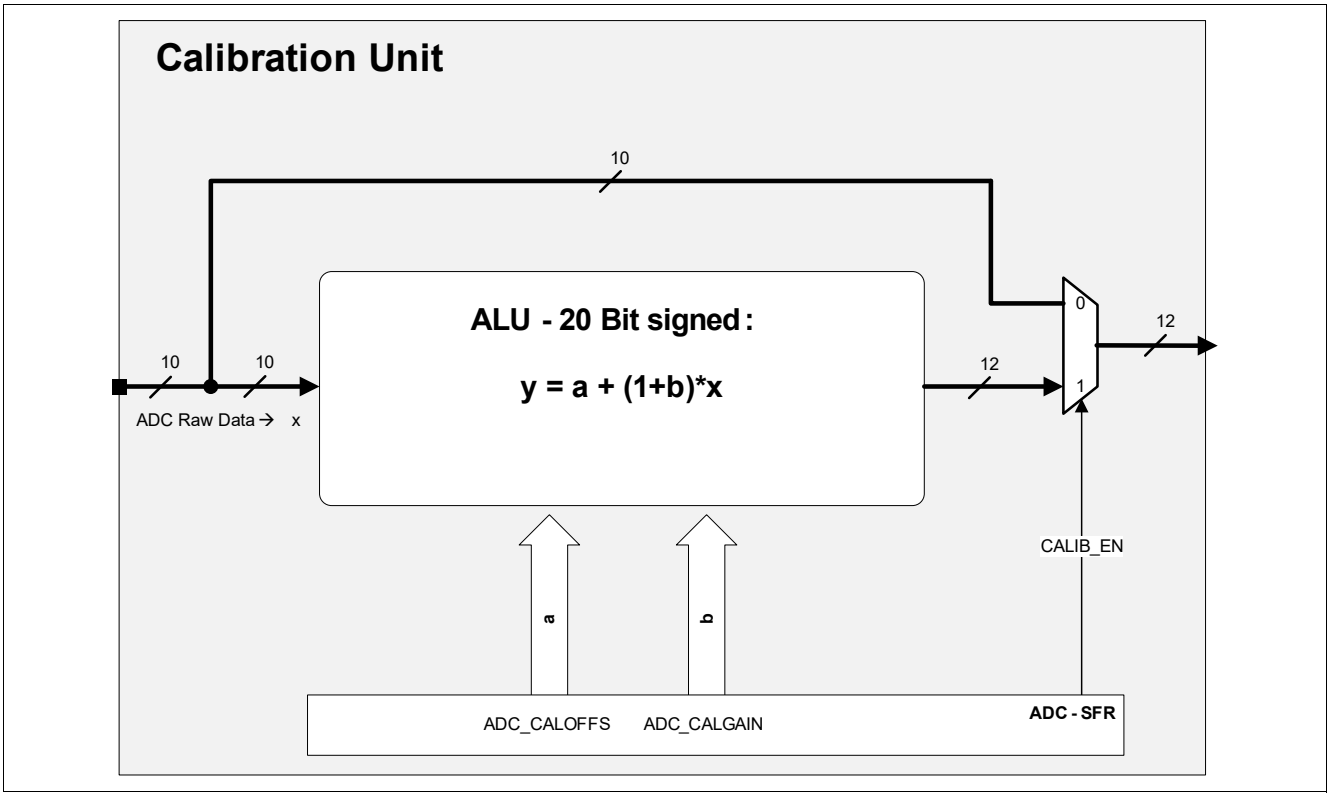


Figure 198 Structure of Calibration Unit

24.6.2 Calibration Unit Control Registers

The Calibration Unit can be configured by the **SFR** Register shown below. All calibration registers can be written by the user. This allows an in-system recalibration of a dedicated measurement.

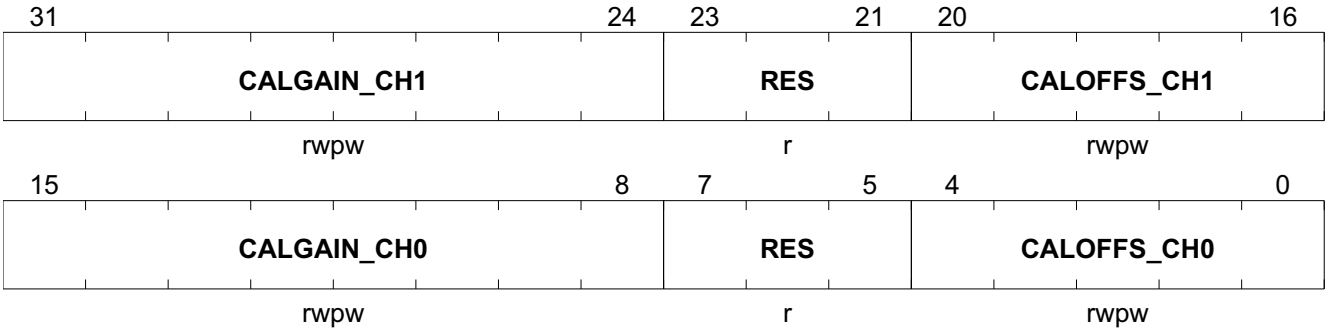
Table 454 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Calibration Unit Control Registers,			
ADC1_CAL_CH0_1	Calibration for Channel 0 and 1	48_H	see **Table 455**
ADC1_CAL_CH2_3	Calibration for Channel 2 and 3	$4C_H$	see **Table 456**
ADC1_CAL_CH4_5	Calibration for Channel 4 and 5	50_H	see **Table 457**
ADC1_CAL_CH6_7	Calibration for Channel 6 and 7	54_H	see **Table 458**
ADC1_CAL_CH8_9	Calibration for Channel 8 and 9	58_H	see **Table 459**
ADC1_CAL_CH10_11	Calibration for Channel 10 and 11	$5C_H$	see **Table 460**
ADC1_CAL_CH12_13	Calibration for Channel 12 and 13	138_H	see **Table 461**

The registers are addressed wordwise.

Measurement Unit 1 Calibration for Channel 0 & 1

ADC1_CAL_CH0_1	Offset	Reset Value
Calibration for Channel 0 & 1	48_H	see Table 455



Field	Bits	Type	Description
CALGAIN_CH1	31:24	rwpw	**Gain Calibration for channel 1** For ADC output set CALIB_EN_1 = 0
RES	23:21	r	**Reserved** Always read as 0
CALOFFS_CH1	20:16	rwpw	**Offset Calibration for channel 1** For ADC output set CALIB_EN_1 = 0
CALGAIN_CH0	15:8	rwpw	**Gain Calibration for channel 0** For ADC output set CALIB_EN_0 = 0

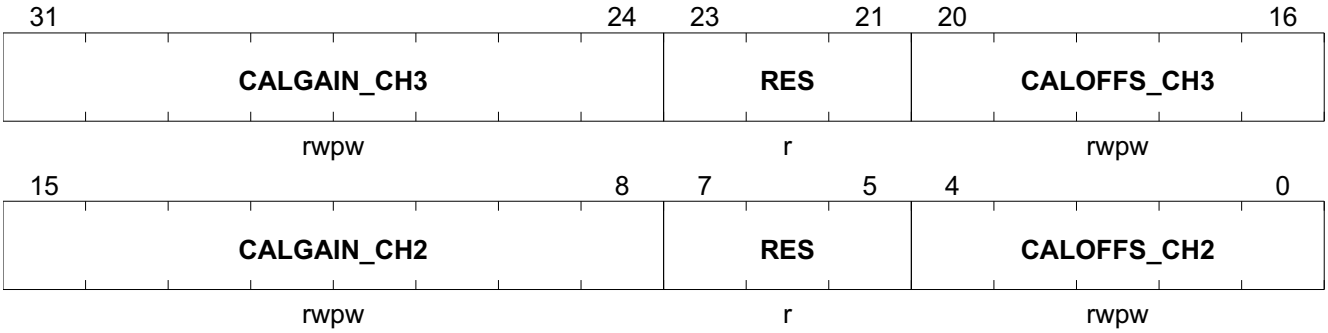
Field	Bits	Type	Description
RES	7:5	r	**Reserved** Always read as 0
CALOFFS_CH0	4:0	rwpw	**Offset Calibration for channel 0** For ADC output set CALIB_EN_0 = 0

Table 455 RESET of ADC1_CAL_CH0_1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	TRIM		

Measurement Unit 1 Calibration for Channel 2 & 3

ADC1_CAL_CH2_3	Offset	Reset Value
Calibration for Channel 2 & 3	$4C_H$	see Table 456



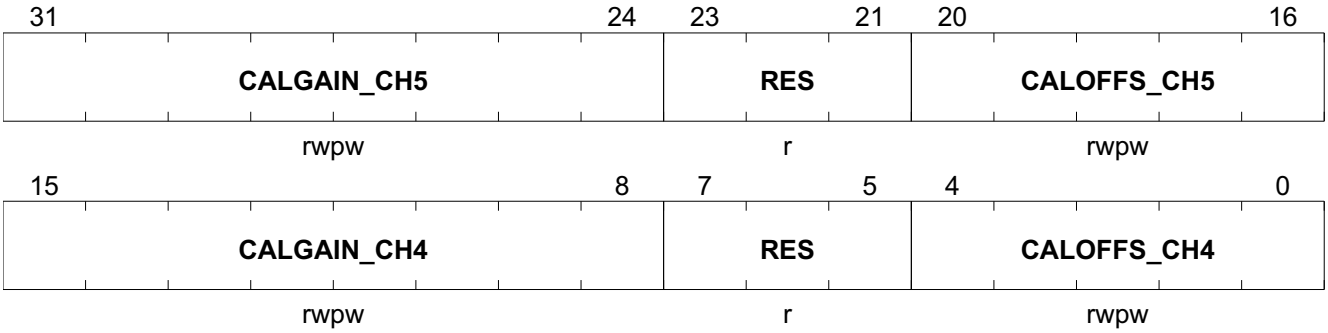
Field	Bits	Type	Description
CALGAIN_CH3	31:24	rwpw	**Gain Calibration for channel 3** For ADC output set CALIB_EN_3 = 0
RES	23:21	r	**Reserved** Always read as 0
CALOFFS_CH3	20:16	rwpw	**Offset Calibration for channel 3** For ADC output set CALIB_EN_3 = 0
CALGAIN_CH2	15:8	rwpw	**Gain Calibration for channel 2** For ADC output set CALIB_EN_2 = 0
RES	7:5	r	**Reserved** Always read as 0
CALOFFS_CH2	4:0	rwpw	**Offset Calibration for channel 2** For ADC output set CALIB_EN_2 = 0

Table 456 RESET of ADC1_CAL_CH2_3

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	TRIM		

Measurement Unit 1 Calibration for Channel 4 & 5

ADC1_CAL_CH4_5	Offset	Reset Value
Calibration for Channel 4 & 5	50_H	see Table 457



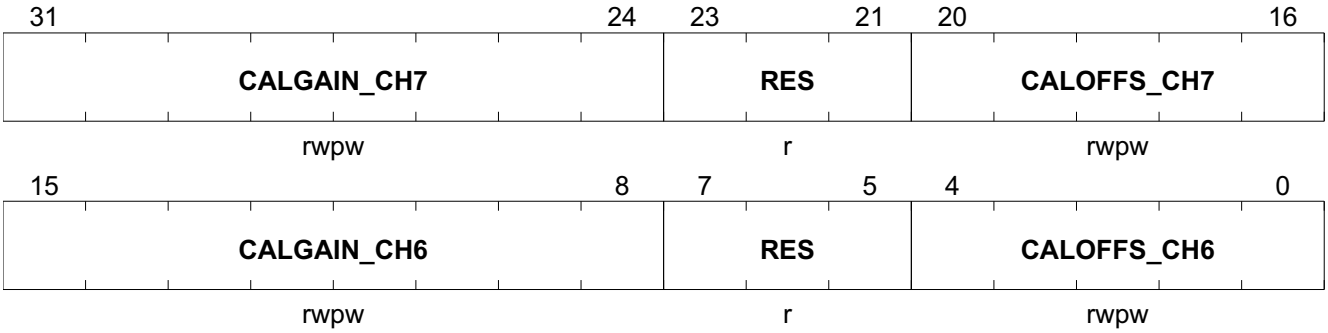
Field	Bits	Type	Description
CALGAIN_CH5	31:24	rwpw	**Gain Calibration for channel 5** For ADC output set CALIB_EN_5 = 0
RES	23:21	r	**Reserved** Always read as 0
CALOFFS_CH5	20:16	rwpw	**Offset Calibration for channel 5** For ADC output set CALIB_EN_5 = 0
CALGAIN_CH4	15:8	rwpw	**Gain Calibration for channel 4** For ADC output set CALIB_EN_4 = 0
RES	7:5	r	**Reserved** Always read as 0
CALOFFS_CH4	4:0	rwpw	**Offset Calibration for channel 4** For ADC output set CALIB_EN_4 = 0

Table 457 RESET of ADC1_CAL_CH4_5

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	TRIM		

Measurement Unit 1 Calibration for Channel 6 & 7

ADC1_CAL_CH6_7	Offset	Reset Value
Calibration for Channel 6 & 7	54_H	see Table 458



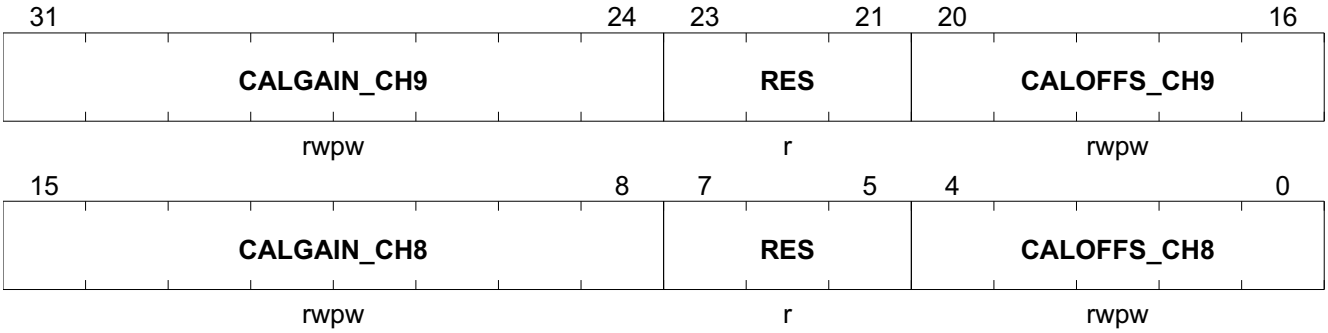
Field	Bits	Type	Description
CALGAIN_CH7	31:24	rwpw	**Gain Calibration for channel 7** For ADC output set CALIB_EN_7 = 0
RES	23:21	r	**Reserved** Always read as 0
CALOFFS_CH7	20:16	rwpw	**Offset Calibration for channel 7** For ADC output set CALIB_EN_7 = 0
CALGAIN_CH6	15:8	rwpw	**Gain Calibration for channel 6** For ADC output set CALIB_EN_6 = 0
RES	7:5	r	**Reserved** Always read as 0
CALOFFS_CH6	4:0	rwpw	**Offset Calibration for channel 6** For ADC output set CALIB_EN_6 = 0

Table 458 RESET of ADC1_CAL_CH6_7

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	TRIM		

Measurement Unit 1 Calibration for Channel 8 & 9

ADC1_CAL_CH8_9	Offset	Reset Value
Calibration for Channel 8 & 9	58_H	see Table 459



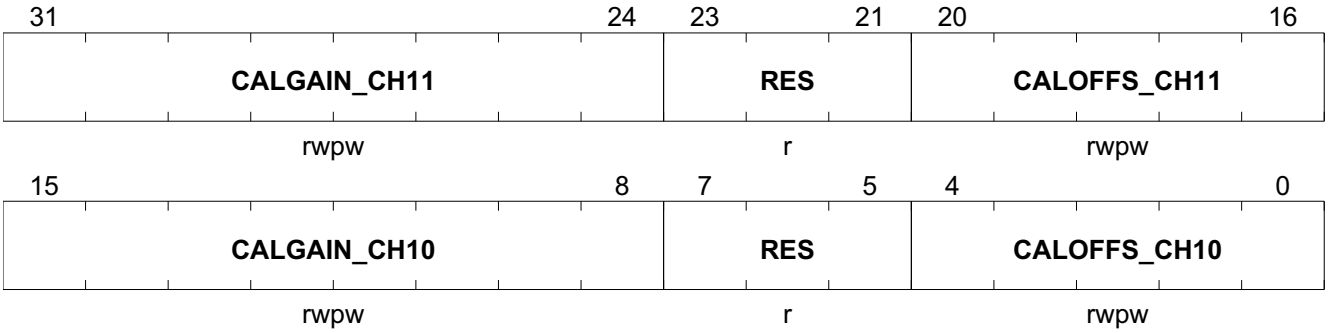
Field	Bits	Type	Description
CALGAIN_CH9	31:24	rwpw	**Gain Calibration for channel 9** For ADC output set CALIB_EN_9 = 0
RES	23:21	r	**Reserved** Always read as 0
CALOFFS_CH9	20:16	rwpw	**Offset Calibration for channel 9** For ADC output set CALIB_EN_9 = 0
CALGAIN_CH8	15:8	rwpw	**Gain Calibration for channel 8** For ADC output set CALIB_EN_8 = 0
RES	7:5	r	**Reserved** Always read as 0
CALOFFS_CH8	4:0	rwpw	**Offset Calibration for channel 8** For ADC output set CALIB_EN_8 = 0

Table 459 RESET of ADC1_CAL_CH8_9

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	TRIM		

Measurement Unit 1 Calibration for Channel 10 & 11

ADC1_CAL_CH10_11	Offset	Reset Value
Calibration for Channel 10 & 11	$5C_H$	see Table 460



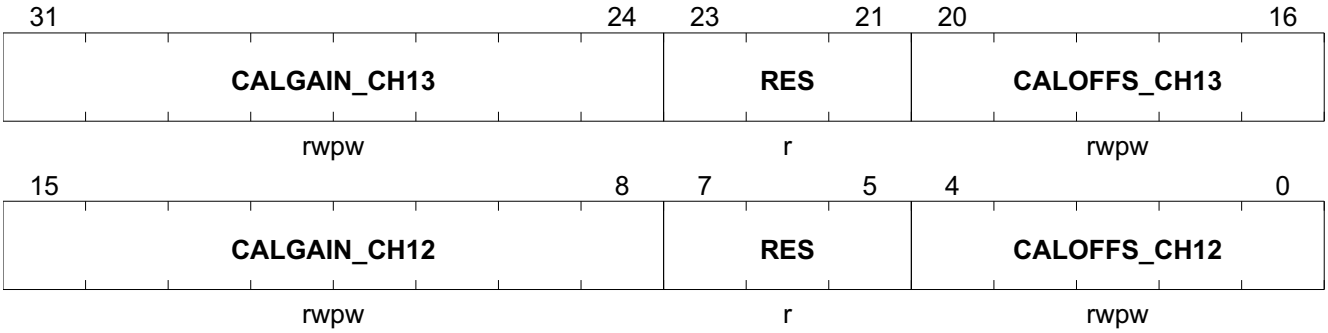
Field	Bits	Type	Description
CALGAIN_CH11	31:24	rwpw	**Gain Calibration for channel 11** For ADC output set CALIB_EN_11 = 0
RES	23:21	r	**Reserved** Always read as 0
CALOFFS_CH11	20:16	rwpw	**Offset Calibration for channel 11** For ADC output set CALIB_EN_11 = 0
CALGAIN_CH10	15:8	rwpw	**Gain Calibration for channel 10** For ADC output set CALIB_EN_10 = 0
RES	7:5	r	**Reserved** Always read as 0
CALOFFS_CH10	4:0	rwpw	**Offset Calibration for channel 10** For ADC output set CALIB_EN_10 = 0

Table 460 RESET of ADC1_CAL_CH10_11

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	TRIM		

Measurement Unit 1 Calibration for Channel 12 & 13

ADC1_CAL_CH12_13	Offset	Reset Value
Calibration for Channel 12 & 13	138_H	see Table 461



Field	Bits	Type	Description
CALGAIN_CH13	31:24	rwpw	**Gain Calibration for channel 13** For ADC output set CALIB_EN_13 = 0
RES	23:21	r	**Reserved** Always read as 0
CALOFFS_CH13	20:16	rwpw	**Offset Calibration for channel 13** For ADC output set CALIB_EN_13 = 0
CALGAIN_CH12	15:8	rwpw	**Gain Calibration for channel 12** For ADC output set CALIB_EN_12 = 0
RES	7:5	r	**Reserved** Always read as 0
CALOFFS_CH12	4:0	rwpw	**Offset Calibration for channel 12** For ADC output set CALIB_EN_12 = 0

Table 461 RESET of ADC1_CAL_CH12_13

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	TRIM		

24.7 IIR-Filter

24.7.1 Functional Description

To cancel low frequency noise out of the measured signal, every channel of the digital signal includes a first order IIR Filter. The structure of the IIR Filter is shown in the picture below.

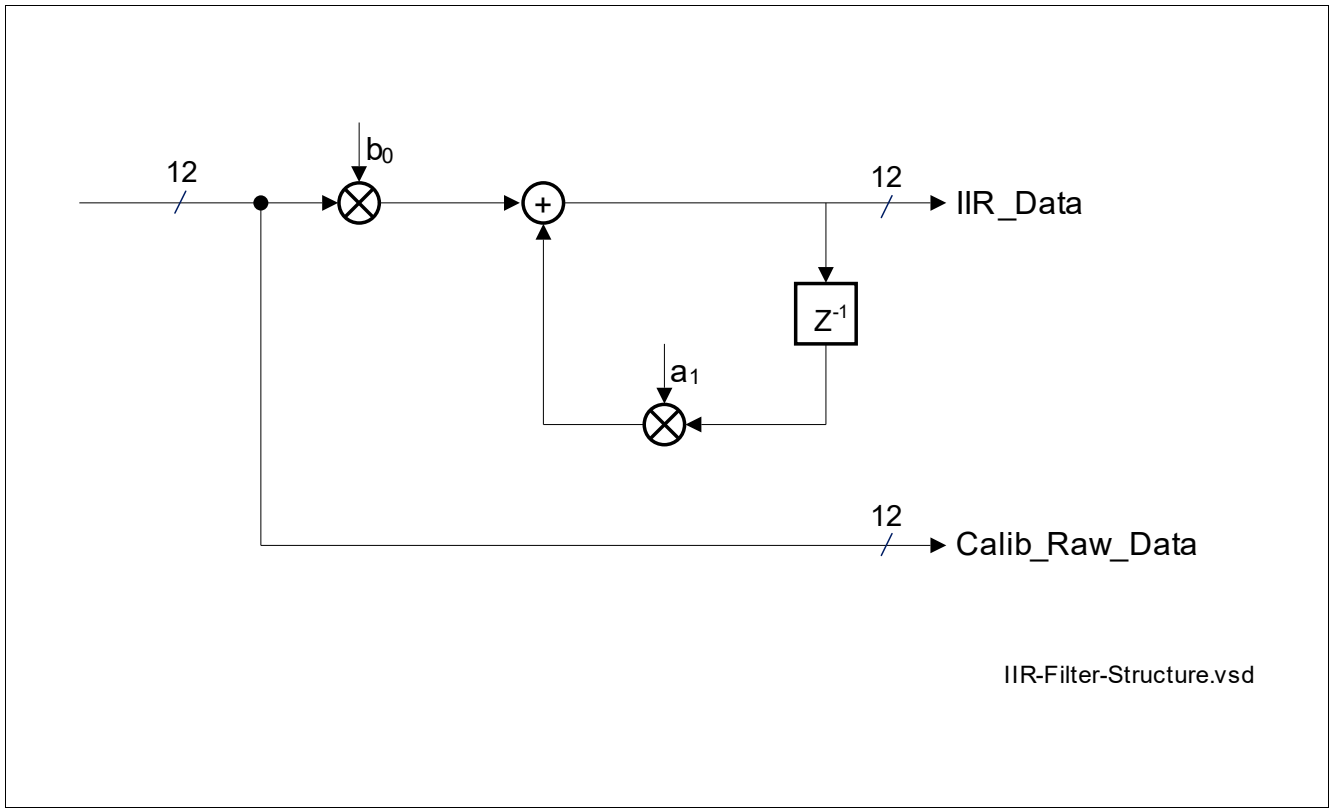


Figure 199 IIR-Filter Implementation Structure

$$H_{IIR}(z) = \frac{b}{1 - a * z^{-1}} \tag{24.5}$$

This filter allows an effective suppression of high-frequency components like noise or crosstalk caused by HF-components in order to avoid the generation of unwanted interrupts. The coefficient b can be expressed as:

$$b = 1 - a \tag{24.6}$$

The IIR Filter transfer function is shown in the plot below.

$$H_{IIR}(z) = \frac{1 - a}{(1 - a * z^{-1})} \tag{24.7}$$

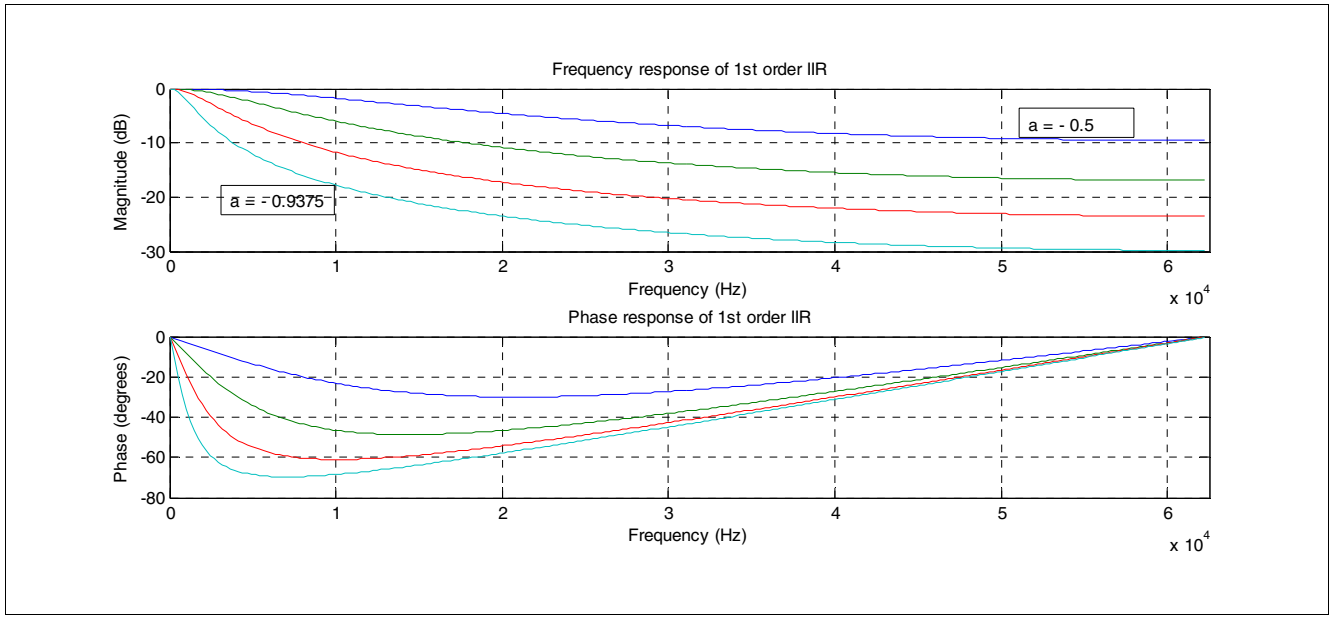


Figure 200 IIR filter transfer function for different filter length fl (1MHz corresponds to 1/2*channel sampling frequency)

24.7.1.1 Step Response

The IIR filter's step response time is shown in the figure below:

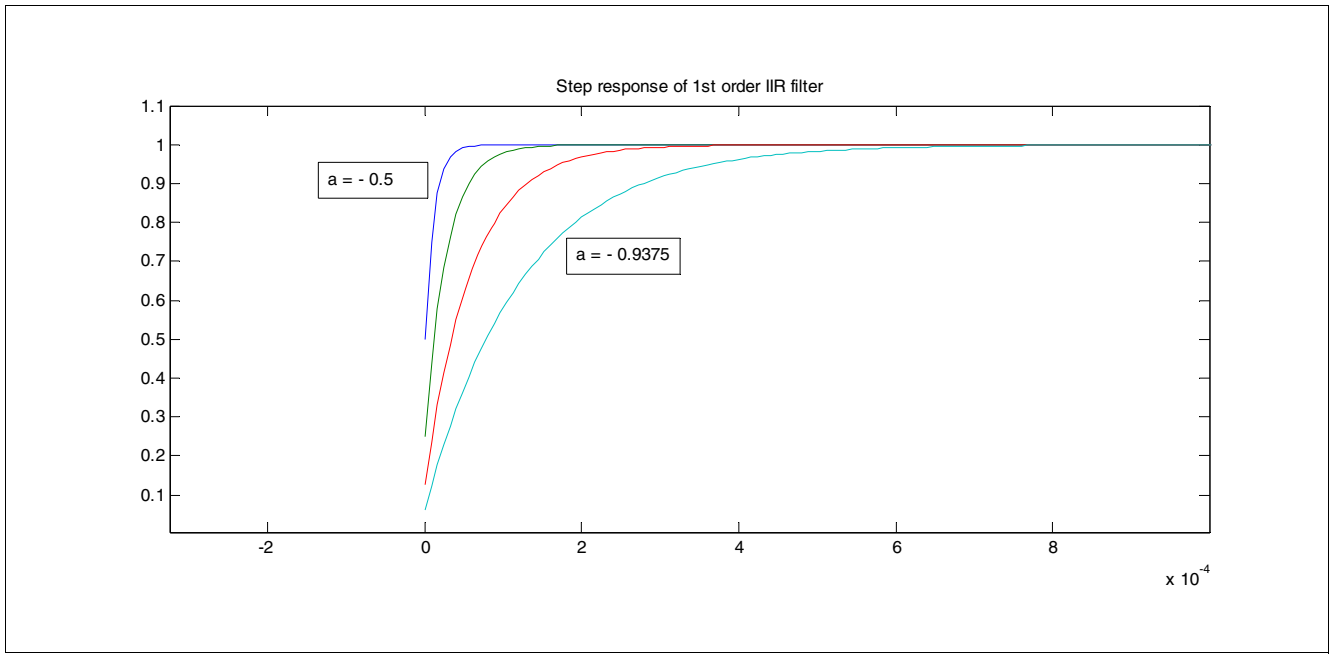


Figure 201 IIR Step Response Time

Table 462 summarizes the main filter characteristics.

Table 462 IIR filter characteristics

Filter coefficient	Group delay at=ω0	Normalized -3dB frequency [1]	-3dB frequency @ $f_{s\_ch}/2$=250 kHz
a	**τ[samples]**	**$f_{-3dB}/(f_{s\_ch}/2)$**	**f_{-3dB} [Hz]**
2^{-1}	2		
2^{-2}	4		
2^{-3}	8		
2^{-4}	16		

1) The filter's - 3dB frequency is normalized to half the channel sampling frequency (Nyquist frequency)

24.7.2 IIR Filter Control Registers

The IIR Filter can also be configured by the **sfr** Register shown below.

The **ADC1_FILT_OUT0** to **ADC1_FILT_OUT13** registers are 12 bits wide, but the ADC delivers only a resolution of 10 bits. Bits 1:0 of ADC1_FILT_OUTx contain two bits fractional part (2^{-1},2^{-2}) after calibration and filtering, increasing the resolution to 1/4 LSB. **Table 463** shows how the lower two bits are determined.

Table 463 ADC1_FILT_OUT register setting

ADC1_CTRL2.calib_en	ADC1_CTRL5.filt_out_sel	ADC1_FILT_OUT0.output[1:0]
0	0	"00"[1]
0	1	2 bits fractional part after filtering
1	0	2 bits fractional part after calibration
1	1	2 bits fractional part after calibration and filtering

1) ADC1_FILT_OUTx 11:2 contains the 10 bit ADC output value if calibration and filtering are disabled.

The result of the calibration unit is 12 bits, the output is feed into the IIR filter. The internal result of the IIR filter is 14 bits, the output is converted to 10 bit and feed into the postprocessing. The user can monitor the calculated values in the **ADC1_FILT_OUT0** to **ADC1_FILT_OUT13** registers and gets access to 12 bit wide result information.

Table 464 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
IIR Filter Control Registers,			
ADC1_FILTCOEFF0_13	Filter Coefficients Measurement Unit Channel 0-13	60_H	see **Table 465**
ADC1_FILT_OUT0	ADC1 or Filter Output Channel 0	70_H	see **Table 466**
ADC1_FILT_OUT1	ADC1 or Filter Output Channel 1	74_H	see **Table 467**
ADC1_FILT_OUT2	ADC1 or Filter Output Channel 2	78_H	see **Table 468**
ADC1_FILT_OUT3	ADC1 or Filter Output Channel 3	$7C_H$	see **Table 469**
ADC1_FILT_OUT4	ADC1 or Filter Output Channel 4	80_H	see **Table 470**
ADC1_FILT_OUT5	ADC1 or Filter Output Channel 5	84_H	see **Table 471**
ADC1_FILT_OUT6	ADC1 or Filter Output Channel 6	88_H	see **Table 472**
ADC1_FILT_OUT7	ADC1 or Filter Output Channel 7	$8C_H$	see **Table 473**
ADC1_FILT_OUT8	ADC1 or Filter Output Channel 8	90_H	see **Table 474**
ADC1_FILT_OUT9	ADC1 or Filter Output Channel 9	94_H	see **Table 475**
ADC1_FILT_OUT10	ADC1 or Filter Output Channel 10	98_H	see **Table 476**
ADC1_FILT_OUT11	ADC1 or Filter Output Channel 11	$9C_H$	see **Table 477**
ADC1_DIFFCH_OUT1	ADC1 Differential Channel Output 1	$A0_H$	see **Table 480**
ADC1_FILT_OUT12	ADC1 or Filter Output Channel 12	110_H	see **Table 478**

Table 464 Register Overview (cont'd)

Register Short Name	Register Long Name	Offset Address	Reset Value
ADC1_FILT_OUTEIM	ADC1 or Filter Output of EIM	120_H	see Table 481
ADC1_FILT_OUT13	ADC1 or Filter Output Channel 13	140_H	see Table 479

The registers are addressed wordwise.

Filter Coefficients Measurement Unit 1 Channel 0-15

ADC1_FILTCOEFF0_13	Offset	Reset Value
Filter Coefficients Measurement Unit Channel 0-13	60_H	see Table 465

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RES				CH13		CH12		CH11		CH10		CH9		CH8	
r				rw		rw		rw		rw		rw		rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH7		CH6		CH5		CH4		CH3		CH2		CH1		CH0	
rw		rw		rw		rw		rw		rw		rw		rw	

Field	Bits	Type	Description
RES	31:28	r	**Reserved** Always read as 0
CH13	27:26	rw	**Filter Coefficients ADC channel 13** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
CH12	25:24	rw	**Filter Coefficients ADC channel 12** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
CH11	23:22	rw	**Filter Coefficients ADC channel 11** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample

Field	Bits	Type	Description
CH10	21:20	rw	**Filter Coefficients ADC channel 10** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
CH9	19:18	rw	**Filter Coefficients ADC channel 9** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
CH8	17:16	rw	**Filter Coefficients ADC channel 8** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
CH7	15:14	rw	**Filter Coefficients ADC channel 7** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
CH6	13:12	rw	**Filter Coefficients ADC channel 6** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
CH5	11:10	rw	**Filter Coefficients ADC channel 5** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
CH4	9:8	rw	**Filter Coefficients ADC channel 4** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
CH3	7:6	rw	**Filter Coefficients ADC channel 3** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
CH2	5:4	rw	**Filter Coefficients ADC channel 2** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample

Field	Bits	Type	Description
CH1	3:2	rw	**Filter Coefficients ADC channel 1** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample
CH0	1:0	rw	**Filter Coefficients ADC channel 0** 00_B **1/2**, weight of current sample 01_B **1/4**, weight of current sample 10_B **1/8**, weight of current sample 11_B **1/16**, weight of current sample

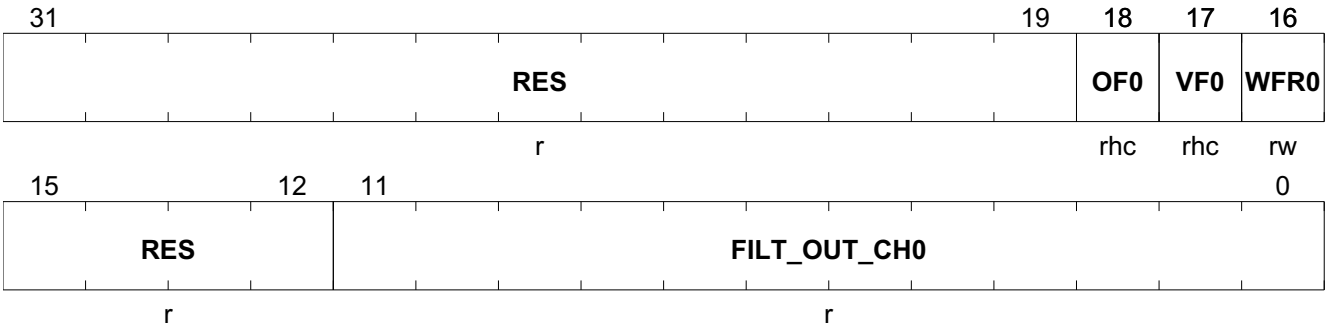
Table 465 RESET of ADC1_FILTCOEFF0_13

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0AAA\ AAAA_H$	RESET_TYPE_4		
TRIM_100_TP	$0AAA\ AAAA_H$	RESET		

ADC1 or Filter Output Channel 0

This registers reflects the current value of channel 0 of the measurement chain, which is assigned to Supply Voltage VS of the system.

ADC1_FILT_OUT0	Offset	Reset Value
ADC1 or Filter Output Channel 0	70_H	see Table 466



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
OF0	18	rhc	**Overrun Flag** Indicates if the result register is overwritten with new content (bit is set if VFx = 1 and new result is updated by hardware. *Note: Only set in WFRx = DISABLE and no software mode, clear on read of FILT_OUT_CH0 register* 0_B **NO OVERRUN** , Result register not overwritten 1_B **OVERRUN**, Result register overwritten
VF0	17	rhc	**Valid Flag** Indicates valid contents in result register bit field ADC1_OUT_CH0 *Note: Bit is set by hardware on update of result register and it is cleared by software once the FILT_OUT_CH0 register is read. The hardware update has higher priority than the software read in case the event occurs in the same cycle.* 0_B **NOT VALID**, No new valid data available 1_B **VALID**, Result register contains valid data and has not yet been read
WFR0	16	rw	**Wait for Read Mode** Enables wait for read mode for result register 0_B **DISABLE**, overwrite mode 1_B **ENABLE**, wait for read mode enabled

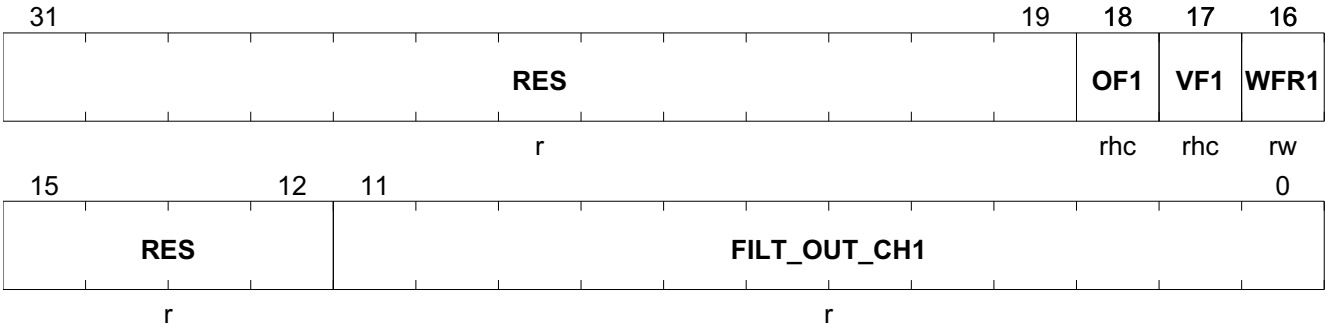
Field	Bits	Type	Description
RES	15:12	r	**Reserved** Always read as 0
FILT_OUT_CH0	11:0	r	**ADC or filter output value channel 0** For ADC output set ADC1_FILT_UPLO_CTRL.FUL_PP_CH0_EN = 0

Table 466 RESET of ADC1_FILT_OUT0

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 $0XXX_H$	RESET_TYPE_3		Exact reset value: 0000 0000 0000 0000 0000 XXXX XXXX XXXX(binary)

ADC1 or Filter Output Channel 1

ADC1_FILT_OUT1	Offset	Reset Value
ADC1 or Filter Output Channel 1	74_H	see Table 467



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
OF1	18	rhc	**Overrun Flag** Indicates if the result register is overwritten with new content (bit is set if VFx = 1 and new result is updated by hardware. *Note: Only set in WFRx = DISABLE and no software mode, clear on read of FILT_OUT_CH1 register* 0_B **NO OVERRUN** , Result register not overwritten 1_B **OVERRUN**, Result register overwritten
VF1	17	rhc	**Valid Flag** Indicates valid contents in result register bit field ADC1_OUT_CH1 *Note: Bit is set by hardware on update of result register and it is cleared by software once the FILT_OUT_CH1 register is read. The hardware update has higher priority than the software read in case the event occurs in the same cycle.* 0_B **NOT VALID**, No new valid data available 1_B **VALID**, Result register contains valid data and has not yet been read
WFR1	16	rw	**Wait for Read Mode** Enables wait for read mode for result register 0_B **DISABLE**, overwrite mode 1_B **ENABLE**, wait for read mode enabled
RES	15:12	r	**Reserved** Always read as 0

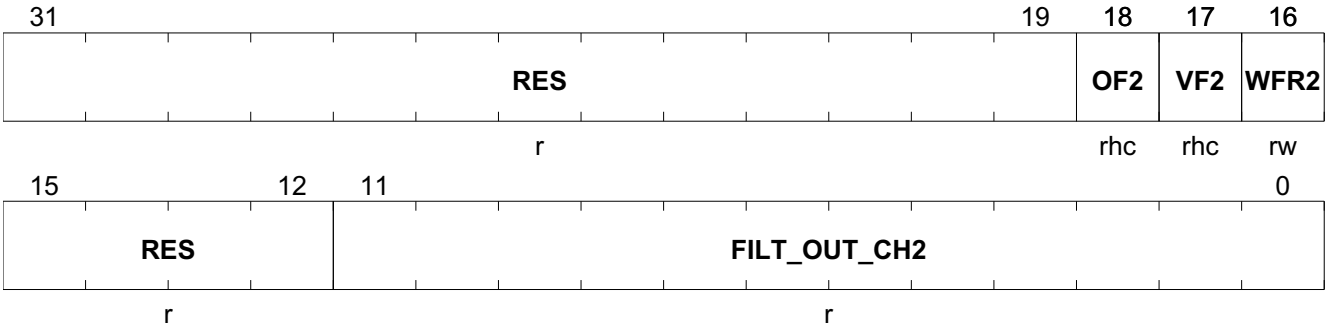
Field	Bits	Type	Description
FILT_OUT_CH1	11:0	r	**ADC or filter output value channel 1** For ADC output set ADC1_FILT_UPLO_CTRL.FUL_PP_CH1_EN = 0

Table 467 RESET of ADC1_FILT_OUT1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 $0XXX_H$	RESET_TYPE_3		Exact reset value: 0000 0000 0000 0000 0000 XXXX XXXX XXXX(binary)

ADC1 or Filter Output Channel 2

ADC1_FILT_OUT2	Offset	Reset Value
ADC1 or Filter Output Channel 2	78_H	see Table 468



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
OF2	18	rhc	**Overrun Flag** Indicates if the result register is overwritten with new content (bit is set if VFx = 1 and new result is updated by hardware. *Note: Only set in WFRx = DISABLE and no software mode, clear on read of FILT_OUT_CH2 register* 0_B **NO OVERRUN** , Result register not overwritten 1_B **OVERRUN**, Result register overwritten
VF2	17	rhc	**Valid Flag** Indicates valid contents in result register bit field ADC1_OUT_CH2 *Note: Bit is set by hardware on update of result register and it is cleared by software once the FILT_OUT_CH2 register is read. The hardware update has higher priority than the software read in case the event occurs in the same cycle.* 0_B **NOT VALID**, No new valid data available 1_B **VALID**, Result register contains valid data and has not yet been read
WFR2	16	rw	**Wait for Read Mode** Enables wait for read mode for result register 0_B **DISABLE**, overwrite mode 1_B **ENABLE**, wait for read mode enabled
RES	15:12	r	**Reserved** Always read as 0

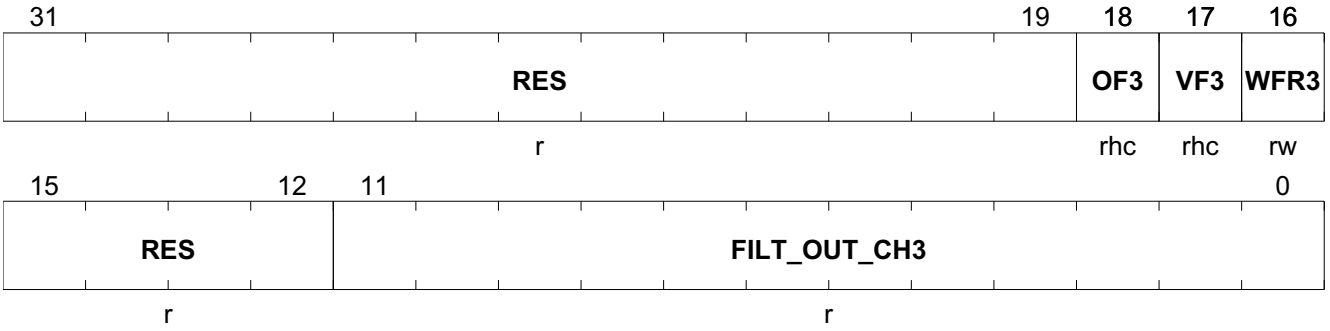
Field	Bits	Type	Description
FILT_OUT_CH2	11:0	r	**ADC or filter output value channel 2** For ADC output set ADC1_FILT_UPLO_CTRL.FUL_PP_CH2_EN = 0

Table 468 RESET of ADC1_FILT_OUT2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 $0XXX_H$	RESET_TYPE_3		Exact reset value: 0000 0000 0000 0000 0000 XXXX XXXX XXXX(binary)

ADC1 or Filter Output Channel 3

ADC1_FILT_OUT3	Offset	Reset Value
ADC1 or Filter Output Channel 3	$7C_H$	see Table 469



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
OF3	18	rhc	**Overrun Flag** Indicates if the result register is overwritten with new content (bit is set if VFx = 1 and new result is updated by hardware. *Note: Only set in WFRx = DISABLE and no software mode, clear on read of FILT_OUT_CH3 register* 0_B **NO OVERRUN** , Result register not overwritten 1_B **OVERRUN**, Result register overwritten
VF3	17	rhc	**Valid Flag** Indicates valid contents in result register bit field ADC1_OUT_CH3 *Note: Bit is set by hardware on update of result register and it is cleared by software once the FILT_OUT_CH3 register is read. The hardware update has higher priority than the software read in case the event occurs in the same cycle.* 0_B **NOT VALID**, No new valid data available 1_B **VALID**, Result register contains valid data and has not yet been read
WFR3	16	rw	**Wait for Read Mode** Enables wait for read mode for result register 0_B **DISABLE**, overwrite mode 1_B **ENABLE**, wait for read mode enabled
RES	15:12	r	**Reserved** Always read as 0

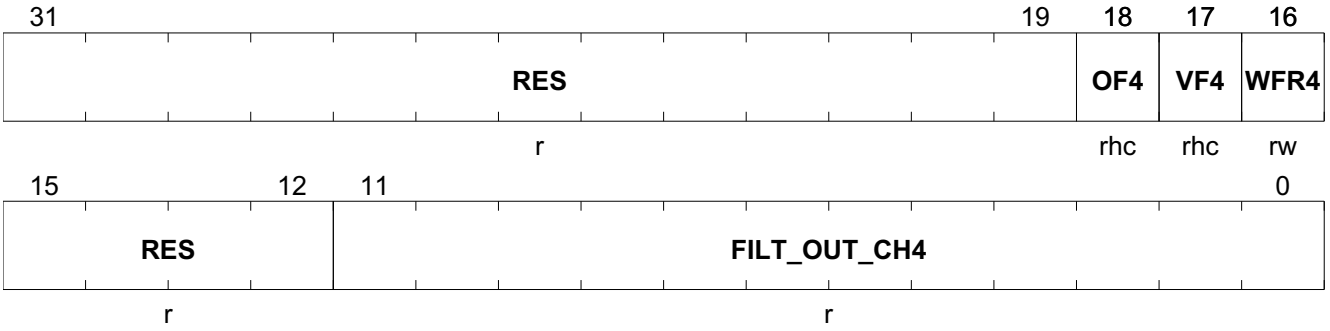
Field	Bits	Type	Description
FILT_OUT_CH3	11:0	r	**ADC or filter output value channel 3** For ADC output set ADC1_FILT_UPLO_CTRL.FUL_PP_CH3_EN = 0

Table 469 RESET of ADC1_FILT_OUT3

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 $0XXX_H$	RESET_TYPE_3		Exact reset value: 0000 0000 0000 0000 0000 XXXX XXXX XXXX(binary)

ADC1 or Filter Output Channel 4

ADC1_FILT_OUT4	Offset	Reset Value
ADC1 or Filter Output Channel 4	80_H	see Table 470



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
OF4	18	rhc	**Overrun Flag** Indicates if the result register is overwritten with new content (bit is set if VFx = 1 and new result is updated by hardware. *Note: Only set in WFRx = DISABLE and no software mode, clear on read of FILT_OUT_CH4 register* 0_B **NO OVERRUN** , Result register not overwritten 1_B **OVERRUN**, Result register overwritten
VF4	17	rhc	**Valid Flag** Indicates valid contents in result register bit field ADC1_OUT_CH4 *Note: Bit is set by hardware on update of result register and it is cleared by software once the FILT_OUT_CH4 register is read. The hardware update has higher priority than the software read in case the event occurs in the same cycle.* 0_B **NOT VALID**, No new valid data available 1_B **VALID**, Result register contains valid data and has not yet been read
WFR4	16	rw	**Wait for Read Mode** Enables wait for read mode for result register 0_B **DISABLE**, overwrite mode 1_B **ENABLE**, wait for read mode enabled
RES	15:12	r	**Reserved** Always read as 0

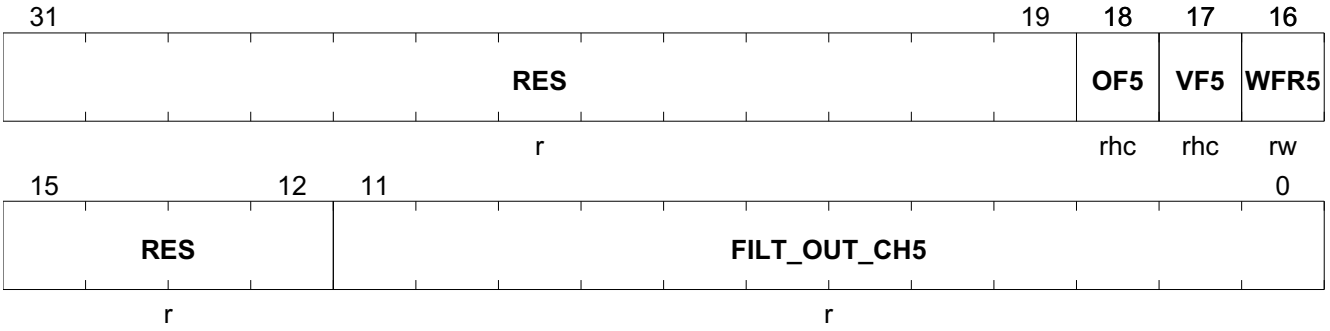
Field	Bits	Type	Description
FILT_OUT_CH4	11:0	r	**ADC or filter output value channel 4** For ADC output set ADC1_FILT_UPLO_CTRL.FUL_PP_CH4_EN = 0

Table 470 RESET of ADC1_FILT_OUT4

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 $0XXX_H$	RESET_TYPE_3		Exact reset value: 0000 0000 0000 0000 0000 XXXX XXXX XXXX(binary)

ADC1 or Filter Output Channel 5

ADC1_FILT_OUT5	Offset	Reset Value
ADC1 or Filter Output Channel 5	84_H	see Table 471



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
OF5	18	rhc	**Overrun Flag** Indicates if the result register is overwritten with new content (bit is set if VFx = 1 and new result is updated by hardware. *Note: Only set in WFRx = DISABLE and no software mode, clear on read of FILT_OUT_CH5 register* 0_B **NO OVERRUN** , Result register not overwritten 1_B **OVERRUN**, Result register overwritten
VF5	17	rhc	**Valid Flag** Indicates valid contents in result register bit field ADC1_OUT_CH5 *Note: Bit is set by hardware on update of result register and it is cleared by software once the FILT_OUT_CH5 register is read. The hardware update has higher priority than the software read in case the event occurs in the same cycle.* 0_B **NOT VALID**, No new valid data available 1_B **VALID**, Result register contains valid data and has not yet been read
WFR5	16	rw	**Wait for Read Mode** Enables wait for read mode for result register 0_B **DISABLE**, overwrite mode 1_B **ENABLE**, wait for read mode enabled
RES	15:12	r	**Reserved** Always read as 0

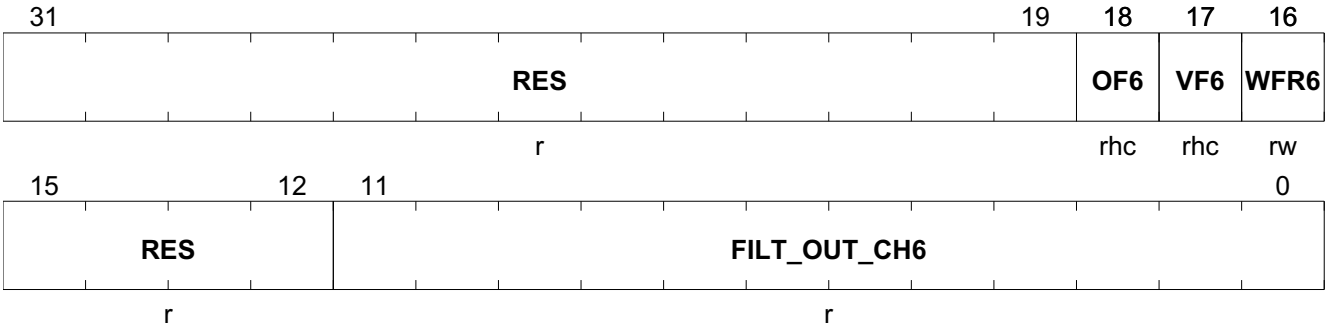
Field	Bits	Type	Description
FILT_OUT_CH5	11:0	r	**ADC or filter output value channel 5** For ADC output set ADC1_FILT_UPLO_CTRL.FUL_PP_CH5_EN = 0

Table 471 RESET of ADC1_FILT_OUT5

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 $0XXX_H$	RESET_TYPE_3		Exact reset value: 0000 0000 0000 0000 0000 XXXX XXXX XXXX(binary)

ADC1 or Filter Output Channel 6

ADC1_FILT_OUT6	Offset	Reset Value
ADC1 or Filter Output Channel 6	88_H	see Table 472



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
OF6	18	rhc	**Overrun Flag** Indicates if the result register is overwritten with new content (bit is set if VFx = 1 and new result is updated by hardware. *Note: Only set in WFRx = DISABLE and no software mode, clear on read of FILT_OUT_CH6 register* 0_B **NO OVERRUN** , Result register not overwritten 1_B **OVERRUN**, Result register overwritten
VF6	17	rhc	**Valid Flag** Indicates valid contents in result register bit field ADC1_OUT_CH6 *Note: Bit is set by hardware on update of result register and it is cleared by software once the FILT_OUT_CH6 register is read. The hardware update has higher priority than the software read in case the event occurs in the same cycle.* 0_B **NOT VALID**, No new valid data available 1_B **VALID**, Result register contains valid data and has not yet been read
WFR6	16	rw	**Wait for Read Mode** Enables wait for read mode for result register 0_B **DISABLE**, overwrite mode 1_B **ENABLE**, wait for read mode enabled
RES	15:12	r	**Reserved** Always read as 0

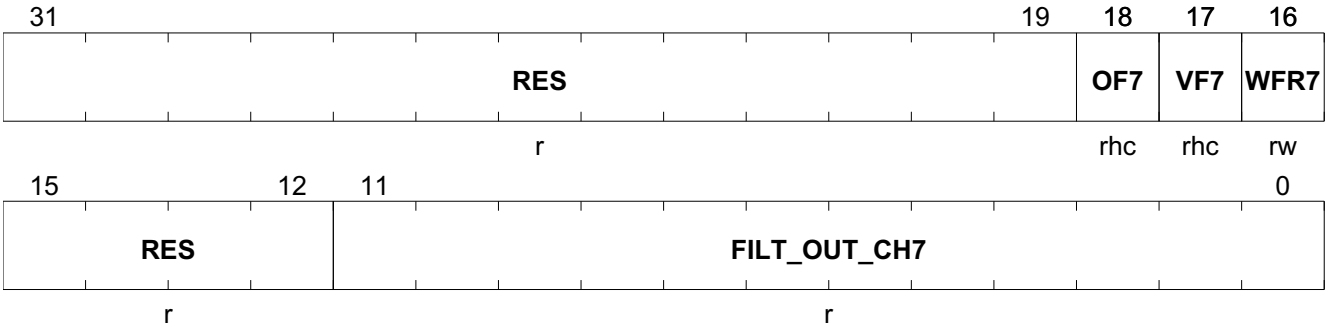
Field	Bits	Type	Description
FILT_OUT_CH6	11:0	r	**ADC or filter output value channel 6** For ADC output set ADC1_FILT_UPLO_CTRL.FUL_PP_CH6_EN = 0

Table 472 RESET of ADC1_FILT_OUT6

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 $0XXX_H$	RESET_TYPE_3		Exact reset value: 0000 0000 0000 0000 0000 XXXX XXXX XXXX(binary)

ADC1 or Filter Output Channel 7

ADC1_FILT_OUT7	Offset	Reset Value
ADC1 or Filter Output Channel 7	$8C_H$	see Table 473



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
OF7	18	rhc	**Overrun Flag** Indicates if the result register is overwritten with new content (bit is set if VFx = 1 and new result is updated by hardware. *Note: Only set in WFRx = DISABLE and no software mode, clear on read of FILT_OUT_CH7 register* 0_B **NO OVERRUN** , Result register not overwritten 1_B **OVERRUN**, Result register overwritten
VF7	17	rhc	**Valid Flag** Indicates valid contents in result register bit field ADC1_OUT_CH7 *Note: Bit is set by hardware on update of result register and it is cleared by software once the FILT_OUT_CH7 register is read. The hardware update has higher priority than the software read in case the event occurs in the same cycle.* 0_B **NOT VALID**, No new valid data available 1_B **VALID**, Result register contains valid data and has not yet been read
WFR7	16	rw	**Wait for Read Mode** Enables wait for read mode for result register 0_B **DISABLE**, overwrite mode 1_B **ENABLE**, wait for read mode enabled
RES	15:12	r	**Reserved** Always read as 0

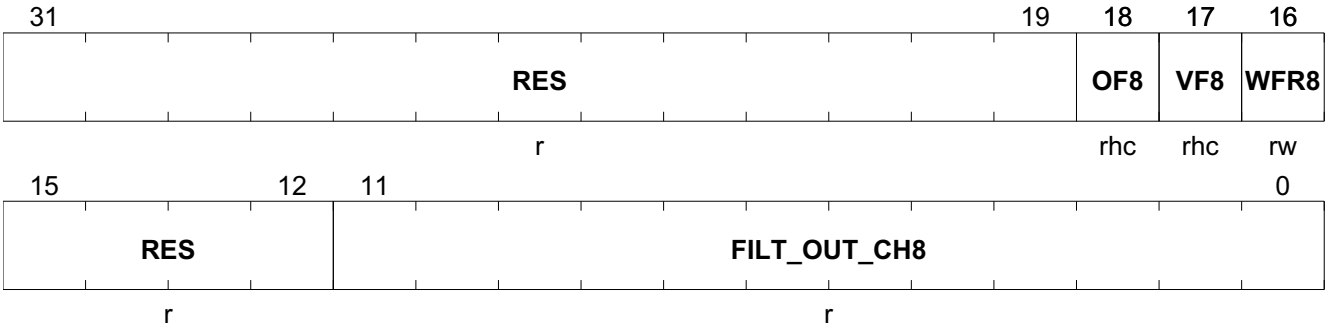
Field	Bits	Type	Description
FILT_OUT_CH7	11:0	r	**ADC or filter output value channel 7** For ADC output set ADC1_FILT_UPLO_CTRL.FUL_PP_CH7_EN = 0

Table 473 RESET of ADC1_FILT_OUT7

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 $0XXX_H$	RESET_TYPE_3		Exact reset value: 0000 0000 0000 0000 0000 XXXX XXXX XXXX(binary)

ADC1 or Filter Output Channel 8

ADC1_FILT_OUT8	Offset	Reset Value
ADC1 or Filter Output Channel 8	90_H	see Table 474



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
OF8	18	rhc	**Overrun Flag** Indicates if the result register is overwritten with new content (bit is set if VFx = 1 and new result is updated by hardware. *Note: Only set in WFRx = DISABLE and no software mode, clear on read of FILT_OUT_CH8 register* 0_B **NO OVERRUN** , Result register not overwritten 1_B **OVERRUN**, Result register overwritten
VF8	17	rhc	**Valid Flag** Indicates valid contents in result register bit field ADC1_OUT_CH8 *Note: Bit is set by hardware on update of result register and it is cleared by software once the FILT_OUT_CH8 register is read. The hardware update has higher priority than the software read in case the event occurs in the same cycle.* 0_B **NOT VALID**, No new valid data available 1_B **VALID**, Result register contains valid data and has not yet been read
WFR8	16	rw	**Wait for Read Mode** Enables wait for read mode for result register 0_B **DISABLE**, overwrite mode 1_B **ENABLE**, wait for read mode enabled
RES	15:12	r	**Reserved** Always read as 0

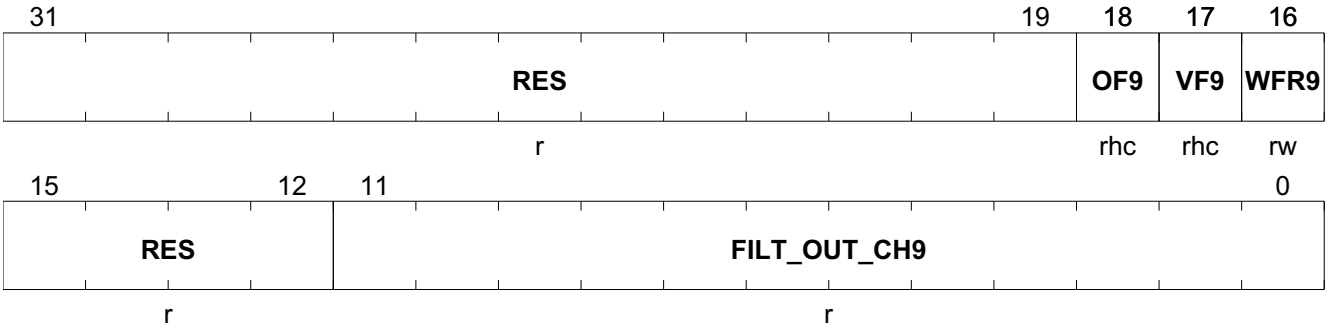
Field	Bits	Type	Description
FILT_OUT_CH8	11:0	r	**ADC or filter output value channel 8** For ADC output set ADC1_FILT_UPLO_CTRL.FUL_PP_CH8_EN = 0

Table 474 RESET of ADC1_FILT_OUT8

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 $0XXX_H$	RESET_TYPE_3		Exact reset value: 0000 0000 0000 0000 0000 XXXX XXXX XXXX(binary)

ADC1 or Filter Output Channel 9

ADC1_FILT_OUT9	Offset	Reset Value
ADC1 or Filter Output Channel 9	94_H	see Table 475



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
OF9	18	rhc	**Overrun Flag** Indicates if the result register is overwritten with new content (bit is set if VFx = 1 and new result is updated by hardware. *Note: Only set in WFRx = DISABLE and no software mode, clear on read of FILT_OUT_CH9 register* 0_B **NO OVERRUN** , Result register not overwritten 1_B **OVERRUN**, Result register overwritten
VF9	17	rhc	**Valid Flag** Indicates valid contents in result register bit field ADC1_OUT_CH9 *Note: Bit is set by hardware on update of result register and it is cleared by software once the FILT_OUT_CH9 register is read. The hardware update has higher priority than the software read in case the event occurs in the same cycle.* 0_B **NOT VALID**, No new valid data available 1_B **VALID**, Result register contains valid data and has not yet been read
WFR9	16	rw	**Wait for Read Mode** Enables wait for read mode for result register 0_B **DISABLE**, overwrite mode 1_B **ENABLE**, wait for read mode enabled
RES	15:12	r	**Reserved** Always read as 0

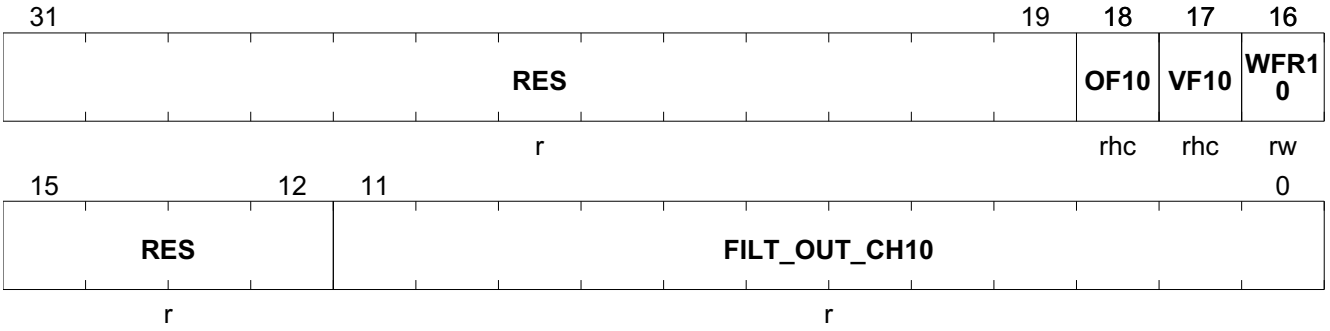
Field	Bits	Type	Description
FILT_OUT_CH9	11:0	r	**ADC or filter output value channel 9** For ADC output set ADC1_FILT_UPLO_CTRL.FUL_PP_CH9_EN = 0

Table 475 RESET of ADC1_FILT_OUT9

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 $0XXX_H$	RESET_TYPE_3		Exact reset value: 0000 0000 0000 0000 0000 XXXX XXXX XXXX(binary)

ADC1 or Filter Output Channel 10

ADC1_FILT_OUT10	Offset	Reset Value
ADC1 or Filter Output Channel 10	98_H	see Table 476



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
OF10	18	rhc	**Overrun Flag** Indicates if the result register is overwritten with new content (bit is set if VFx = 1 and new result is updated by hardware. *Note: Only set in WFRx = DISABLE and no software mode, clear on read of FILT_OUT_CH10 register* 0_B **NO OVERRUN** , Result register not overwritten 1_B **OVERRUN**, Result register overwritten
VF10	17	rhc	**Valid Flag** Indicates valid contents in result register bit field ADC1_OUT_CH10 *Note: Bit is set by hardware on update of result register and it is cleared by software once the FILT_OUT_CH10 register is read. The hardware update has higher priority than the software read in case the event occurs in the same cycle.* 0_B **NOT VALID**, No new valid data available 1_B **VALID**, Result register contains valid data and has not yet been read
WFR10	16	rw	**Wait for Read Mode** Enables wait for read mode for result register 0_B **DISABLE**, overwrite mode 1_B **ENABLE**, wait for read mode enabled
RES	15:12	r	**Reserved** Always read as 0

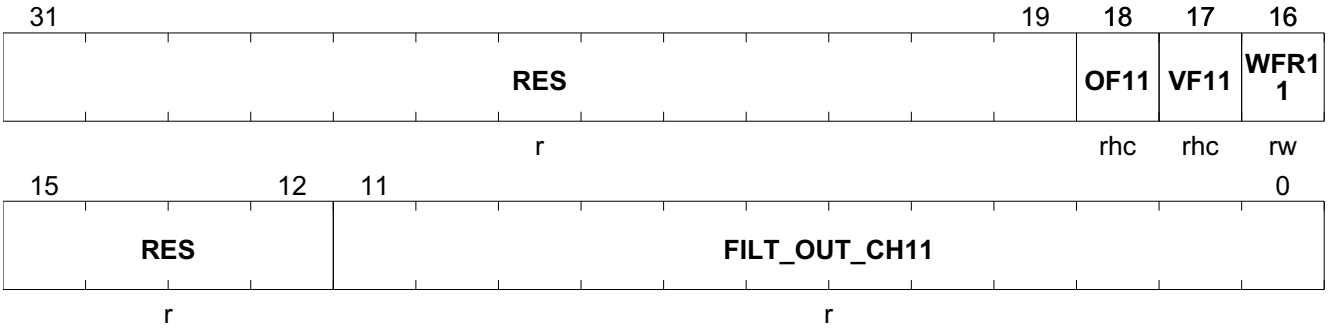
Field	Bits	Type	Description
FILT_OUT_CH10	11:0	r	**ADC or filter output value channel 10** For ADC output set ADC1_FILT_UPLO_CTRL.FUL_PP_CH10_EN = 0

Table 476 RESET of ADC1_FILT_OUT10

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 $0XXX_H$	RESET_TYPE_3		Exact reset value: 0000 0000 0000 0000 0000 XXXX XXXX XXXX(binary)

ADC1 or Filter Output Channel 11

ADC1_FILT_OUT11	Offset	Reset Value
ADC1 or Filter Output Channel 11	$9C_H$	see Table 477



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
OF11	18	rhc	**Overrun Flag** Indicates if the result register is overwritten with new content (bit is set if VFx = 1 and new result is updated by hardware. *Note: Only set in WFRx = DISABLE and no software mode, clear on read of FILT_OUT_CH11 register* 0_B **NO OVERRUN** , Result register not overwritten 1_B **OVERRUN**, Result register overwritten
VF11	17	rhc	**Valid Flag** Indicates valid contents in result register bit field ADC1_OUT_CH11 *Note: Bit is set by hardware on update of result register and it is cleared by software once the FILT_OUT_CH11 register is read. The hardware update has higher priority than the software read in case the event occurs in the same cycle.* 0_B **NOT VALID**, No new valid data available 1_B **VALID**, Result register contains valid data and has not yet been read
WFR11	16	rw	**Wait for Read Mode** Enables wait for read mode for result register 0_B **DISABLE**, overwrite mode 1_B **ENABLE**, wait for read mode enabled
RES	15:12	r	**Reserved** Always read as 0

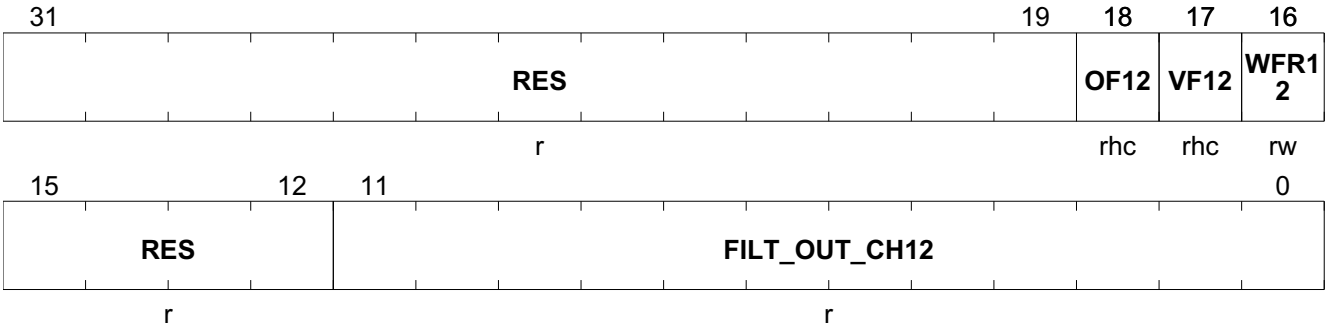
Field	Bits	Type	Description
FILT_OUT_CH11	11:0	r	**ADC or filter output value channel 11** For ADC output set ADC1_FILT_UPLO_CTRL.FUL_PP_CH11_EN = 0

Table 477 RESET of ADC1_FILT_OUT11

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 $0XXX_H$	RESET_TYPE_3		Exact reset value: 0000 0000 0000 0000 0000 XXXX XXXX XXXX(binary)

ADC1 or Filter Output Channel 12

ADC1_FILT_OUT12	Offset	Reset Value
ADC1 or Filter Output Channel 12	110_H	see Table 478



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
OF12	18	rhc	**Overrun Flag** Indicates if the result register is overwritten with new content (bit is set if VFx = 1 and new result is updated by hardware. *Note: Only set in WFRx = DISABLE and no software mode, clear on read of FILT_OUT_CH12 register* 0_B **NO OVERRUN** , Result register not overwritten 1_B **OVERRUN**, Result register overwritten
VF12	17	rhc	**Valid Flag** Indicates valid contents in result register bit field ADC1_OUT_CH12 *Note: Bit is set by hardware on update of result register and it is cleared by software once the FILT_OUT_CH12 register is read. The hardware update has higher priority than the software read in case the event occurs in the same cycle.* 0_B **NOT VALID**, No new valid data available 1_B **VALID**, Result register contains valid data and has not yet been read
WFR12	16	rw	**Wait for Read Mode** Enables wait for read mode for result register 0_B **DISABLE**, overwrite mode 1_B **ENABLE**, wait for read mode enabled
RES	15:12	r	**Reserved** Always read as 0

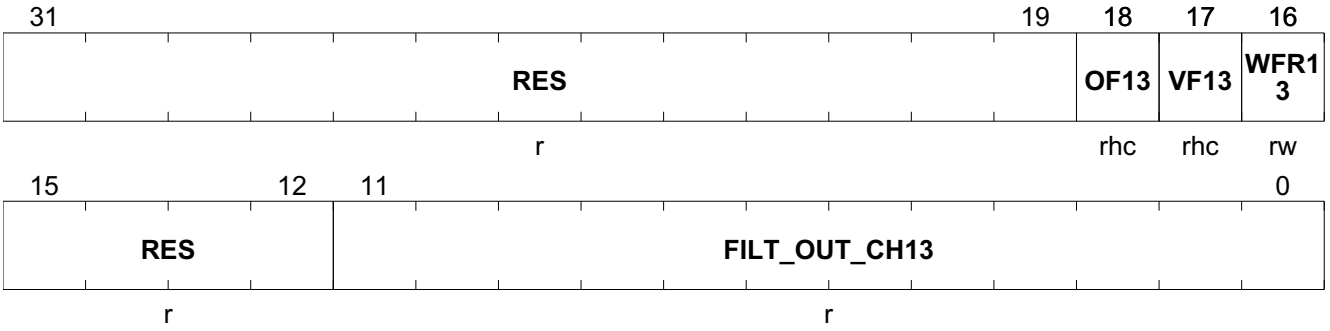
Field	Bits	Type	Description
FILT_OUT_CH12	11:0	r	**ADC or filter output value channel 12** For ADC output set ADC1_FILT_UPLO_CTRL.FUL_PP_CH12_EN = 0

Table 478 RESET of ADC1_FILT_OUT12

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 $0XXX_H$	RESET_TYPE_3		Exact reset value: 0000 0000 0000 0000 0000 XXXX XXXX XXXX(binary)

ADC1 or Filter Output Channel 13

ADC1_FILT_OUT13	Offset	Reset Value
ADC1 or Filter Output Channel 13	140_H	see Table 479



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
OF13	18	rhc	**Overrun Flag** Indicates if the result register is overwritten with new content (bit is set if VFx = 1 and new result is updated by hardware. *Note: Only set in WFRx = DISABLE and no software mode, clear on read of FILT_OUT_CH13 register* 0_B **NO OVERRUN** , Result register not overwritten 1_B **OVERRUN**, Result register overwritten
VF13	17	rhc	**Valid Flag** Indicates valid contents in result register bit field ADC1_OUT_CH13 *Note: Bit is set by hardware on update of result register and it is cleared by software once the FILT_OUT_CH13 register is read. The hardware update has higher priority than the software read in case the event occurs in the same cycle.* 0_B **NOT VALID**, No new valid data available 1_B **VALID**, Result register contains valid data and has not yet been read
WFR13	16	rw	**Wait for Read Mode** Enables wait for read mode for result register 0_B **DISABLE**, overwrite mode 1_B **ENABLE**, wait for read mode enabled
RES	15:12	r	**Reserved** Always read as 0

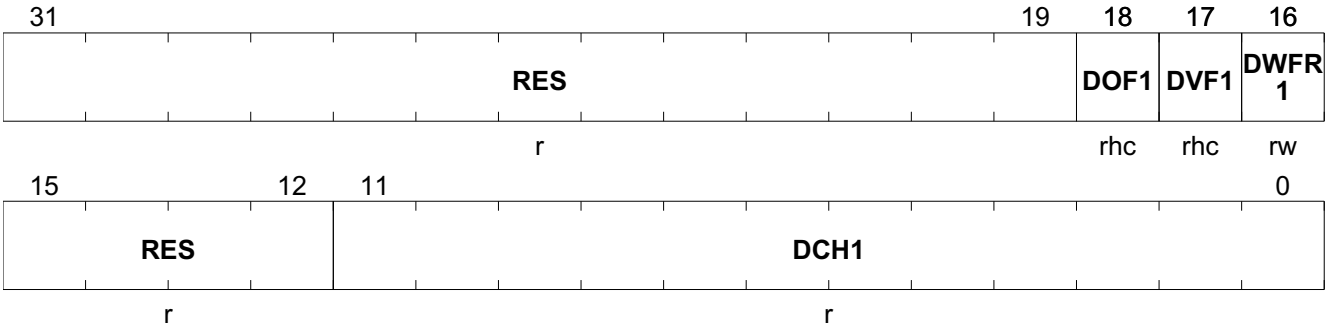
Field	Bits	Type	Description
FILT_OUT_CH13	11:0	r	**ADC or filter output value channel 13** For ADC output set ADC1_FILT_UPLO_CTRL.FUL_PP_CH13_EN = 0

Table 479 RESET of ADC1_FILT_OUT13

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 $0XXX_H$	RESET_TYPE_3		Exact reset value: 0000 0000 0000 0000 0000 XXXX XXXX XXXX(binary)

ADC1 Differential Channel Output 1

ADC1_DIFFCH_OUT1	Offset	Reset Value
ADC1 Differential Channel Output 1	$A0_H$	see Table 480



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
DOF1	18	rhc	**Overrun Flag** Indicates if the result register is overwritten with new content (bit is set if VFx = 1 and new result is updated by hardware. *Note: Only set in WFRx = DISABLE and no software mode, clear on read of DCH1 register* 0_B **NO OVERRUN** , Result register not overwritten 1_B **OVERRUN**, Result register overwritten
DVF1	17	rhc	**Valid Flag** Indicates valid contents in result register bit field ADC1_DOUT1 *Note: Bit is set by hardware on update of result register and it is cleared by software once the DCH1 register is read. The hardware update has higher priority than the software read in case the event occurs in the same cycle.* 0_B **NOT VALID**, No new valid data available 1_B **VALID**, Result register contains valid data and has not yet been read
DWFR1	16	rw	**Wait for Read Mode** Enables wait for read mode for result register 0_B **DISABLE**, overwrite mode 1_B **ENABLE**, wait for read mode enabled
RES	15:12	r	**Reserved** Always read as 0

Field	Bits	Type	Description
DCH1	11:0	r	**ADC differential output value 1**

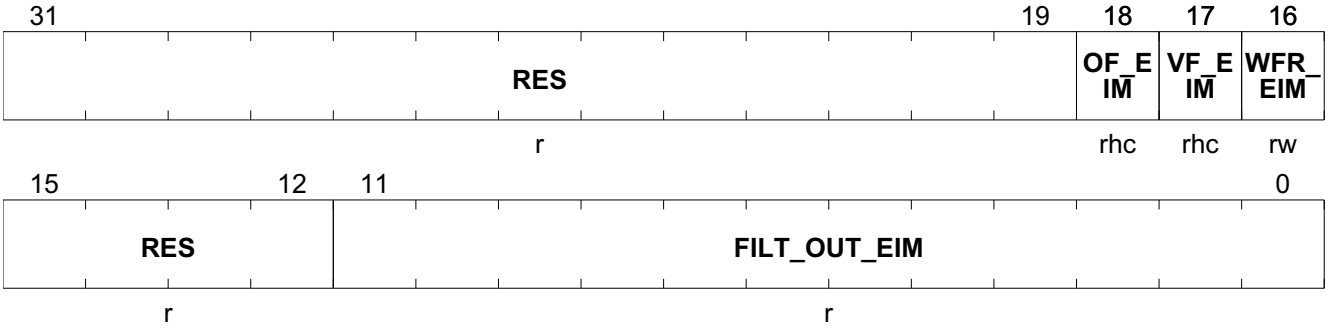
Table 480 RESET of ADC1_DIFFCH_OUT1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 $0XXX_H$	RESET_TYPE_3		Exact reset value: 0000 0000 0000 0000 0000 XXXX XXXX XXXX(binary)

ADC1 or Filter Output for EIM Measurement

Note: This Channel is not included in the sequencer. EIM Mode uses the postprocessing chain of the selected EIM channel.

ADC1_FILT_OUTEIM	Offset	Reset Value
ADC1 or Filter Output of EIM	120_H	see Table 481



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
OF_EIM	18	rhc	**Overrun Flag** Indicates if the result register is overwritten with new content (bit is set if VFx = 1 and new result is updated by hardware. *Note: Only set in WFRx = DISABLE and no software mode, clear on read of FILT_OUT_EIM register* 0_B **NO OVERRUN** , Result register not overwritten 1_B **OVERRUN**, Result register overwritten
VF_EIM	17	rhc	**Valid Flag** Indicates valid contents in result register bit field of last EIM Measurement *Note: Bit is set by hardware on update of result register and it is cleared by software once the FILT_OUT_EIM register is read. The hardware update has higher priority than the software read in case the event occurs in the same cycle.* 0_B **NOT VALID**, No new valid data available 1_B **VALID**, Result register contains valid data and has not yet been read

Field	Bits	Type	Description
WFR_EIM	16	rw	**Wait for Read Mode** Enables wait for read mode for result register 0_B **DISABLE**, overwrite mode 1_B **ENABLE**, wait for read mode enabled
RES	15:12	r	**Reserved** Always read as 0
FILT_OUT_EIM	11:0	r	**ADC or filter output value for last EIM measurement**

Table 481 RESET of ADC1_FILT_OUTEIM

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 $0XXX_H$	RESET_TYPE_3		Exact reset value: 0000 0000 0000 0000 0000 XXXX XXXX XXXX(binary)

24.8 Signal Processing

24.8.1 Functional Description

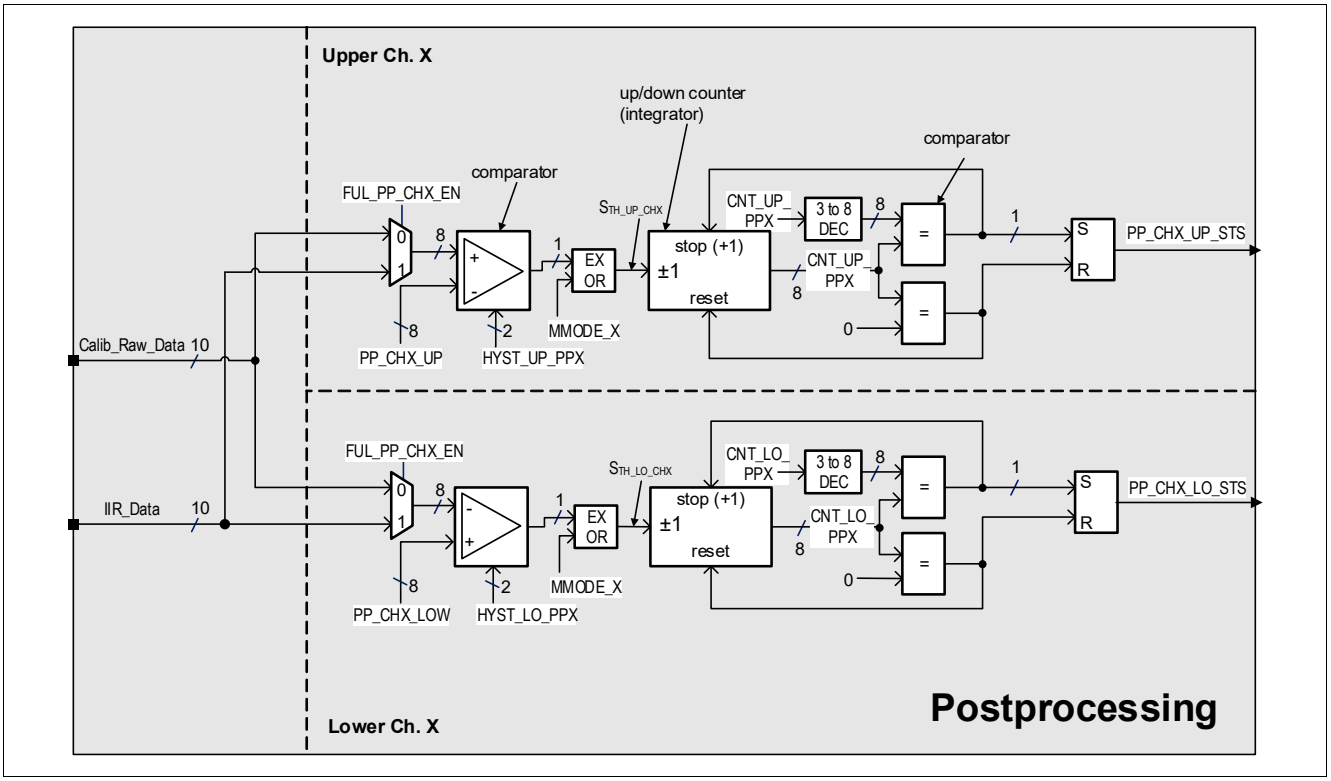


Figure 202 Postprocessing channel block diagram for voltage measurements

As shown in **Figure 202** an adjustable filter can be applied for the upper and the lower measurement channel, which averages 2, 4, 8 or 16 measurement values continuously. The 8 MSBs of the filtered signal or the demultiplexed ADC output signal FILT_OUT_CHX are compared with an upper threshold PP_CHX_UP and a lower threshold PP_CHX_LOW. When the thresholds are exceeded, the comparator outputs get active. For all measurement modes a freely adjustable hysteresis can be defined which is defined with the HYST_UP_PPX and HYST_LO_PPX values.

In addition to the first filter stage, the second filters (counters) integrate the comparator output values S_{TH_UP/LO_CHX} until an individual upper and lower timing threshold $2^{CNT\_UP/LO\_PPX}$ is reached. When reaching the upper timing threshold $2^{CNT\_UP\_PPX}$, the upper counter increment is stalled and the status output PP_CHX_UP_STS is set. For MMODE_X = MMODEOV, the inverted lower comparator output signal $S_{TH\_LO\_CHX}$ is normalized again. When the output signal is above PP_CHX_LOW, the lower counter is incremented until the max. threshold $2^{CNT\_LO\_PPX}$ is reached. Individual interrupts for the upper and lower channel can be triggered with the rising edge of the status signals PP_CHX_UP/LO_STS.

In general the IIR filter stage suppresses higher frequency noise efficiently and triggering with the upper and lower threshold PP_CHX_UP/LOW are dependent on the measured values. Hence short high-level spikes might pass the thresholds. In opposite to the first stage the nature of the second filter stage is more a time filter, which is less dependent on the measurement values but on event durations of S_{TH_LO/UP_CHX} as generated by the first comparator stage. Therefore the second stage has a lower noise suppression performance for higher frequencies and also adds a delay for the trigger time proportional to $2^{CNT\_UP/LO\_PPX}$.

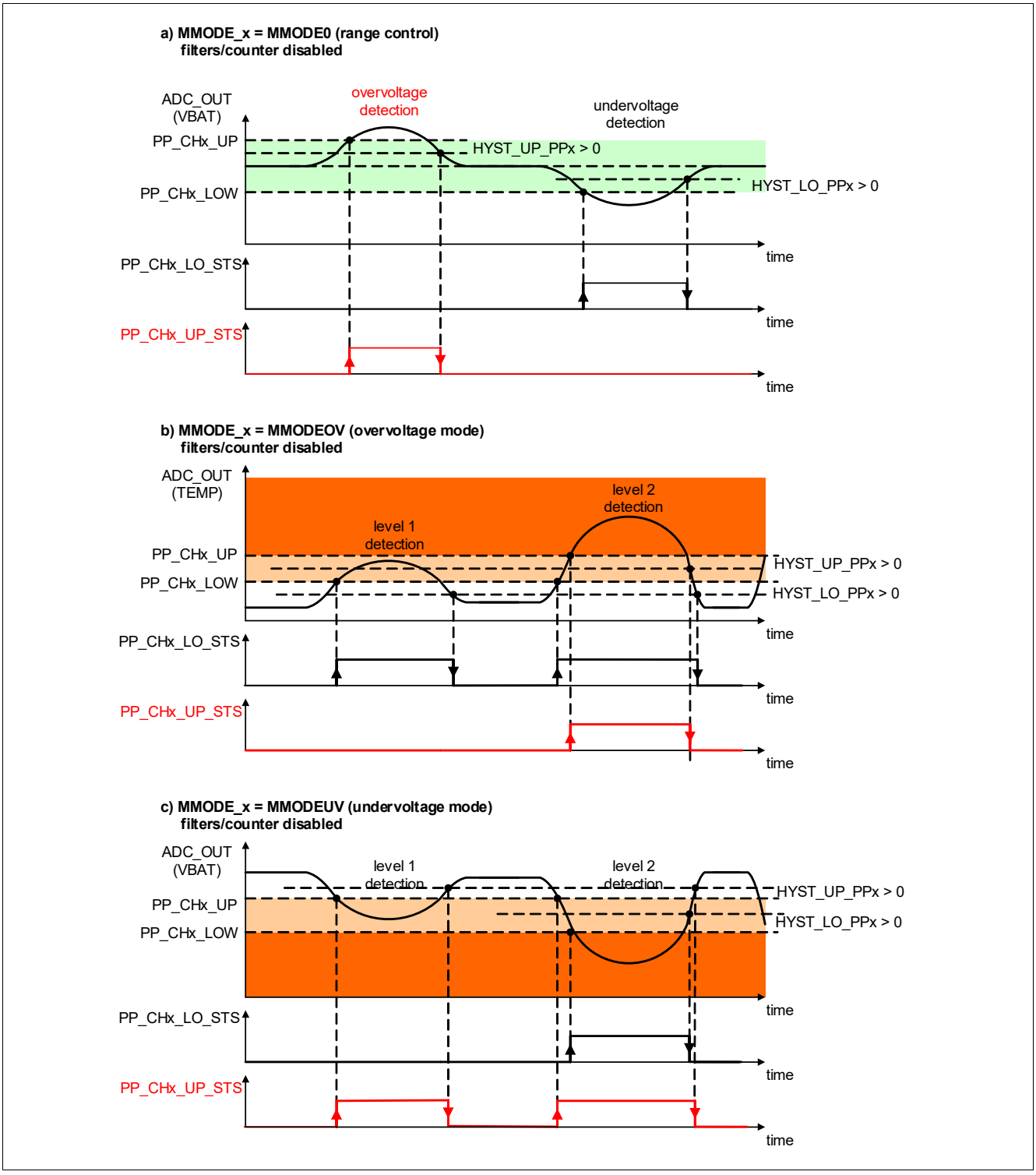


Figure 203 Measurement examples of a measurement channel with disabled filters

Figure 203 shows three examples, an over- and undervoltage detection (e.g. VBAT_SENSE monitoring), a 2-step overvoltage and a 2-step undervoltage detection. The modes MMODEOV/UV can be used as prewarning for the application software (e.g. close to supply undervoltage).

24.8.2 Postprocessing Control Registers

The Postprocessing block is fully controllable by the below listed sfr Registers.

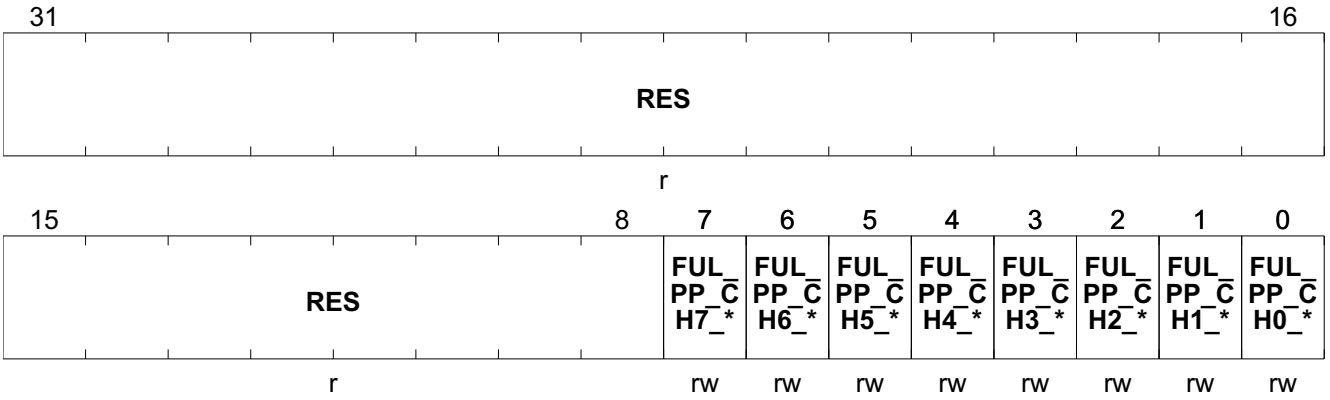
Table 482 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Postprocessing Control Registers,			
ADC1_TH0_3_LOWER	Lower Comparator Trigger Level Post-Processing-Channel 0-3	40_H	see Table 491
ADC1_TH4_7_LOWER	Lower Comparator Trigger Level Post-Processing-Channel 4-7	44_H	see Table 492
ADC1_FILT_UPLO_CTRL	Upper And Lower Threshold Filter Enable	$B0_H$	see Table 483
ADC1_DCHTH1_4_LOWER	Lower Comparator Trigger Level Differential Channel 1	$C4_H$	see Table 493
ADC1_TH0_3_UPPER	Upper Comparator Trigger Level Post-Processing-Channel 0-3	$C8_H$	see Table 485
ADC1_TH4_7_UPPER	Upper Comparator Trigger Level Post-Processing-Channel 4-7	CC_H	see Table 486
ADC1_DCHTH1_4_UPPER	Upper Comparator Trigger Level Differential Channel 1	$D4_H$	see Table 487
ADC1_CNT0_3_LOWER	Lower Counter Trigger Level Post-Processing-Channel 0-3	$D8_H$	see Table 494
ADC1_CNT4_7_LOWER	Lower Counter Trigger Level Post-Processing-Channel 4-7	DC_H	see Table 495
ADC1_DCHCNT1_4_LOWER	Lower Counter Trigger Level Differential Channel 1	$E4_H$	see Table 496
ADC1_CNT0_3_UPPER	Upper Counter Trigger Level Post-Processing-Channel 0-3	$E8_H$	see Table 488
ADC1_CNT4_7_UPPER	Upper Counter Trigger Level Post-Processing-Channel 4-7	EC_H	see Table 489
ADC1_DCHCNT1_4_UPPER	Upper Counter Trigger Level Differential Channel 1	$F4_H$	see Table 490
ADC1_MMODE0_7	Overvoltage Measurement Mode of Post-Processing-Channel 0-7	$F8_H$	see Table 484
ADC1_PP_MAP0_3	Post-Processing Mapping Channel 0-3	118_H	see Table 497
ADC1_PP_MAP4_7	Post-Processing Mapping Channel 4-7	$11C_H$	see Table 497

The registers are addressed wordwise.

Upper And Lower Threshold Filter Enable

ADC1_FILT_UPLO_CTRL	Offset	Reset Value
Upper And Lower Threshold Filter Enable	$B0_H$	see Table 483



Field	Bits	Type	Description
RES	31:8	r	**Reserved** Always read as 0
FUL_PP_CH7_EN	7	rw	**Upper and lower threshold IIR filter enable Post-Processing-Channel 7** 0_B **Disable**, 1_B **Enable**,
FUL_PP_CH6_EN	6	rw	**Upper and lower threshold IIR filter enable Post-Processing-Channel 6** 0_B **Disable**, 1_B **Enable**,
FUL_PP_CH5_EN	5	rw	**Upper and lower threshold IIR filter enable Post-Processing-Channel 5** 0_B **Disable**, 1_B **Enable**,
FUL_PP_CH4_EN	4	rw	**Upper and lower threshold IIR filter enable Post-Processing-Channel 4** 0_B **Disable**, 1_B **Enable**,
FUL_PP_CH3_EN	3	rw	**Upper and lower threshold IIR filter enable Post-Processing-Channel 3** 0_B **Disable**, 1_B **Enable**,
FUL_PP_CH2_EN	2	rw	**Upper and lower threshold IIR filter enable Post-Processing-Channel 2** 0_B **Disable**, 1_B **Enable**,

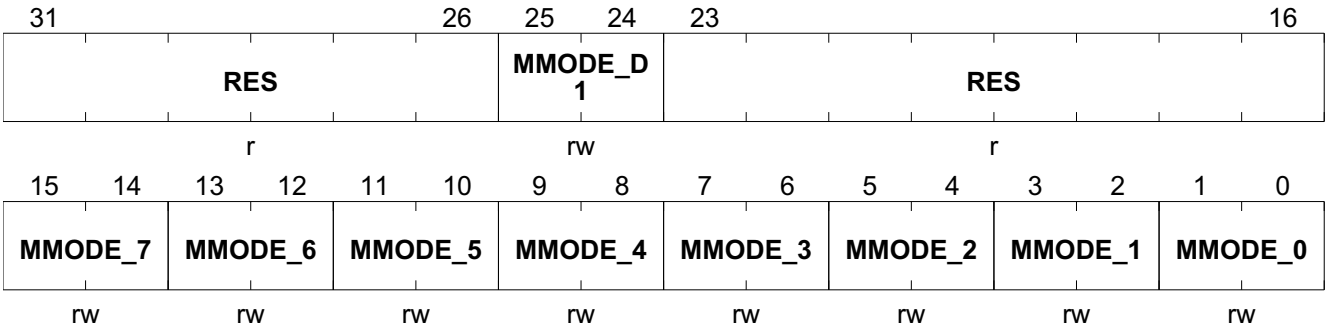
Field	Bits	Type	Description
FUL_PP_CH1_EN	1	rw	**Upper and lower threshold IIR filter enable Post-Processing-Channel 1** 0_B **Disable**, 1_B **Enable**,
FUL_PP_CH0_EN	0	rw	**Upper and lower threshold IIR filter enable Post-Processing-Channel 0** 0_B **Disable**, 1_B **Enable**,

Table 483 RESET of ADC1_FILT_UPLO_CTRL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	0000 00FF$_H$	RESET_TYPE_4		
TRIM_100_TP	000000FF$_H$	RESET		

Overvoltage Measurement Mode of Post-Processing-Channel 0-7

ADC1_MMODE0_7	Offset	Reset Value
Overvoltage Measurement Mode of Post-Processing-Channel 0-7	**F8$_H$**	**see Table 484**



Field	Bits	Type	Description
RES	31:26	r	**Reserved** Always read as 0
MMODE_D1	25:24	rw	**Measurement mode Differential Channel 1** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved
RES	23:16	r	**Reserved** Always read as 0

Field	Bits	Type	Description
MMODE_7	15:14	rw	**Measurement mode Post-Processing-Channel 7** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved
MMODE_6	13:12	rw	**Measurement mode Post-Processing-Channel 6** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved
MMODE_5	11:10	rw	**Measurement mode Post-Processing-Channel 5** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved
MMODE_4	9:8	rw	**Measurement mode Post-Processing-Channel 4** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved
MMODE_3	7:6	rw	**Measurement mode Post-Processing-Channel 3** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved
MMODE_2	5:4	rw	**Measurement mode Post-Processing-Channel 2** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved
MMODE_1	3:2	rw	**Measurement mode Post-Processing-Channel 1** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved

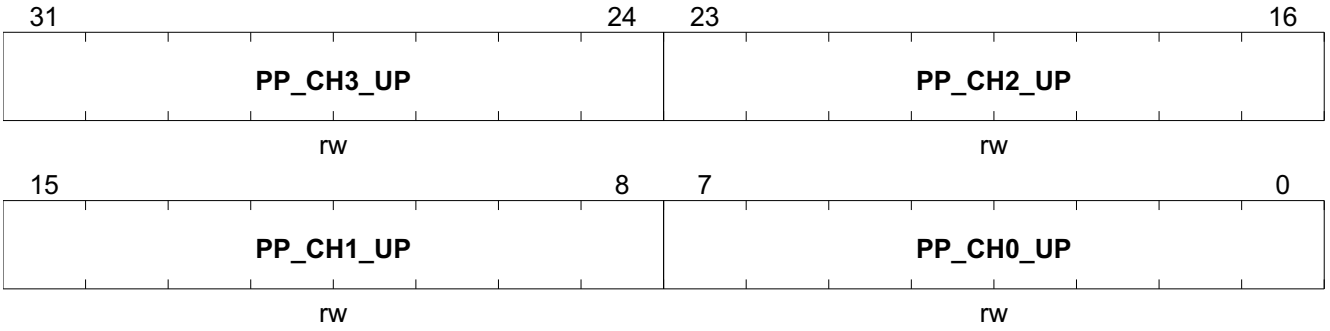
Field	Bits	Type	Description
MMODE_0	1:0	rw	**Measurement mode Post-Processing-Channel 0** 00_B **MMODE0**, upper and lower voltage/limit measurement 01_B **MMODEUV**, undervoltage/-limit measurement 10_B **MMODEOV**, overvoltage/-limit measurement 11_B **RESERVED**, reserved

Table 484 RESET of ADC1_MMODE0_7

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	RESET		

Upper Comparator Trigger Level Post-Processing-Channel 0-3

ADC1_TH0_3_UPPER	Offset	Reset Value
Upper Comparator Trigger Level Post-Processing-Channel 0-3	$C8_H$	see Table 485



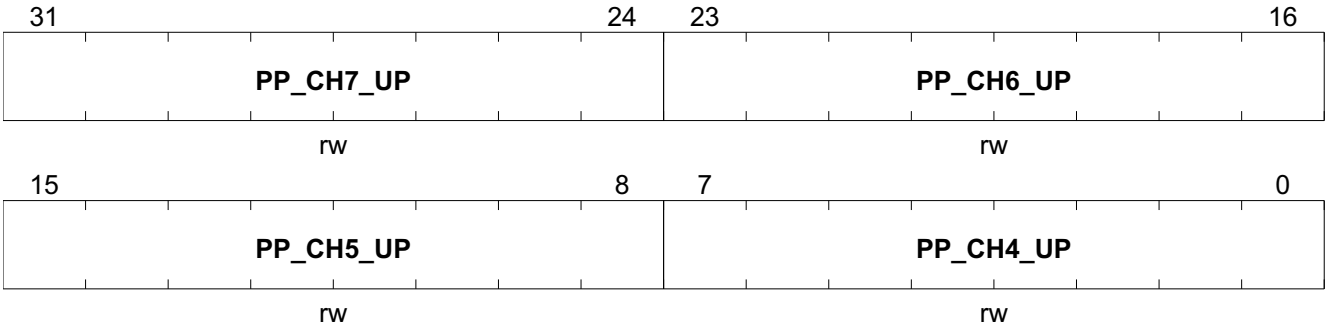
Field	Bits	Type	Description
PP_CH3_UP	31:24	rw	**Post-Processing-Channel 3 upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value
PP_CH2_UP	23:16	rw	**Post-Processing-Channel 2 upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value
PP_CH1_UP	15:8	rw	**Post-Processing-Channel 1 upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value
PP_CH0_UP	7:0	rw	**Post-Processing-Channel 0 upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value

Table 485 RESET of ADC1_TH0_3_UPPER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	FFFF C5C0$_H$	RESET_TYPE_4		
TRIM_100_TP	FFFF C5C0$_H$	RESET		

Upper Comparator Trigger Level Post-Processing-Channel 4-7

ADC1_TH4_7_UPPER	Offset	Reset Value
Upper Comparator Trigger Level Post-Processing-Channel 4-7	CC_H	see Table 486



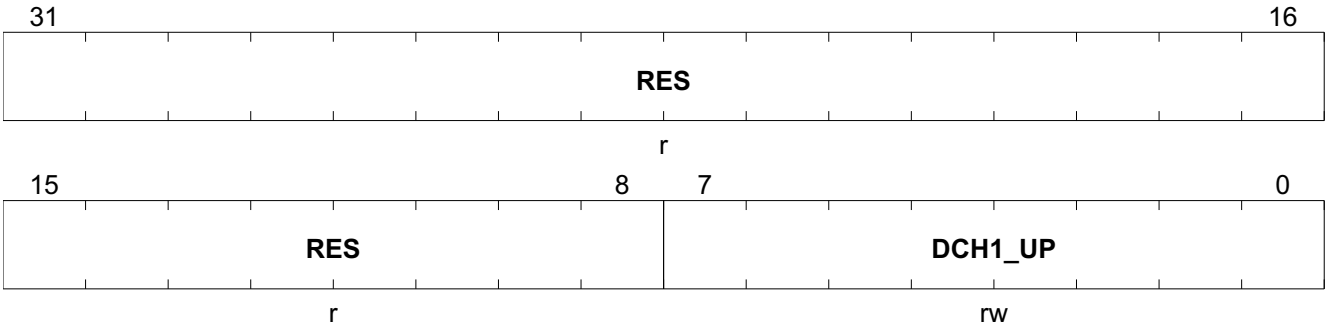
Field	Bits	Type	Description
PP_CH7_UP	31:24	rw	**Post-Processing-Channel 7 upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value
PP_CH6_UP	23:16	rw	**Post-Processing-Channel 6upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value
PP_CH5_UP	15:8	rw	**Post-Processing-Channel 5 upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value
PP_CH4_UP	7:0	rw	**Post-Processing-Channel 4 upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value

Table 486 RESET of ADC1_TH4_7_UPPER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	FFFF FFFF$_H$	RESET_TYPE_4		
TRIM_100_TP	FFFF FFFF$_H$	RESET		

Upper Comparator Trigger Level Differential Channel 1

ADC1_DCHTH1_4_UPPER	Offset	Reset Value
Upper Comparator Trigger Level Differential Channel 1	$D4_H$	see Table 487



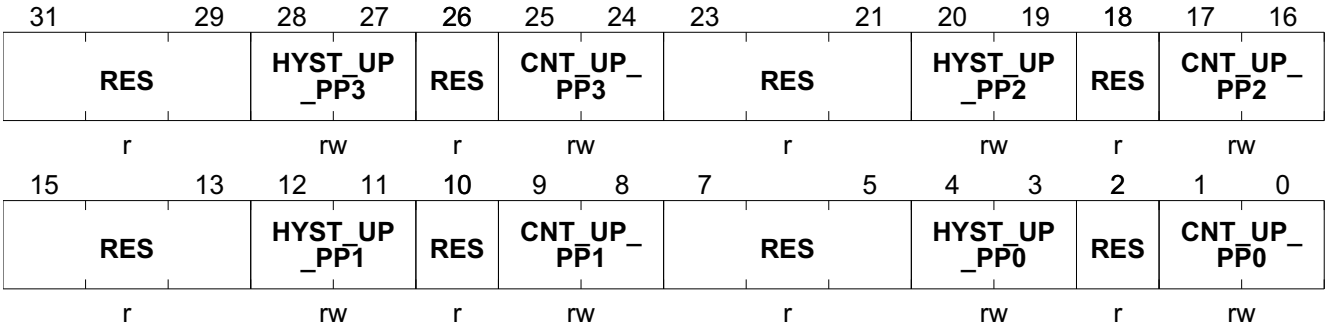
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Always read as 0
DCH1_UP	7:0	rw	**Differential Channel 1 upper trigger level** 00_H **0**, min. threshold value FF_H **255**, max. threshold value

Table 487 RESET of ADC1_DCHTH1_4_UPPER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	0000 $00FF_H$	RESET_TYPE_4		
TRIM_100_TP	0000 $00FF_H$	RESET		

Upper Counter Trigger Level Post-Processing-Channel 0-3

ADC1_CNT0_3_UPPER	Offset	Reset Value
Upper Counter Trigger Level Post-Processing-Channel 0-3	$E8_H$	see Table 488



Field	Bits	Type	Description
RES	31:29	r	**Reserved** Always read as 0
HYST_UP_PP3	28:27	rw	**Post-Processing-Channel 3 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	26	r	**Reserved** Always read as 0
CNT_UP_PP3	25:24	rw	**Upper timer trigger threshold Post-Processing-Channel 3** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	23:21	r	**Reserved** Always read as 0
HYST_UP_PP2	20:19	rw	**Post-Processing-Channel 2 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	18	r	**Reserved** Always read as 0

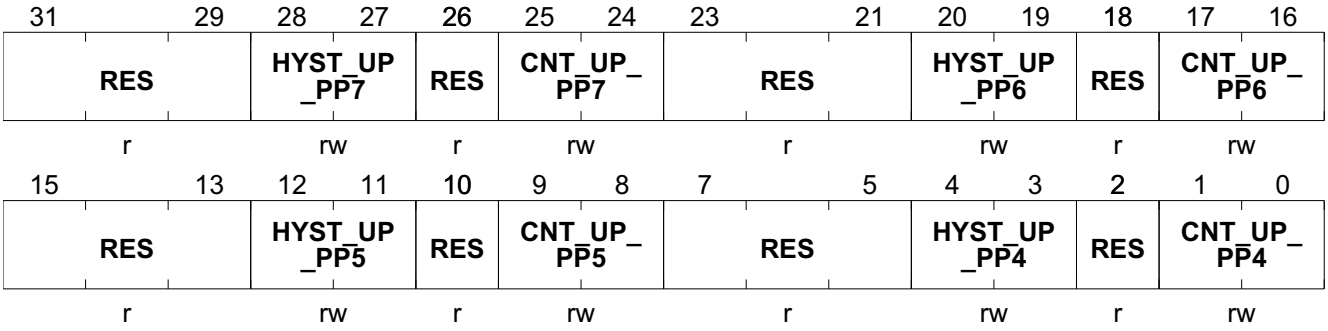
Field	Bits	Type	Description
CNT_UP_PP2	17:16	rw	**Upper timer trigger threshold Post-Processing-Channel 2** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	15:13	r	**Reserved** Always read as 0
HYST_UP_PP1	12:11	rw	**Post-Processing-Channel 1 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	10	r	**Reserved** Always read as 0
CNT_UP_PP1	9:8	rw	**Upper timer trigger threshold Post-Processing-Channel 1** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	7:5	r	**Reserved** Always read as 0
HYST_UP_PP0	4:3	rw	**Post-Processing-Channel 0 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	2	r	**Reserved** Always read as 0
CNT_UP_PP0	1:0	rw	**Upper timer trigger threshold Post-Processing-Channel 0** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements

Table 488 RESET of ADC1_CNT0_3_UPPER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	0000 1B1A$_H$	RESET_TYPE_4		
TRIM_100_TP	0000 1B1A$_H$	RESET		

Upper Counter Trigger Level Post-Processing-Channel 4-7

ADC1_CNT4_7_UPPER	Offset	Reset Value
Upper Counter Trigger Level Post-Processing-Channel 4-7	EC_H	see Table 489



Field	Bits	Type	Description
RES	31:29	r	**Reserved** Always read as 0
HYST_UP_PP7	28:27	rw	**Post-Processing-Channel 7 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	26	r	**Reserved** Always read as 0
CNT_UP_PP7	25:24	rw	**Upper timer trigger threshold Post-Processing-Channel 7** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	23:21	r	**Reserved** Always read as 0
HYST_UP_PP6	20:19	rw	**Post-Processing-Channel 6 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	18	r	**Reserved** Always read as 0

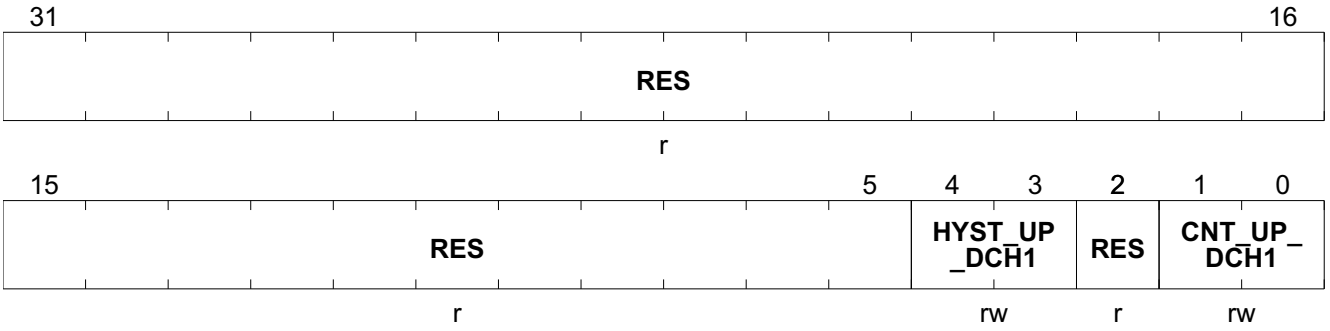
Field	Bits	Type	Description
CNT_UP_PP6	17:16	rw	**Upper timer trigger threshold Post-Processing-Channel 6** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	15:13	r	**Reserved** Always read as 0
HYST_UP_PP5	12:11	rw	**Post-Processing-Channel 5 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	10	r	**Reserved** Always read as 0
CNT_UP_PP5	9:8	rw	**Upper timer trigger threshold Post-Processing-Channel 5** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	7:5	r	**Reserved** Always read as 0
HYST_UP_PP4	4:3	rw	**Post-Processing-Channel 4 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	2	r	**Reserved** Always read as 0
CNT_UP_PP4	1:0	rw	**Upper timer trigger threshold Post-Processing-Channel 4** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements

Table 489 RESET of ADC1_CNT4_7_UPPER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	RESET		

Upper Counter Trigger Level Differential Channel 1

ADC1_DCHCNT1_4_UPPER	Offset	Reset Value
Upper Counter Trigger Level Differential Channel 1	$F4_H$	see Table 490



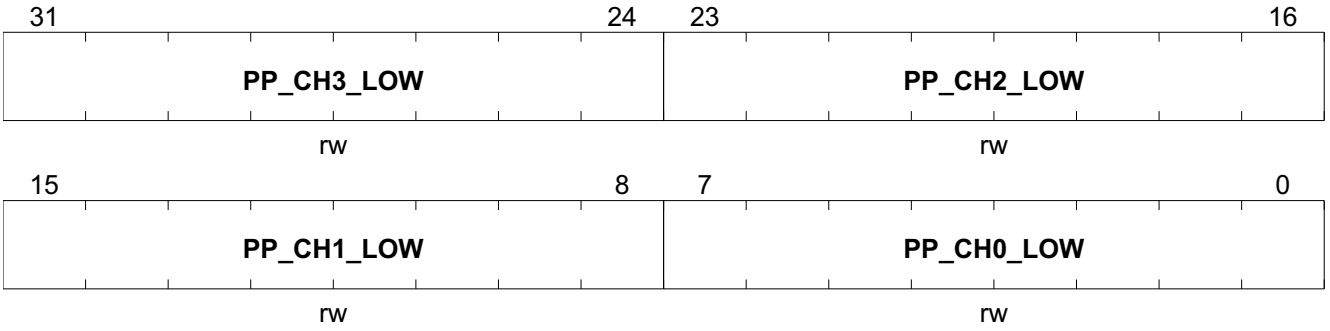
Field	Bits	Type	Description
RES	31:5	r	**Reserved** Always read as 0
HYST_UP_DCH1	4:3	rw	**Differential Channel 1 upper hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	2	r	**Reserved** Always read as 0
CNT_UP_DCH1	1:0	rw	**Upper timer trigger threshold Differential Channel 1** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements

Table 490 RESET of ADC1_DCHCNT1_4_UPPER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	0000 0000_H	RESET_TYPE_4		
TRIM_100_TP	0000 0000_H	RESET		

Lower Comparator Trigger Level Post-Processing-Channel 0-3

ADC1_TH0_3_LOWER	Offset	Reset Value
Lower Comparator Trigger Level Post-Processing-Channel 0-3	40_H	see Table 491



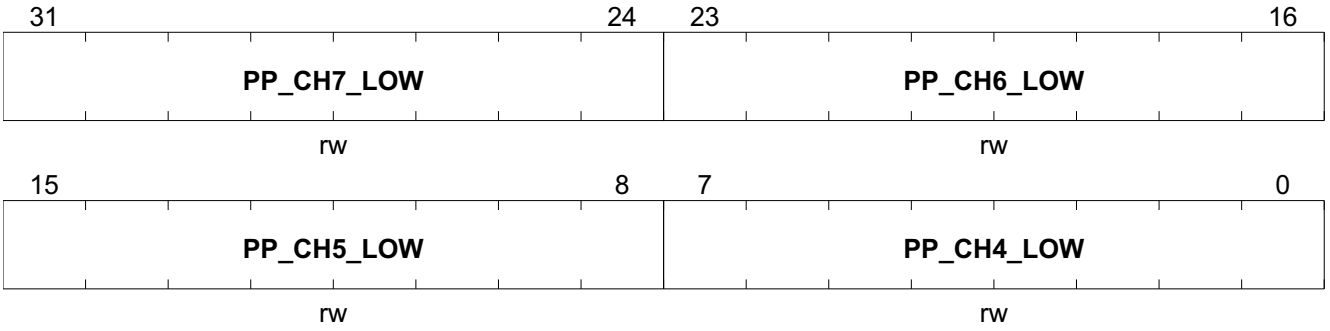
Field	Bits	Type	Description
PP_CH3_LOW	31:24	rw	**Post-Processing-Channel 3 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value
PP_CH2_LOW	23:16	rw	**Post-Processing-Channel 2 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value
PP_CH1_LOW	15:8	rw	**Post-Processing-Channel 1 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value
PP_CH0_LOW	7:0	rw	**Post-Processing-Channel 0 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value

Table 491 RESET of ADC1_TH0_3_LOWER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	0000 $423A_H$	RESET_TYPE_4		
TRIM_100_TP	0000 $423A_H$	RESET		

Lower Comparator Trigger Level Post-Processing-Channel 4-7

ADC1_TH4_7_LOWER	Offset	Reset Value
Lower Comparator Trigger Level Post-Processing-Channel 4-7	44_H	see Table 492



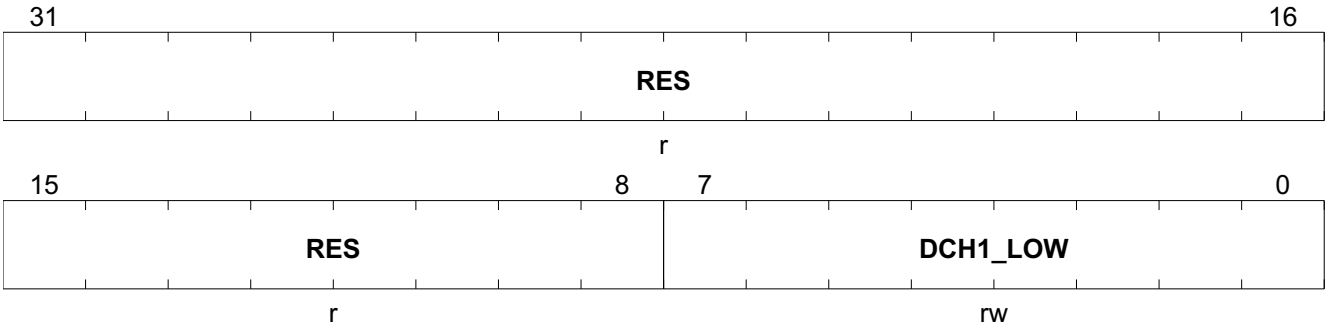
Field	Bits	Type	Description
PP_CH7_LOW	31:24	rw	**Post-Processing-Channel 7 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value
PP_CH6_LOW	23:16	rw	**Post-Processing-Channel 6 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value
PP_CH5_LOW	15:8	rw	**Post-Processing-Channel 5 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value
PP_CH4_LOW	7:0	rw	**Post-Processing-Channel 4 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value

Table 492 RESET of ADC1_TH4_7_LOWER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	RESET		

Lower Comparator Trigger Level Differential Channel 1

ADC1_DCHTH1_4_LOWER	Offset	Reset Value
Lower Comparator Trigger Level Differential Channel 1	$C4_H$	see Table 493



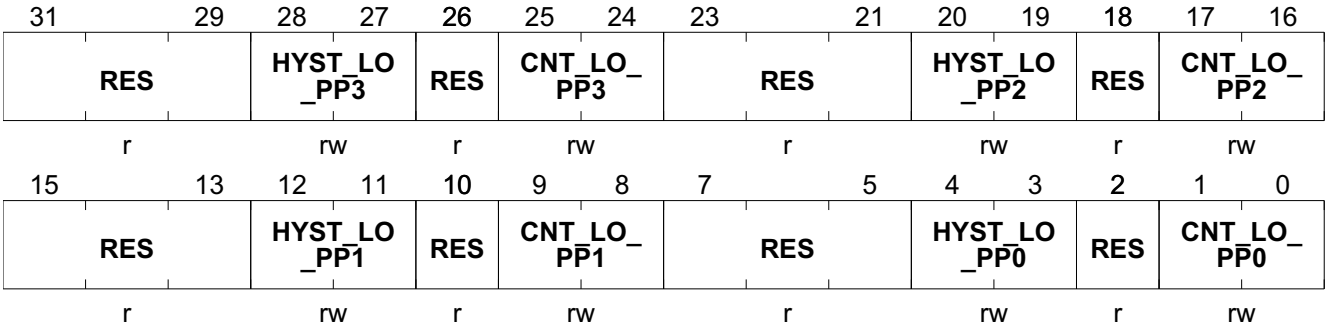
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Always read as 0
DCH1_LOW	7:0	rw	**Differential Channel 1 lower trigger level** 00_H **0**, Min. threshold value FF_H **255**, Max. threshold value

Table 493 RESET of ADC1_DCHTH1_4_LOWER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	RESET		

Lower Counter Trigger Level Post-Processing-Channel 0-3

ADC1_CNT0_3_LOWER	Offset	Reset Value
Lower Counter Trigger Level Post-Processing-Channel 0-3	$D8_H$	see Table 494



Field	Bits	Type	Description
RES	31:29	r	**Reserved** Always read as 0
HYST_LO_PP3	28:27	rw	**Post-Processing-Channel 3 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	26	r	**Reserved** Always read as 0
CNT_LO_PP3	25:24	rw	**Lower timer trigger threshold Post-Processing-Channel 3** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	23:21	r	**Reserved** Always read as 0
HYST_LO_PP2	20:19	rw	**Post-Processing-Channel 2 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	18	r	**Reserved** Always read as 0

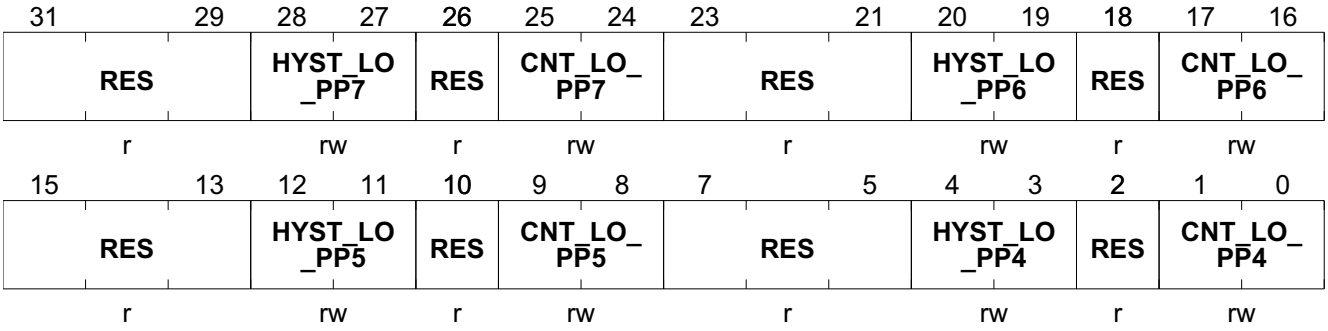
Field	Bits	Type	Description
CNT_LO_PP2	17:16	rw	**Lower timer trigger threshold Post-Processing-Channel 2** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	15:13	r	**Reserved** Always read as 0
HYST_LO_PP1	12:11	rw	**Post-Processing-Channel 1 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	10	r	**Reserved** Always read as 0
CNT_LO_PP1	9:8	rw	**Lower timer trigger threshold Post-Processing-Channel 1** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	7:5	r	**Reserved** Always read as 0
HYST_LO_PP0	4:3	rw	**Post-Processing-Channel 0 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	2	r	**Reserved** Always read as 0
CNT_LO_PP0	1:0	rw	**Lower timer trigger threshold Post-Processing-Channel 0** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements

Table 494 RESET of ADC1_CNT0_3_LOWER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	0000 1312_H	RESET_TYPE_4		
TRIM_100_TP	0000 1312_H	RESET		

Lower Counter Trigger Level Post-Processing-Channel 4-7

ADC1_CNT4_7_LOWER	Offset	Reset Value
Lower Counter Trigger Level Post-Processing-Channel 4-7	DC_H	see Table 495



Field	Bits	Type	Description
RES	31:29	r	**Reserved** Always read as 0
HYST_LO_PP7	28:27	rw	**Post-Processing-Channel 7 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	26	r	**Reserved** Always read as 0
CNT_LO_PP7	25:24	rw	**Lower timer trigger threshold Post-Processing-Channel 7** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	23:21	r	**Reserved** Always read as 0
HYST_LO_PP6	20:19	rw	**Channel 6 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	18	r	**Reserved** Always read as 0

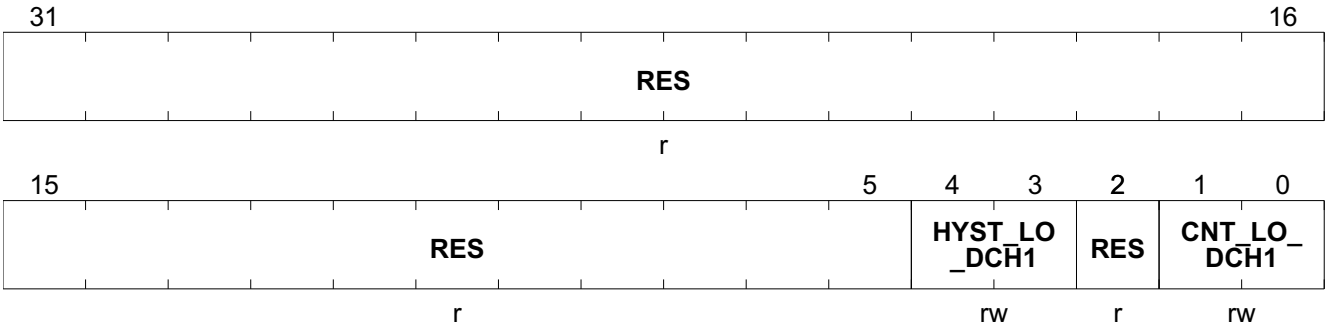
Field	Bits	Type	Description
CNT_LO_PP6	17:16	rw	**Lower timer trigger threshold Post-Processing-Channel 6** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	15:13	r	**Reserved** Always read as 0
HYST_LO_PP5	12:11	rw	**Post-Processing-Channel 5 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	10	r	**Reserved** Always read as 0
CNT_LO_PP5	9:8	rw	**Lower timer trigger threshold Post-Processing-Channel 5** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements
RES	7:5	r	**Reserved** Always read as 0
HYST_LO_PP4	4:3	rw	**Post-Processing-Channel 4 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	2	r	**Reserved** Always read as 0
CNT_LO_PP4	1:0	rw	**Lower timer trigger threshold Post-Processing-Channel 4** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements

Table 495 RESET of ADC1_CNT4_7_LOWER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	RESET		

Lower Counter Trigger Level Differential Channel 1

ADC1_DCHCNT1_4_LOWER	Offset	Reset Value
Lower Counter Trigger Level Differential Channel 1	$E4_H$	see Table 496



Field	Bits	Type	Description
RES	31:5	r	**Reserved** Always read as 0
HYST_LO_DCH1	4:3	rw	**Differential Channel 1 lower hysteresis** 0_H **HYSTOFF**, hysteresis switched off 1_H **HYST4**, hysteresis = 4 2_H **HYST8**, hysteresis = 8 3_H **HYST16**, hysteresis = 16
RES	2	r	**Reserved** Always read as 0
CNT_LO_DCH1	1:0	rw	**Lower timer trigger threshold Differential Channel 1** 0_H **1**, 1 measurement 1_H **2**, 2 measurements 2_H **4**, 4 measurements 3_H **7**, 7 measurements

Table 496 RESET of ADC1_DCHCNT1_4_LOWER

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	0000 0000_H	RESET_TYPE_4		
TRIM_100_TP	0000 0000_H	RESET		

Analog Digital Converter ADC10B (ADC1)

Post-Processing Mapping Channel 0-3

ADC1_PP_MAP0_3	Offset	Reset Value
Post-Processing Mapping Channel 0-3	118_H	see Table 497

31	30	29	28	27	24	23	22	21	20	19	16
EN_PP_M*	RESET_P*	RES		PP_MAP3		EN_PP_M*	RESET_P*	RES		PP_MAP2	
rw	rw	r		rw		rw	rw	r		rw	

15	14	13	12	11	8	7	6	5	4	3	0
EN_PP_M*	RESET_P*	RES		RES		EN_PP_M*	RESET_P*	RES		RES	
rw	rw	r		r		rw	rw	r		r	

Field	Bits	Type	Description
EN_PP_MAP3	31	rw	**Mapping Enable for Post-Processing-Channel 3** Enable/disable the triggering of the post processing channel 3 0_B **Disabled**, Mapping Disabled 1_B **Enabled**, Mapping Enabled
RESET_PP_MAP3	30	rw	**Post-Processing Reset for Mapped Post-Processing-Channel 3** Reset of STS, up/lo counter in post processing channel 3 0_B **Running**, Post-Processing running 1_B **Reset**, Post-Processing reset
RES	29:28	r	**Reserved** Always read as 0
PP_MAP3	27:24	rw	**Mapping of Entry Channel to Post-Processing-Channel 3** 0_H **Ch0**, Entry Channel 0 D_H **Ch13**, Entry Channel 13 E_H **Reserved**, F_H **Reserved**,
EN_PP_MAP2	23	rw	**Mapping Enable for Post-Processing-Channel 2** Enable/disable the triggering of the post processing channel 2 0_B **Disabled**, Mapping Disabled 1_B **Enabled**, Mapping Enabled
RESET_PP_MAP2	22	rw	**Post-Processing Reset for Mapped Post-Processing-Channel 2** Reset of STS, up/lo counter in post processing channel 2 0_B **Running**, Post-Processing running 1_B **Reset**, Post-Processing reset
RES	21:20	r	**Reserved** Always read as 0

Field	Bits	Type	Description
PP_MAP2	19:16	rw	**Mapping of Entry Channel to Post-Processing-Channel 2** 0_H **Ch0**, Entry Channel 0 D_H **Ch13**, Entry Channel 13 E_H **Reserved**, F_H **Reserved**,
EN_PP_MAP1	15	rw	**Mapping Enable for Post-Processing-Channel 1** Enable/disable the triggering of the post processing channel 1 0_B **Disabled**, Mapping Disabled 1_B **Enabled**, Mapping Enabled
RESET_PP_MAP1	14	rw	**Post-Processing Reset for Mapped Post-Processing-Channel 1** Reset of STS, up/lo counter in post processing channel 1 0_B **Running**, Post-Processing running 1_B **Reset**, Post-Processing reset
RES	13:12	r	**Reserved** Always read as 0
RES	11:8	r	**Reserved** Always read as 1_H
EN_PP_MAP0	7	rw	**Mapping Enable for Post-Processing-Channel 0** Enable/disable the triggering of the post processing channel 0 0_B **Disabled**, Mapping Disabled 1_B **Enabled**, Mapping Enabled
RESET_PP_MAP0	6	rw	**Post-Processing Reset for Mapped Post-Processing-Channel 0** Reset of STS, up/lo counter in post processing channel 0 0_B **Running**, Post-Processing running 1_B **Reset**, Post-Processing reset
RES	5:4	r	**Reserved** Always read as 0
RES	3:0	r	**Reserved** Always read as 0

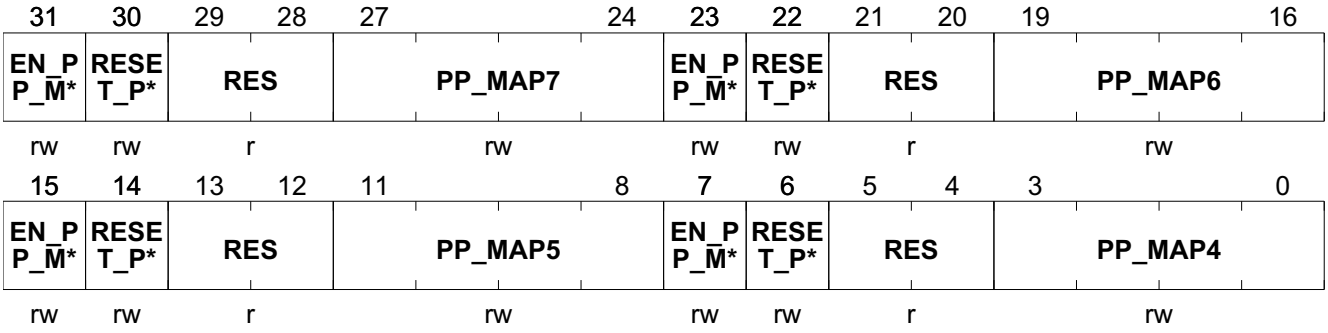
Table 497 RESET of ADC1_PP_MAP0_3

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	0302 0100_H	RESET_TYPE_4		
TRIM_100_TP	0302 0100_H	RESET		

Analog Digital Converter ADC10B (ADC1)

Post-Processing Mapping Channel 4-7

ADC1_PP_MAP4_7	Offset	Reset Value
Post-Processing Mapping Channel 4-7	$11C_H$	see Table 498



Field	Bits	Type	Description
EN_PP_MAP7	31	rw	**Mapping Enable for Post-Processing-Channel 7** Enable/disable the triggering of the post processing channel 7 0_B **Disabled**, Mapping Disabled 1_B **Enabled**, Mapping Enabled
RESET_PP_MAP7	30	rw	**Post-Processing Reset for Mapped Post-Processing-Channel 7** Reset of STS, up/lo counter in post processing channel 7 0_B **Running**, Post-Processing running 1_B **Reset**, Post-Processing reset
RES	29:28	r	**Reserved** Always read as 0
PP_MAP7	27:24	rw	**Mapping of Entry Channel to Post-Processing-Channel 7** 0_H **Ch0**, Entry Channel 0 D_H **Ch13**, Entry Channel 13 E_H **Reserved**, F_H **Reserved**,
EN_PP_MAP6	23	rw	**Mapping Enable for Post-Processing-Channel 6** Enable/disable the triggering of the post processing channel 6 0_B **Disabled**, Mapping Disabled 1_B **Enabled**, Mapping Enabled
RESET_PP_MAP6	22	rw	**Post-Processing Reset for Mapped Post-Processing-Channel 6** Reset of STS, up/lo counter in post processing channel 6 0_B **Running**, Post-Processing running 1_B **Reset**, Post-Processing reset
RES	21:20	r	**Reserved** Always read as 0

Field	Bits	Type	Description
PP_MAP6	19:16	rw	**Mapping of Entry Channel to Post-Processing-Channel 6** 0_H **Ch0**, Entry Channel 0 D_H **Ch13**, Entry Channel 13 E_H **Reserved**, F_H **Reserved**,
EN_PP_MAP5	15	rw	**Mapping Enable for Post-Processing-Channel 5** Enable/disable the triggering of the post processing channel 5 0_B **Disabled**, Mapping Disabled 1_B **Enabled**, Mapping Enabled
RESET_PP_MAP5	14	rw	**Post-Processing Reset for Mapped Post-Processing-Channel 5** Reset of STS, up/lo counter in post processing channel 5 0_B **Running**, Post-Processing running 1_B **Reset**, Post-Processing reset
RES	13:12	r	**Reserved** Always read as 0
PP_MAP5	11:8	rw	**Mapping of Entry Channel to Post-Processing-Channel 5** 0_H **Ch0**, Entry Channel 0 D_H **Ch13**, Entry Channel 13 E_H **Reserved**, F_H **Reserved**,
EN_PP_MAP4	7	rw	**Mapping Enable for Post-Processing-Channel 4** Enable/disable the triggering of the post processing channel 4 0_B **Disabled**, Mapping Disabled 1_B **Enabled**, Mapping Enabled
RESET_PP_MAP4	6	rw	**Post-Processing Reset for Mapped Post-Processing-Channel 4** Reset of STS, up/lo counter in post processing channel 4 0_B **Running**, Post-Processing running 1_B **Reset**, Post-Processing reset
RES	5:4	r	**Reserved** Always read as 0
PP_MAP4	3:0	rw	**Mapping of Entry Channel to Post-Processing-Channel 4** 0_H **Ch0**, Entry Channel 0 D_H **Ch13**, Entry Channel 13 E_H **Reserved**, F_H **Reserved**,

Table 498 RESET of ADC1_PP_MAP4_7

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	0807 0604$_H$	RESET_TYPE_4		
TRIM_100_TP	0807 0604$_H$	RESET		

24.9 Interrupt Handling

24.9.1 Functional Description

Figure 205 shows the interrupt generation of ADC1.The generated interrupts are assigned to several nodes. The exact mapping can be red in the corresponding interrupt chapter of this device.

Note: all status flags and interrupt status flags are blanked within the startup procedure of the sequencer. The purpose of this is to avoid wrong setting of those flags due to settling behaviour of the integrated filter structures.

Figure 204 gives an Overview of the dedicated interupt structure for MON, P2.x inputs and Current Sense Amplifier using the flexible post processing assignment scheme. In principle each of the available post processing channels can be assigned to to any available channel on the sequencer and with this to any available input voltage of ADC1. Based on the post processing assignment the intelligent interupt assignment will link the input channel to its corresponding interupt node.

Example:

Post processing channel 0 is connected to MON1 (ADC1_PP_MAP0_3.PP_MAP0 = 0x2)

-> Post processingchannel 0 linked to MON interupt node.

POst processing channel 2 is connected to P2.0 (ADC1_PP_MAP0_3.PP_MAP1 = 0x6)

-> Post processing channel is linked to P2.x Interupt node.

Figure 206, **Figure 207**, **Figure 208** showing a detailed description of the dedicated interupt node assignment.

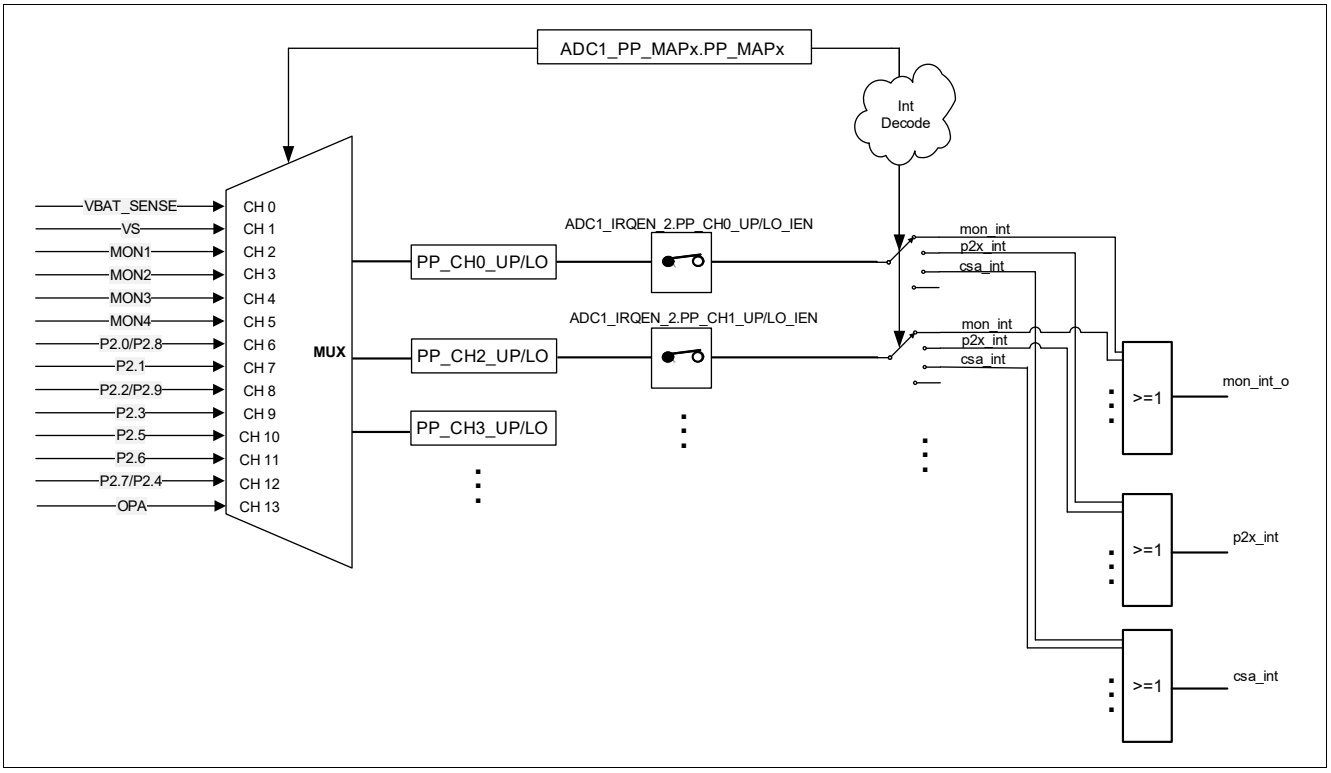


Figure 204 Overview flexible post processing to Interupt node assignment

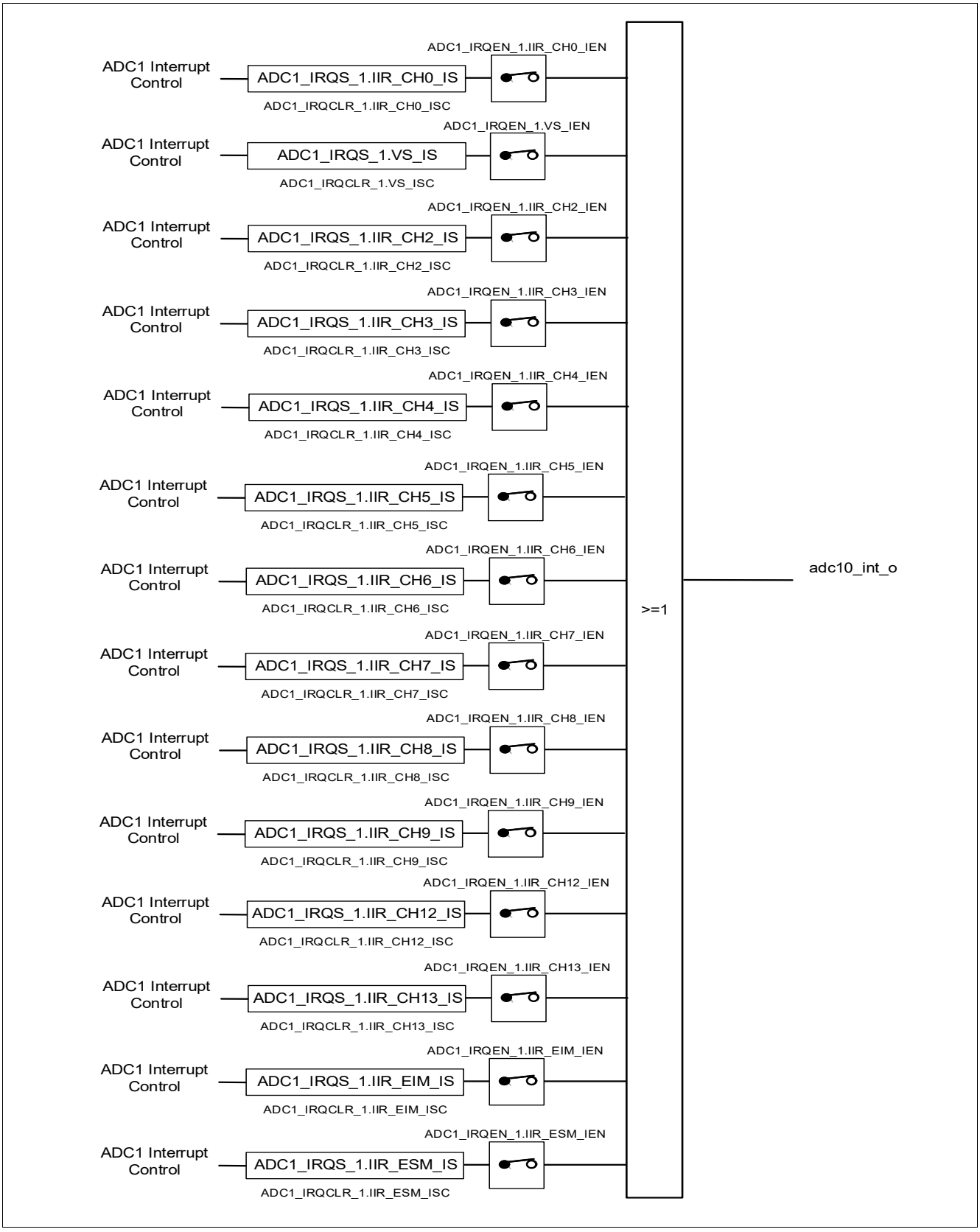


Figure 205 ADC1 Interrupt Generation of all existing channels

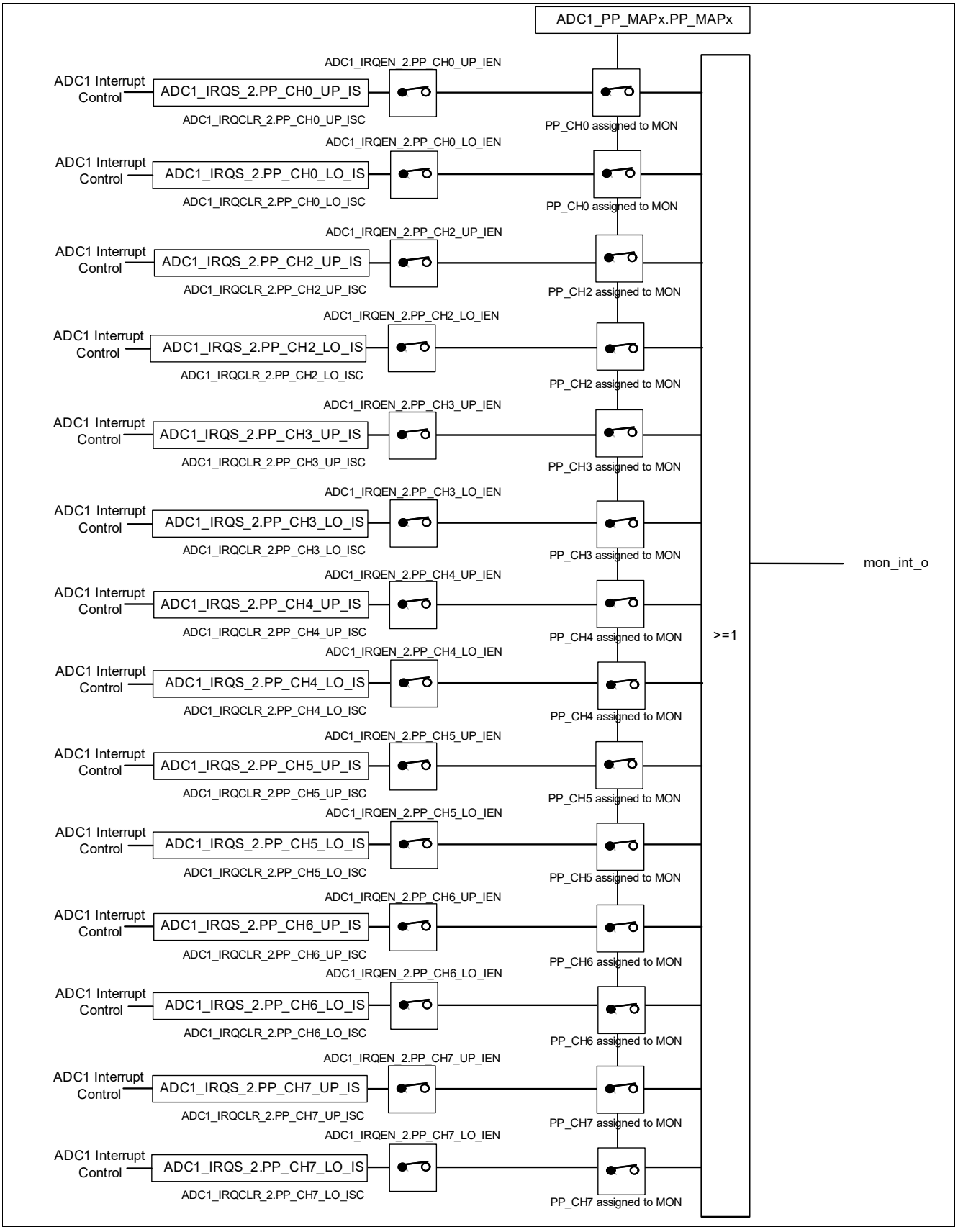


Figure 206 ADC1 Interrupt Generation for MON

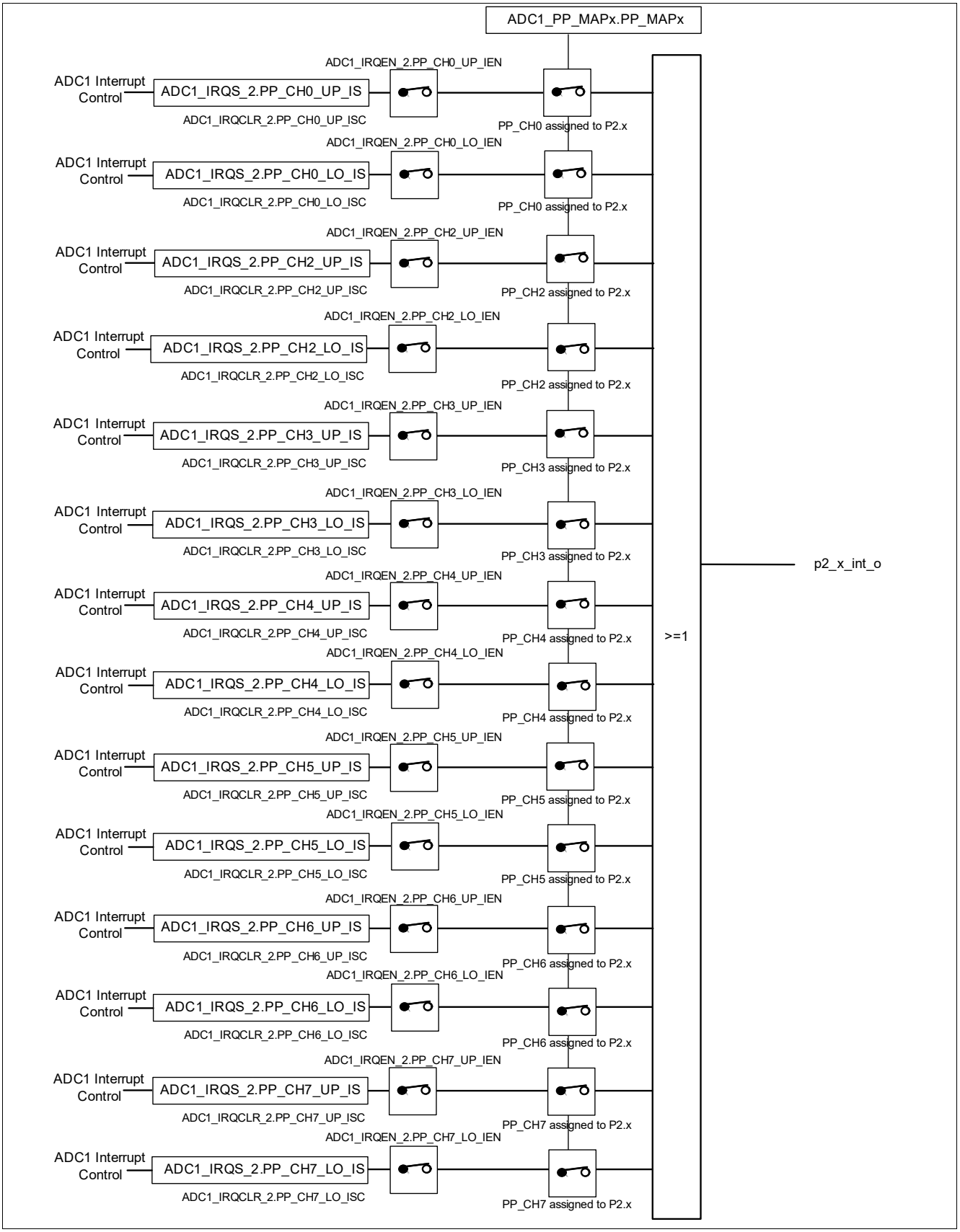


Figure 207 ADC1 Interrupt Generation for P2.x

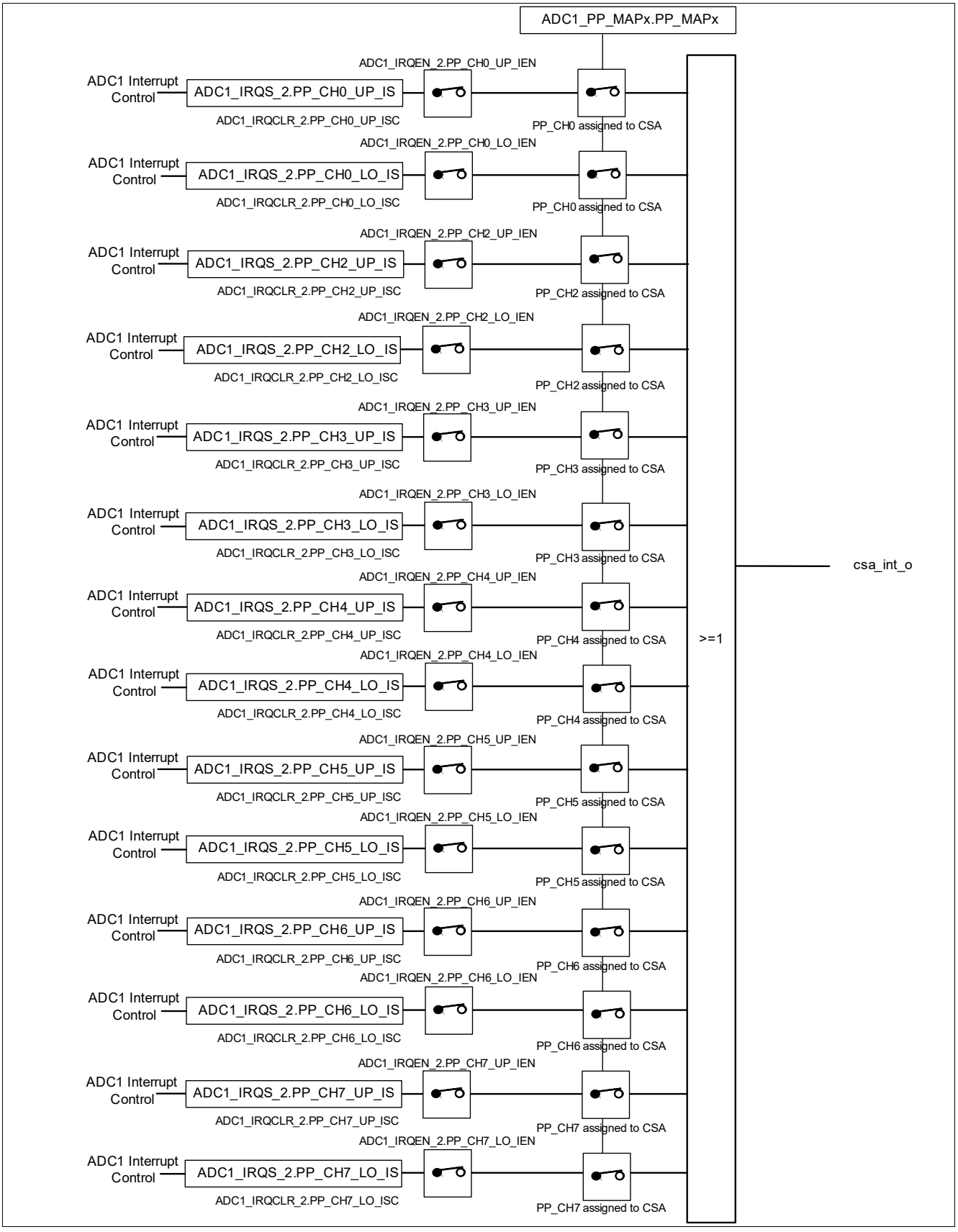


Figure 208 ADC1 Interrupt Generation for Current Sense Amplifier

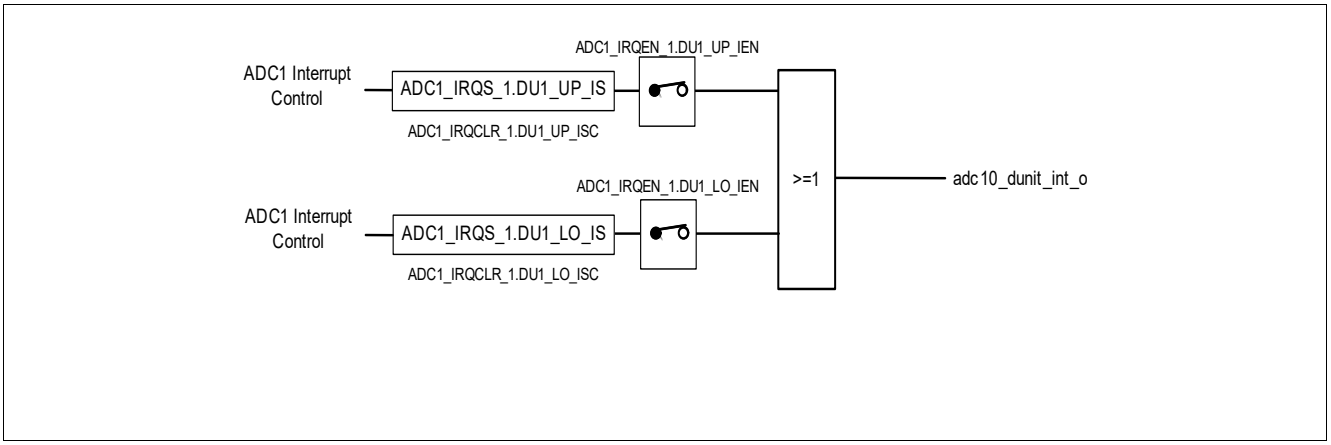


Figure 209 ADC1 Interrupt Generation for Differential Unit

24.9.2 Interrupt Registers

Table 499 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Interrupt Registers,			
ADC1_IRQS_1	ADC1 Interrupt Status 1 Register	64_H	see **Table 500**
ADC1_IRQEN_1	ADC1 Interrupt Enable 1 Register	68_H	see **Table 505**
ADC1_IRQCLR_1	ADC1 Interrupt Status Clear 1 Register	$6C_H$	see **Table 503**
ADC1_IRQS_2	ADC1 Interrupt Status 2 Register	100_H	see **Table 501**
ADC1_STS_2	ADC1 Status 2 Register	104_H	see **Table 502**
ADC1_IRQCLR_2	ADC1 Interrupt Status Clear 2 Register	108_H	see **Table 504**
ADC1_IRQEN_2	ADC1 Interrupt Enable 2 Register	$10C_H$	see **Table 506**
ADC1_STS_1	ADC1 Status 1Register	124_H	see **Table 507**
ADC1_STSCLR_1	ADC1 Status Clear 1 Register	128_H	see **Table 508**

The registers are addressed wordwise.

Measurement Unit 1 Interrupt Status 1 Register

ADC1_IRQS_1	Offset	Reset Value
ADC1 Interrupt Status 1 Register	64_H	see Table 500

31	26	25	24	23	18	17	16
RES		DU1UP_IS	DU1LO_IS	RES		ESM_IS	EIM_IS
r		rwhxre	rwhxre	r		rwhxre	rwhxre

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RES		IIR_CH13*	IIR_CH12*	IIR_CH11*	IIR_CH10*	IIR_CH9*	IIR_CH8*	IIR_CH7*	IIR_CH6*	IIR_CH5*	IIR_CH4*	IIR_CH3*	IIR_CH2*	VS_IS	IIR_CH0*
r		rwhxre	rwhxre	rwhxre	rwhxre	rwhxre	rwhxre	rwhxre	rwhxre	rwhxre	rwhxre	rwhxre	rwhxre	rwhxre	rwhxre

Field	Bits	Type	Description
RES	31:26	r	**Reserved** Always read as 0
DU1UP_IS	25	rwhxre	**ADC1 Differential Unit 1 (DU1) upper Channel Interrupt Status** Conversion of Channel has finished 0_B **INACTIVE**, No DU upper Channel Interrupt has occurred 1_B **ACTIVE**, DU upper Channel Interrupt has occurred
DU1LO_IS	24	rwhxre	**ADC1 Differential Unit 1 (DU1) lower Channel Interrupt Status** Conversion of Channel has finished 0_B **INACTIVE**, No DU lower Channel Interrupt has occurred 1_B **ACTIVE**, DU lower Channel Interrupt has occurred
RES	23:18	r	**Reserved** Always read as 0
ESM_IS	17	rwhxre	**Exceptional Sequence Measurement (ESM) Status** 0_B **INACTIVE**, No ESM has occurred 1_B **ACTIVE**, ESM occurred
EIM_IS	16	rwhxre	**Exceptional Interrupt Measurement (EIM) Status** 0_B **INACTIVE**, No EIM occurred 1_B **ACTIVE**, EIM occurred
RES	15:14	r	**Reserved** Always read as 0
IIR_CH13_IS	13	rwhxre	**ADC1 IIR-Filter-Channel 13 Interrupt Status** Conversion of Channel has finished 0_B **INACTIVE**, No Channel 13 Interrupt has occurred 1_B **ACTIVE**, Channel 13 Interrupt has occurred

Field	Bits	Type	Description
IIR_CH12_IS	12	rwhxre	**ADC1 IIR-Filter-Channel 12 Interrupt Status** Conversion of Channel has finished 0_B **INACTIVE**, No Channel 12 Interrupt has occurred 1_B **ACTIVE**, Channel 12 Interrupt has occurred
IIR_CH11_IS	11	rwhxre	**ADC1 IIR-Filter-Channel 11 Interrupt Status** Conversion of Channel has finished 0_B **INACTIVE**, No Channel 11 Interrupt has occurred 1_B **ACTIVE**, Channel 11 Interrupt has occurred
IIR_CH10_IS	10	rwhxre	**ADC1 IIR-Filter-Channel 10 Interrupt Status** Conversion of Channel has finished 0_B **INACTIVE**, No Channel 10 Interrupt has occurred 1_B **ACTIVE**, Channel 10 Interrupt has occurred
IIR_CH9_IS	9	rwhxre	**ADC1 IIR-Filter-Channel 9 Interrupt Status** Conversion of Channel has finished 0_B **INACTIVE**, No Channel 9 Interrupt has occurred 1_B **ACTIVE**, Channel 9 Interrupt has occurred
IIR_CH8_IS	8	rwhxre	**ADC1 IIR-Filter-Channel 8 Interrupt Status** Conversion of Channel has finished 0_B **INACTIVE**, No Channel 8 Interrupt has occurred 1_B **ACTIVE**, Channel 8 Interrupt has occurred
IIR_CH7_IS	7	rwhxre	**ADC1 IIR-Filter-Channel 7 Interrupt Status** Conversion of Channel has finished 0_B **INACTIVE**, No Channel 7 Interrupt has occurred 1_B **ACTIVE**, Channel 7 Interrupt has occurred
IIR_CH6_IS	6	rwhxre	**ADC1 IIR-Filter-Channel 6 Interrupt Status** Conversion of Channel has finished 0_B **INACTIVE**, No Channel 6 Interrupt has occurred 1_B **ACTIVE**, Channel 6 Interrupt has occurred
IIR_CH5_IS	5	rwhxre	**ADC1 IIR-Filter-Channel 5 Interrupt Status** Conversion of Channel has finished 0_B **INACTIVE**, No Channel 5 Interrupt has occurred 1_B **ACTIVE**, Channel 5 Interrupt has occurred
IIR_CH4_IS	4	rwhxre	**ADC1 IIR-Filter-Channel 4 Interrupt Status** Conversion of Channel has finished 0_B **INACTIVE**, No Channel 4 Interrupt has occurred 1_B **ACTIVE**, Channel 4 Interrupt has occurred
IIR_CH3_IS	3	rwhxre	**ADC1 IIR-Filter-Channel 3 Interrupt Status** Conversion of Channel has finished 0_B **INACTIVE**, No Channel 3 Interrupt has occurred 1_B **ACTIVE**, Channel 3 Interrupt has occurred
IIR_CH2_IS	2	rwhxre	**ADC1 IIR-Filter-Channel 2 Interrupt Status** Conversion of Channel has finished 0_B **INACTIVE**, No Channel 2 Interrupt has occurred 1_B **ACTIVE**, Channel 2 Interrupt has occurred

Field	Bits	Type	Description
VS_IS	1	rwhxre	**ADC1 IIR-Filter-Channel 1 Interrupt Status** Conversion of Channel has finished 0_B **INACTIVE**, No Channel 1 Interrupt has occurred 1_B **ACTIVE**, Channel 1 Interrupt has occurred
IIR_CH0_IS	0	rwhxre	**ADC1 IIR-Filter-Channel 0 Interrupt Status** Conversion of Channel has finished 0_B **INACTIVE**, No Channel 0 Interrupt has occurred 1_B **ACTIVE**, Channel 0 Interrupt has occurred

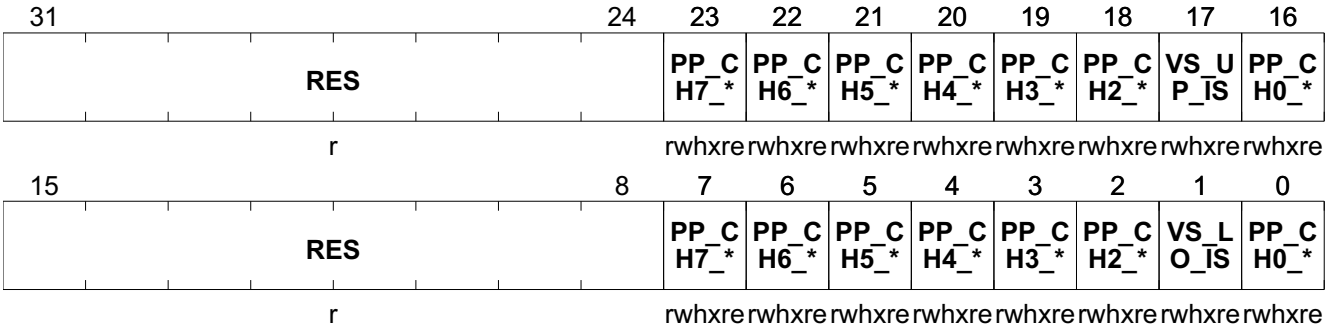
Table 500 RESET of ADC1_IRQS_1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Analog Digital Converter ADC10B (ADC1)

Measurement Unit 1 Status 2 Register

ADC1_IRQS_2	Offset	Reset Value
ADC1 Interrupt Status 2 Register	100_H	see Table 501



Field	Bits	Type	Description
RES	31:24	r	**Reserved** Always read as 0
PP_CH7_UP_IS	23	rwhxre	**ADC1 Post-Processing-Channel 7 Upper Threshold Interrupt Status** 0_B **INACTIVE**, no interrupt has occurred 1_B **ACTIVE**, interrupt has occurred
PP_CH6_UP_IS	22	rwhxre	**ADC1 Post-Processing-Channel 6 Upper Threshold Interrupt Status** 0_B **INACTIVE**, no interrupt has occurred 1_B **ACTIVE**, interrupt has occurred
PP_CH5_UP_IS	21	rwhxre	**ADC1 Post-Processing-Channel 5 Upper Threshold Interrupt Status** 0_B **INACTIVE**, no interrupt has occurred 1_B **ACTIVE**, interrupt has occurred
PP_CH4_UP_IS	20	rwhxre	**ADC1 Post-Processing-Channel 4 Upper Threshold Interrupt Status** 0_B **INACTIVE**, no interrupt has occurred 1_B **ACTIVE**, interrupt has occurred
PP_CH3_UP_IS	19	rwhxre	**ADC1 Post-Processing-Channel 3 Upper Threshold Interrupt Status** 0_B **INACTIVE**, no interrupt has occurred 1_B **ACTIVE**, interrupt has occurred
PP_CH2_UP_IS	18	rwhxre	**ADC1 Post-Processing-Channel 2 Upper Threshold Interrupt Status** 0_B **INACTIVE**, no interrupt has occurred 1_B **ACTIVE**, interrupt has occurred
VS_UP_IS	17	rwhxre	**ADC1 Post-Processing-Channel 1 Upper Threshold Interrupt Status** 0_B **INACTIVE**, no interrupt has occurred 1_B **ACTIVE**, interrupt has occurred

Field	Bits	Type	Description
PP_CH0_UP_IS	16	rwhxre	**ADC1 Post-Processing-Channel 0 Upper Threshold Interrupt Status** 0_B **INACTIVE**, no interrupt has occurred 1_B **ACTIVE**, interrupt has occurred
RES	15:8	r	**Reserved** Always read as 0
PP_CH7_LO_IS	7	rwhxre	**ADC1 Post-Processing-Channel 7 Lower Threshold Interrupt Status** 0_B **INACTIVE**, no interrupt has occurred 1_B **ACTIVE**, interrupt has occurred
PP_CH6_LO_IS	6	rwhxre	**ADC1 Post-Processing-Channel 6 Lower Threshold Interrupt Status** 0_B **INACTIVE**, no interrupt has occurred 1_B **ACTIVE**, interrupt has occurred
PP_CH5_LO_IS	5	rwhxre	**ADC1 Post-Processing-Channel 5 Lower Threshold Interrupt Status** 0_B **INACTIVE**, no interrupt has occurred 1_B **ACTIVE**, interrupt has occurred
PP_CH4_LO_IS	4	rwhxre	**ADC1 Post-Processing-Channel 4 Lower Threshold Interrupt Status** 0_B **INACTIVE**, no interrupt has occurred 1_B **ACTIVE**, interrupt has occurred
PP_CH3_LO_IS	3	rwhxre	**ADC1 Post-Processing-Channel 3 Lower Threshold Interrupt Status** 0_B **INACTIVE**, no interrupt has occurred 1_B **ACTIVE**, interrupt has occurred
PP_CH2_LO_IS	2	rwhxre	**ADC1 Post-Processing-Channel 2 Lower Threshold Interrupt Status** 0_B **INACTIVE**, no interrupt has occurred 1_B **ACTIVE**, interrupt has occurred
VS_LO_IS	1	rwhxre	**ADC1 Post-Processing-Channel 1 Lower Threshold Interrupt Status** 0_B **INACTIVE**, no interrupt has occurred 1_B **ACTIVE**, interrupt has occurred
PP_CH0_LO_IS	0	rwhxre	**ADC1 Post-Processing-Channel 0 Lower Threshold Interrupt Status** 0_B **INACTIVE**, no interrupt has occurred 1_B **ACTIVE**, interrupt has occurred

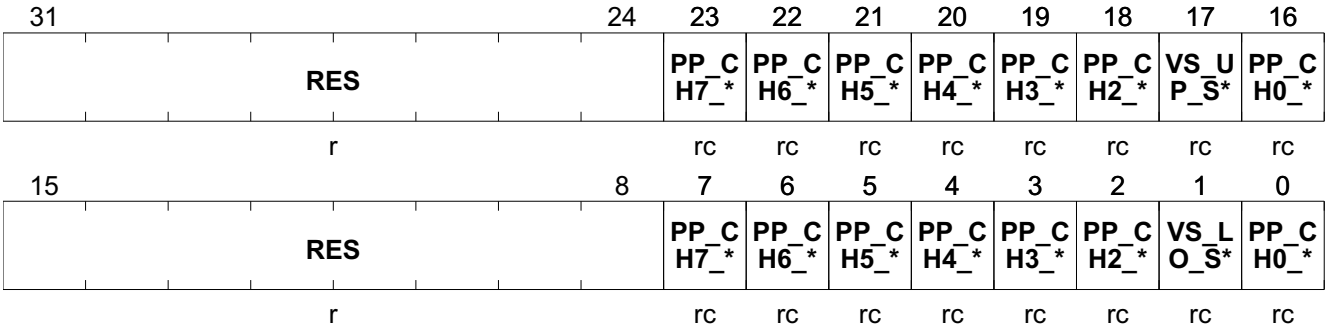
Table 501 RESET of ADC1_IRQS_2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Analog Digital Converter ADC10B (ADC1)

Measurement Unit 1 Status 2 Register

ADC1_STS_2	Offset	Reset Value
ADC1 Status 2 Register	104_H	see Table 502



Field	Bits	Type	Description
RES	31:24	r	**Reserved** Always read as 0
PP_CH7_UP_STS	23	rc	**ADC1 Post-Processing-Channel 7 Upper Threshold Status** 0_B **Below limit**, Status below upper threshold 1_B **Above limit**, Upper threshold exceeded
PP_CH6_UP_STS	22	rc	**ADC1 Post-Processing-Channel 6 Upper Threshold Status** 0_B **Below limit**, Status below upper threshold 1_B **Above limit**, Upper threshold exceeded
PP_CH5_UP_STS	21	rc	**ADC1 Post-Processing-Channel 5 Upper Threshold Status** 0_B **Below limit**, Status below upper threshold 1_B **Above limit**, Upper threshold exceeded
PP_CH4_UP_STS	20	rc	**ADC1 Post-Processing-Channel 4 Upper Threshold Status** 0_B **Below limit**, Status below upper threshold 1_B **Above limit**, Upper threshold exceeded
PP_CH3_UP_STS	19	rc	**ADC1 Post-Processing-Channel 3 Upper Threshold Status** 0_B **Below limit**, Status below upper threshold 1_B **Above limit**, Upper threshold exceeded
PP_CH2_UP_STS	18	rc	**ADC1 Post-Processing-Channel 2 Upper Threshold Status** 0_B **Below limit**, Status below upper threshold 1_B **Above limit**, Upper threshold exceeded
VS_UP_STS	17	rc	**ADC1 Post-Processing-Channel 1 Upper Threshold Status** 0_B **Below limit**, Status below upper threshold 1_B **Above limit**, Upper threshold exceeded

Field	Bits	Type	Description
PP_CH0_UP_STS	16	rc	**ADC1 Post-Processing-Channel 0 Upper Threshold Status** 0_B **Below limit**, Status below upper threshold 1_B **Above limit**, Upper threshold exceeded
RES	15:8	r	**Reserved** Always read as 0
PP_CH7_LO_STS	7	rc	**ADC1 Post-Processing-Channel 7 Lower Threshold Status** 0_B **Below limit**, Status below upper threshold 1_B **Above limit**, Upper threshold exceeded
PP_CH6_LO_STS	6	rc	**ADC1 Post-Processing-Channel 6 Lower Threshold Status** 0_B **Below limit**, Status below upper threshold 1_B **Above limit**, Upper threshold exceeded
PP_CH5_LO_STS	5	rc	**ADC1 Post-Processing-Channel 5 Lower Threshold Status** 0_B **Below limit**, Status below upper threshold 1_B **Above limit**, Upper threshold exceeded
PP_CH4_LO_STS	4	rc	**ADC1 Post-Processing-Channel 4 Lower Threshold Status** 0_B **Below limit**, Status below upper threshold 1_B **Above limit**, Upper threshold exceeded
PP_CH3_LO_STS	3	rc	**ADC1 Post-Processing-Channel 3 Lower Threshold Status** 0_B **Below limit**, Status below upper threshold 1_B **Above limit**, Upper threshold exceeded
PP_CH2_LO_STS	2	rc	**ADC1 Post-Processing-Channel 2 Lower Threshold Status** 0_B **Below limit**, Status below upper threshold 1_B **Above limit**, Upper threshold exceeded
VS_LO_STS	1	rc	**ADC1 Post-Processing-Channel 1 Lower Threshold Status** 0_B **Below limit**, Status below upper threshold 1_B **Above limit**, Upper threshold exceeded
PP_CH0_LO_STS	0	rc	**ADC1 Post-Processing-Channel 0 Lower Threshold Status** 0_B **Below limit**, Status below upper threshold 1_B **Above limit**, Upper threshold exceeded

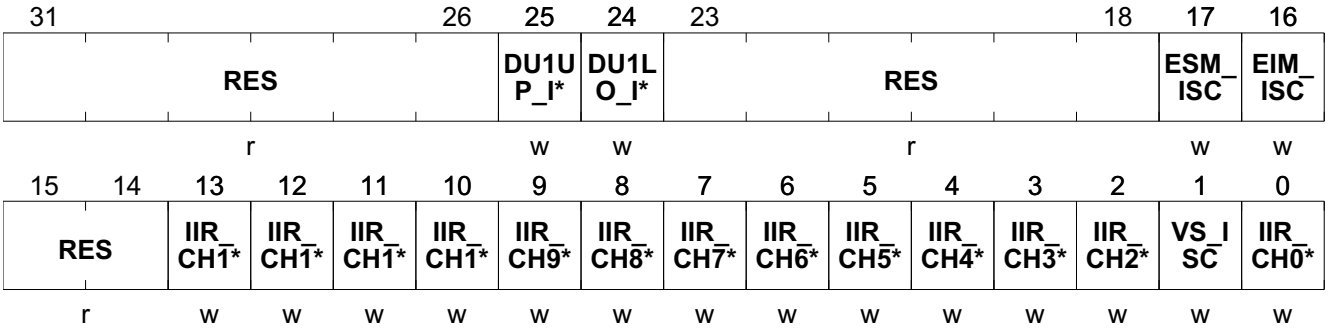
Table 502 RESET of ADC1_STS_2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Analog Digital Converter ADC10B (ADC1)

Measurement Unit 1 Interrupt Status Clear 1 Register

ADC1_IRQCLR_1	Offset	Reset Value
ADC1 Interrupt Status Clear 1 Register	$6C_H$	see Table 503



Field	Bits	Type	Description
RES	31:26	r	**Reserved** Always read as 0
DU1UP_ISC	25	w	**Differential Unit 1 lower Interrupt Status Clear** 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
DU1LO_ISC	24	w	**Differential Unit 1 lower Interrupt Status Clear** 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
RES	23:18	r	**Reserved** Always read as 0
ESM_ISC	17	w	**Exceptional Sequence Measurement (ESM) Status Clear** 0_B **INACTIVE**, No ESM has cleared 1_B **ACTIVE**, ESM cleared
EIM_ISC	16	w	**Exceptional Interrupt Measurement (EIM) Status Clear** 0_B **INACTIVE**, No EIM cleared 1_B **ACTIVE**, EIM cleared
RES	15:14	r	**Reserved** Always read as 0
IIR_CH13_ISC	13	w	**ADC1 IIR-Filter-Channel 13 Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
IIR_CH12_ISC	12	w	**ADC1 IIR-Filter-Channel 12 Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
IIR_CH11_ISC	11	w	**ADC1 IIR-Filter-Channel 11 Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared

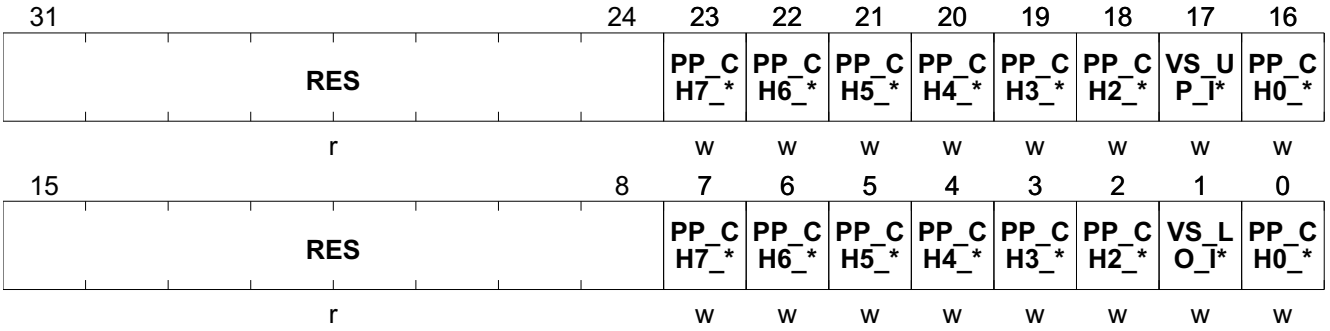
Field	Bits	Type	Description
IIR_CH10_ISC	10	w	**ADC1 IIR-Filter-Channel 10 Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
IIR_CH9_ISC	9	w	**ADC1 IIR-Filter-Channel 9 Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
IIR_CH8_ISC	8	w	**ADC1 IIR-Filter-Channel 8 Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
IIR_CH7_ISC	7	w	**ADC1 IIR-Filter-Channel 7 Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
IIR_CH6_ISC	6	w	**ADC1 IIR-Filter-Channel 6 Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
IIR_CH5_ISC	5	w	**ADC1 IIR-Filter-Channel 5 Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
IIR_CH4_ISC	4	w	**ADC1 IIR-Filter-Channel 4 Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
IIR_CH3_ISC	3	w	**ADC1 IIR-Filter-Channel 3 Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
IIR_CH2_ISC	2	w	**ADC1 IIR-Filter-Channel 2 Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
VS_ISC	1	w	**ADC1 IIR-Filter-Channel 1 Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
IIR_CH0_ISC	0	w	**ADC1 IIR-Filter-Channel 0 Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared

Table 503 RESET of ADC1_IRQCLR_1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Measurement Unit 1 Interrupt Status Clear 2 Register

ADC1_IRQCLR_2	Offset	Reset Value
ADC1 Interrupt Status Clear 2 Register	108_H	see Table 504



Field	Bits	Type	Description
RES	31:24	r	**Reserved** Always read as 0
PP_CH7_UP_ISC	23	w	**ADC1 Post-Processing-Channel 7 Upper Threshold Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
PP_CH6_UP_ISC	22	w	**ADC1 Post-Processing-Channel 6 Upper Threshold Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
PP_CH5_UP_ISC	21	w	**ADC1 Post-Processing-Channel 5 Upper Threshold Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
PP_CH4_UP_ISC	20	w	**ADC1 Post-Processing-Channel 4 Upper Threshold Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
PP_CH3_UP_ISC	19	w	**ADC1 Post-Processing-Channel 3 Upper Threshold Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared

Field	Bits	Type	Description
PP_CH2_UP_ISC	18	w	**ADC1 Post-Processing-Channel 2 Upper Threshold Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
VS_UP_ISC	17	w	**ADC1 Post-Processing-Channel 1 Upper Threshold Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
PP_CH0_UP_ISC	16	w	**ADC1 Post-Processing-Channel 0 Upper Threshold Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
RES	15:8	r	**Reserved** Always read as 0
PP_CH7_LO_ISC	7	w	**ADC1 Post-Processing-Channel 7 Lower Threshold Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
PP_CH6_LO_ISC	6	w	**ADC1 Post-Processing-Channel 6 Lower Threshold Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
PP_CH5_LO_ISC	5	w	**ADC1 Post-Processing-Channel 5 Lower Threshold Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
PP_CH4_LO_ISC	4	w	**ADC1 Post-Processing-Channel 4 Lower Threshold Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
PP_CH3_LO_ISC	3	w	**ADC1 Post-Processing-Channel 3 Lower Threshold Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared

Field	Bits	Type	Description
PP_CH2_LO_ISC	2	w	**ADC1 Post-Processing-Channel 2 Lower Threshold Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
VS_LO_ISC	1	w	**ADC1 Post-Processing-Channel 1 Lower Threshold Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared
PP_CH0_LO_ISC	0	w	**ADC1 Post-Processing-Channel 0 Lower Threshold Interrupt Status Clear** Interrupt status is cleared 0_B **INACTIVE**, interrupt status is not cleared 1_B **ACTIVE**, interrupt status is cleared

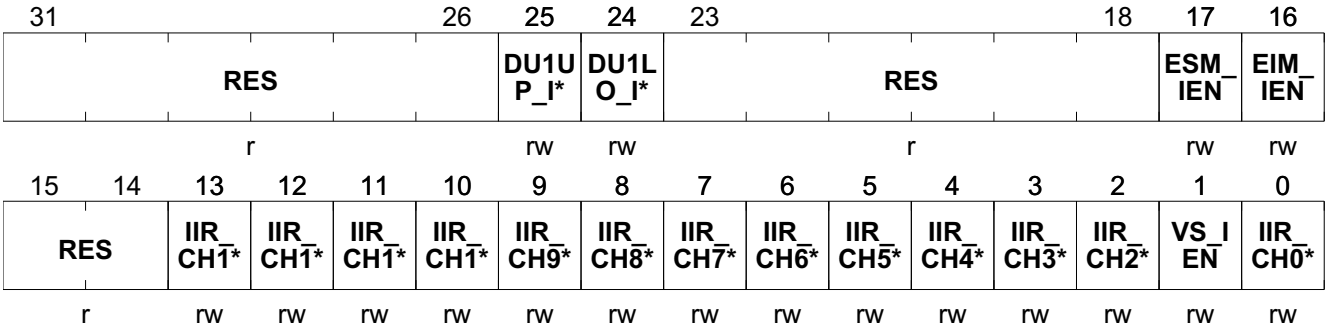
Table 504 RESET of ADC1_IRQCLR_2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000\ 0000_H$	RESET_TYPE_3		

Analog Digital Converter ADC10B (ADC1)

ADC1 Interrupt Enable 1 Register

ADC1_IRQEN_1	Offset	Reset Value
ADC1 Interrupt Enable 1 Register	68_H	see Table 505



Field	Bits	Type	Description
RES	31:26	r	**Reserved** Always read as 0
DU1UP_IEN	25	rw	**Differential Unit 1 upper Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
DU1LO_IEN	24	rw	**Differential Unit 1 lower Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
RES	23:18	r	**Reserved** Always read as 0
ESM_IEN	17	rw	**Exceptional Sequence Measurement (ESM) Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
EIM_IEN	16	rw	**Exceptional Interrupt Measurement (EIM) Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
RES	15:14	r	**Reserved** Always read as 0
IIR_CH13_IEN	13	rw	**ADC1 IIR-Filter-Channel 13 Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
IIR_CH12_IEN	12	rw	**ADC1 IIR-Filter-Channel 12 Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
IIR_CH11_IEN	11	rw	**ADC1 IIR-Filter-Channel 11 Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled

Field	Bits	Type	Description
IIR_CH10_IEN	10	rw	**ADC1 IIR-Filter-Channel 10 Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
IIR_CH9_IEN	9	rw	**ADC1 IIR-Filter-Channel 9 Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
IIR_CH8_IEN	8	rw	**ADC1 IIR-Filter-Channel 8 Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
IIR_CH7_IEN	7	rw	**ADC1 IIR-Filter-Channel 7 Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
IIR_CH6_IEN	6	rw	**ADC1 IIR-Filter-Channel 6 Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
IIR_CH5_IEN	5	rw	**ADC1 IIR-Filter-Channel 5 Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
IIR_CH4_IEN	4	rw	**ADC1 IIR-Filter-Channel 4 Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
IIR_CH3_IEN	3	rw	**ADC1 IIR-Filter-Channel 3 Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
IIR_CH2_IEN	2	rw	**ADC1 IIR-Filter-Channel 2 Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
VS_IEN	1	rw	**ADC1 IIR-Filter-Channel 1 Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
IIR_CH0_IEN	0	rw	**ADC1 IIR-Filter-Channel 0 Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled

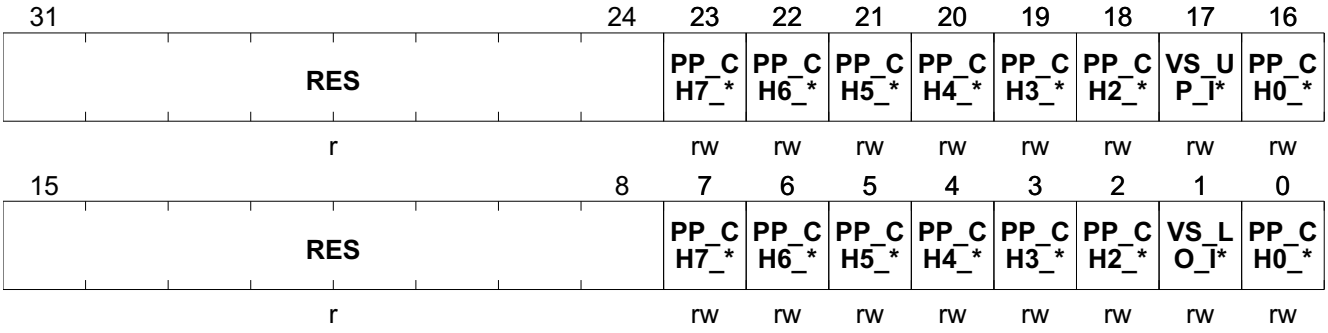
Table 505 RESET of ADC1_IRQEN_1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	RESET		

Analog Digital Converter ADC10B (ADC1)

ADC1 Interrupt Enable 2 Register

ADC1_IRQEN_2	Offset	Reset Value
ADC1 Interrupt Enable 2 Register	$10C_H$	see Table 506



Field	Bits	Type	Description
RES	31:24	r	**Reserved** Always read as 0
PP_CH7_UP_IEN	23	rw	**ADC1 Post-Processing-Channel 7 Upper Threshold Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
PP_CH6_UP_IEN	22	rw	**ADC1 Post-Processing-Channel 6 Upper Threshold Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
PP_CH5_UP_IEN	21	rw	**ADC1 Post-Processing-Channel 5 Upper Threshold Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
PP_CH4_UP_IEN	20	rw	**ADC1 Post-Processing-Channel 4 Upper Threshold Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
PP_CH3_UP_IEN	19	rw	**ADC1 Post-Processing-Channel 3 Upper Threshold Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
PP_CH2_UP_IEN	18	rw	**ADC1 Post-Processing-Channel 2 Upper Threshold Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
VS_UP_IEN	17	rw	**ADC1 Post-Processing-Channel 1 Upper Threshold Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled

Field	Bits	Type	Description
PP_CH0_UP_IEN	16	rw	**ADC1 Post-Processing-Channel 0 Upper Threshold Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
RES	15:8	r	**Reserved** Always read as 0
PP_CH7_LO_IEN	7	rw	**ADC1 Post-Processing-Channel 7 Lower Threshold Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
PP_CH6_LO_IEN	6	rw	**ADC1 Post-Processing-Channel 6 Lower Threshold Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
PP_CH5_LO_IEN	5	rw	**ADC1 Post-Processing-Channel 5 Lower Threshold Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
PP_CH4_LO_IEN	4	rw	**ADC1 Post-Processing-Channel 4 Lower Threshold Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
PP_CH3_LO_IEN	3	rw	**ADC1 Post-Processing-Channel 3 Lower Threshold Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
PP_CH2_LO_IEN	2	rw	**ADC1 Post-Processing-Channel 2 Lower Threshold Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
VS_LO_IEN	1	rw	**ADC1 Post-Processing-Channel 1 Lower Threshold Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled
PP_CH0_LO_IEN	0	rw	**ADC1 Post-Processing-Channel 0 Lower Threshold Interrupt Enable** 0_B **DISABLED** , Interrupt disabled 1_B **ENABLED** , Interrupt enabled

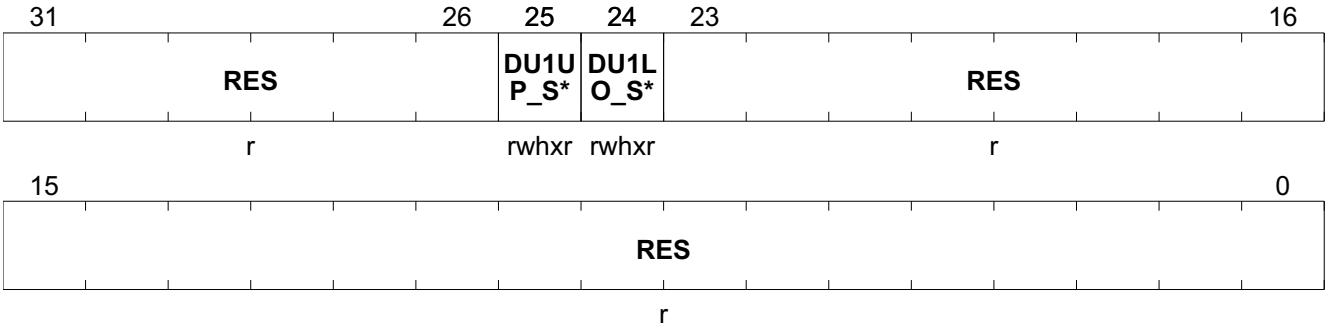
Table 506 RESET of ADC1_IRQEN_2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	RESET		

Analog Digital Converter ADC10B (ADC1)

Measurement Unit 1 Status 1Register

ADC1_STS_1	Offset	Reset Value
ADC1 Status 1Register	124_H	see Table 507



Field	Bits	Type	Description
RES	31:26	r	**Reserved** Always read as 0
DU1UP_STS	25	rwhxr	**ADC1 Differential Unit 1 (DU1) upper Channel Status** Conversion of Channel has finished 0_B **INACTIVE**, No DU upper Channel Status has occurred 1_B **ACTIVE**, DU upper Channel Status has occurred
DU1LO_STS	24	rwhxr	**ADC1 Differential Unit 1 (DU1) lower Channel Status** Conversion of Channel has finished 0_B **INACTIVE**, No DU lower Channel Status has occurred 1_B **ACTIVE**, DU lower Channel Status has occurred
RES	23:0	r	**Reserved** Always read as 0

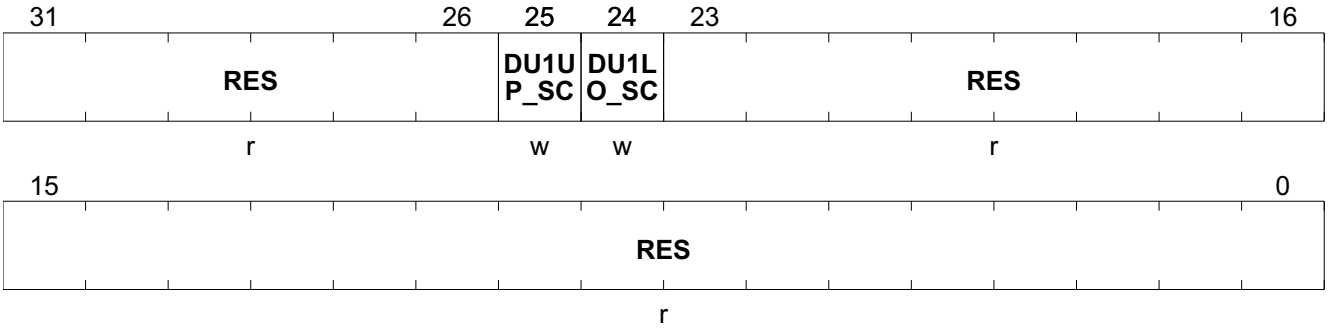
Table 507 RESET of ADC1_STS_1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000_H	RESET_TYPE_3		

Analog Digital Converter ADC10B (ADC1)

Measurement Unit 1 Status Clear 1 Register

ADC1_STSCLR_1	Offset	Reset Value
ADC1 Status Clear 1 Register	128_H	see Table 508



Field	Bits	Type	Description
RES	31:26	r	**Reserved** Always read as 0
DU1UP_SC	25	w	**ADC1 Differential Unit 1 (DU1) upper Channel Status Clear** Conversion of Channel has finished 0_B **INACTIVE**, No DU upper Channel Status has occurred 1_B **ACTIVE**, DU upper Channel Status has occurred
DU1LO_SC	24	w	**ADC1 Differential Unit 1 (DU1) lower Channel Status Clear** Conversion of Channel has finished 0_B **INACTIVE**, No DU lower Channel Status has occurred 1_B **ACTIVE**, DU lower Channel Status has occurred
RES	23:0	r	**Reserved** Always read as 0

Table 508 RESET of ADC1_STSCLR_1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	0000 0000_H	RESET_TYPE_3		

24.10 Differential Measurement Unit

24.10.1 Motivation for Differential Measurement Unit

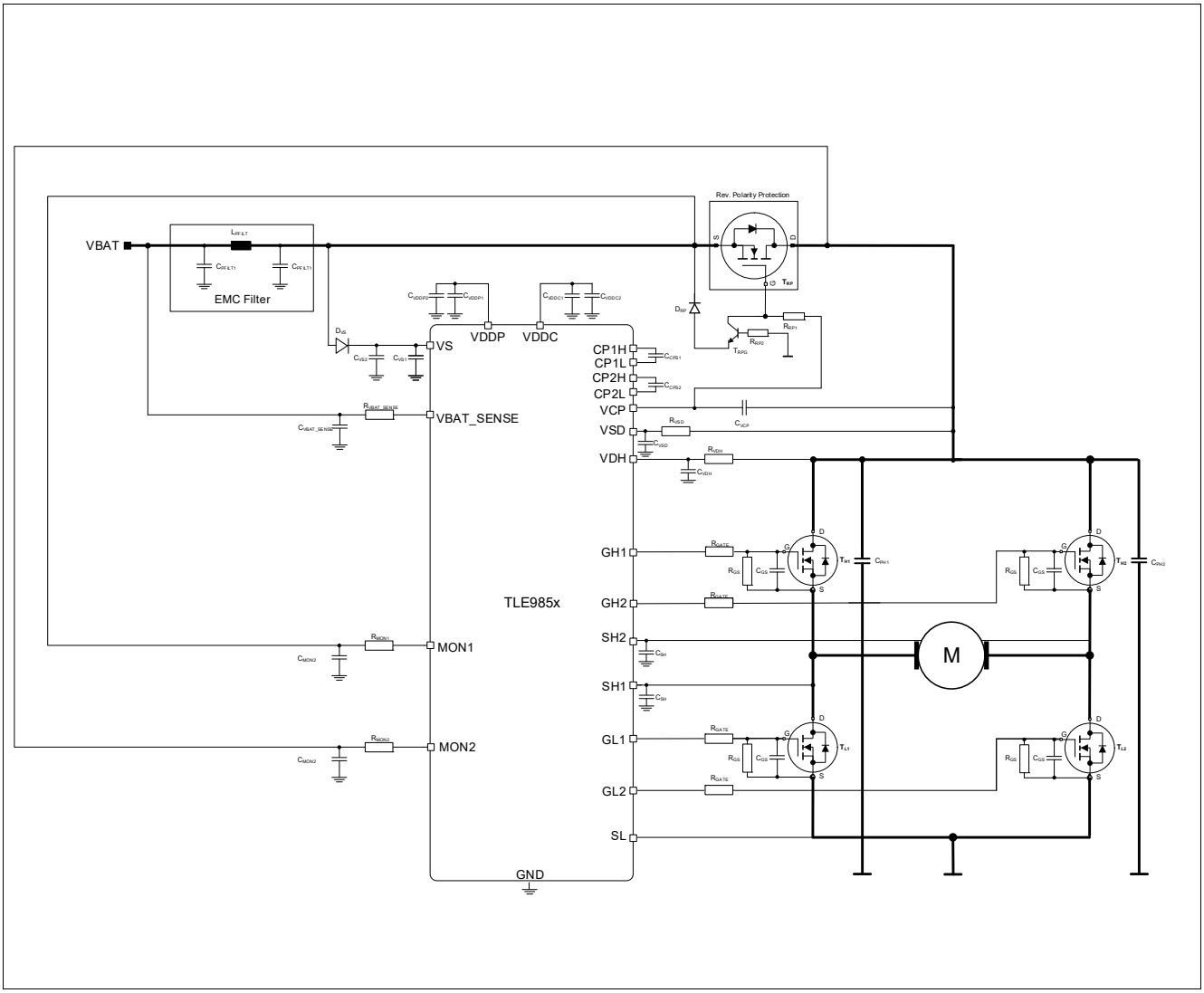


Figure 210 Overcurrent Monitoring of Reverse Polarity MOSFET

24.10.2 Implementation of Differential Measurement Unit

The differential measurement unit is a sub-unit of the digital postprocessing. It calculates the difference between MON1 and MON2. The structure is shown in **Figure 211**.

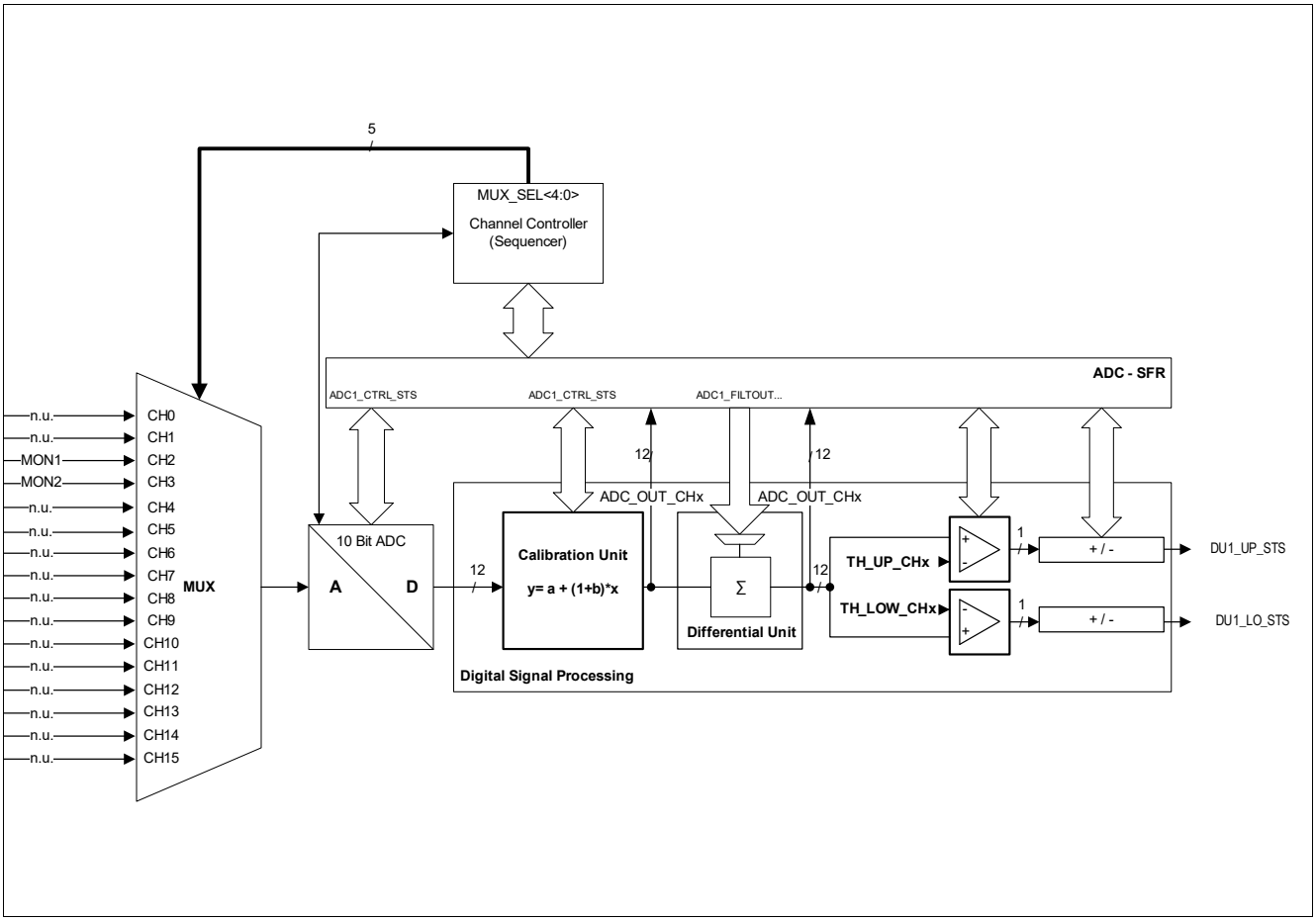


Figure 211 Structure of Differential Measurement Unit

The data processing unit also offers a differential evaluation of the Monitoring Channels MON1 and MON2. This offers the possibility to build up a monitoring of the reverse polarity Power MOS.

Connecting the MON1/MON2 inputs to the reverse polarity power MOS, this can be monitored by the Differential Unit. The Diff. Unit will build the difference out of the result register of MON1 and MON2. The mathematical operation is: DIFF = MON1 - MON2

The control logic for calculating the differential result is built in this way that switching events in the half bridge during a measurement cycle of MON1/MON2 are not taken into account. No calculation is done.

The signal DU1_UP_STS can be used in the Capture/Compare Unit 6 as CTRAP input by configuring Register **CCU6_PISEL0**.ISTRP, for details on the Trap-Handling see **Chapter 18.5**.

24.10.3 ADC1 Differential Unit Input Selection Register

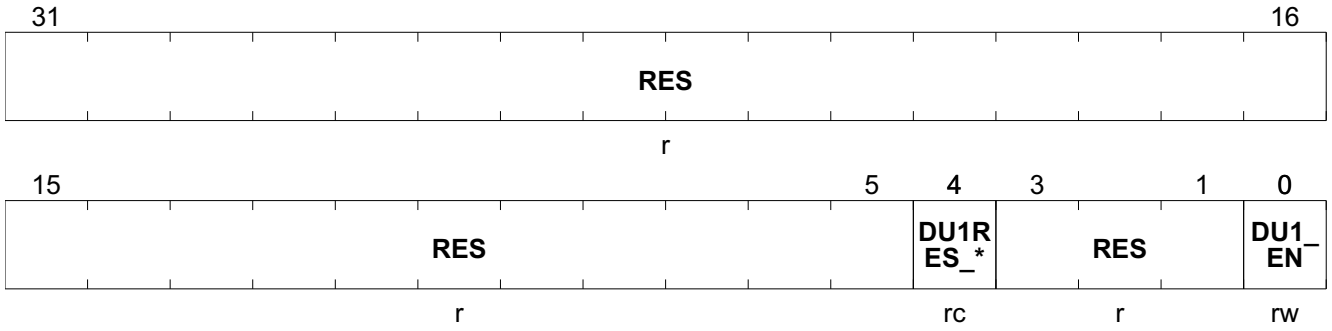
Table 509 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
ADC1 Differential Unit Input Selection Register,			
ADC1_DUIN_SEL	Measurement Unit 1 - Differential Unit Input Selection Register	FC_H	see **Table 510**

The registers are addressed wordwise.

Measurement Unit 1 - Differential Unit Input Selection Register

ADC1_DUIN_SEL	Offset	Reset Value
Measurement Unit 1 - Differential Unit Input Selection Register	FC_H	see Table 510



Field	Bits	Type	Description
RES	31:5	r	**Reserved** Always read as 0
DU1RES_NEG	4	rc	**Differential Unit 1 result negative** *Note: if the calculated result is negative* 0_B **DU1 Result positive**, Differential Unit 1 result positive after calculation 1_B **DU1 Result negative**, Differential Unit 1 result negative after calculation
RES	3:1	r	**Reserved** Always read as 0
DU1_EN	0	rw	**Differential Unit 1 enable** 0_B **DU1 disable**, Differential Unit 1 is disabled 1_B **DU1 enable**, Differential Unit 1 is enabled

Table 510 RESET of ADC1_DUIN_SEL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$0000\ 0000_H$	RESET_TYPE_4		
TRIM_100_TP	$0000\ 0000_H$	RESET		

24.11 Start-up behavior after reset

After the end of a reset phase the measurement sources and the post-processing units need some time for settling. In order to avoid undesired triggering of interrupts until the measurement signal acquisition is in a steady state, the status signals are forced to zero during the start-up phase.

The end of the start-up phase is indicated by the ready signal MI_RDY.

Measurement Core start-up procedure: the startup time of the complete signal chain are 2200 EoC cycles. During startup the IIR-filter coefficient is set to C=2^-1 (fastest response time).

During the startup phase, the DPP will use SQ=1111_1111_1111, regardless of the sequence registers configuration.

During the startup phase, the output registers ADC1_FILTOUT_CHx is normally updated with the converted values. It is recommended to clear all registers before the will be used for applikation purposes.

24.12 Postprocessing Default Values

The following table shows the assigned measurements of the particular channels and the reset default values which are read by FW during power-up. Since all channels are configurable by the user, the reset values can be reconfigured by writting the corresponding sfrs. The mapping of post processing channels can be reassigned in a flexible way

Table 511 Channel allocation and postprocessing default settings (effective after reset)

Channel / MMODE [1]	Analog	Digital [2]	Hyste-resis [3]	IIR - Filter [4]	Counters [5]	Functional Description
PP_Ch. 0/ 0_H **VBAT**	5.79 V	$3A_H$	2_H (8)	2_H (8)	2_H (4)	Battery voltage sense input, lower
	19.27 V	$C0_H$	3_H (16)		2_H (4)	Battery voltage sense input, upper
PP_Ch. 1/ 0_H **VS**	6.59 V	42_H	2_H (8)	2_H (8)	3_H (7)	VS supply voltage input, lower
	19.78 V	$C5_H$	3_H (16)		3_H (7)	VS supply voltage input, upper
PP_Ch. 2/ 0_H **MON1**	0.0 V	00_H	0_H (OFF)	2_H (8)	0_H (1)	MON1, lower
	31.05 V	FF_H	0_H (OFF)		0_H (1)	MON1, upper
PP_Ch. 3/ 0_H **MON2**	0.0 V	00_H	0_H (OFF)	2_H (8)	0_H (1)	MON2, lower
	31.05 V	FF_H	0_H (OFF)		0_H (1)	MON2, upper
PP_Ch. 4/ 0_H **MON3**	0.0 V	00_H	0_H (OFF)	2_H (8)	0_H (1)	MON3, lower
	31.05 V	FF_H	0_H (OFF)		0_H (1)	MON3, upper
PP_Ch. 5/ 0_H **P2.0**	0.0 V	00_H	0_H (OFF)	2_H (8)	0_H (1)	P2.0, lower
	5.33 V	FF_H	0_H (OFF)		0_H (1)	P2.0, upper
PP_Ch. 6/ 0_H **P2.1**	0.0 V	00_H	0_H (OFF)	2_H (8)	0_H (1)	P2.1, lower
	5.33V	FF_H	0_H (OFF)		0_H (1)	P2.1, upper
PP_Ch. 7/ 0_H **P2.2**	0.0 V	00_H	0_H (OFF)	2_H (8)	0_H (1)	P2.2, lower
	5.33 V	FF_H	0_H (OFF)		0_H (1)	P2.2, upper

1) MMODE of each channel is defined by sfr reset values: 0_H range control, 1_H under-voltage mode, 2_H over-voltage mode.
2) register: ADC1/2_TH_LO_CHX / ADC1/2_TH_UP_CHX
3) register: ADC1/2_HYST_LO_CHX / ADC1/2_HYST_UP_CHX; selectable decimal values [0, 4, 8, 16]
4) register: ADC1/2_FILTLEN_CHX
5) register: ADC1/2_CNT_LO_CHX / ADC1/2_CNT_UP_CHX; selectable decimal values [$2^0 2^1 ... 2^7$

25 High-Voltage Monitor Input

25.1 Features

- Four High-voltage inputs with VS/2 threshold voltage
- Wake capability for system stop mode and system sleep mode
- Edge sensitive wake-up feature configurable for transitions from low to high, high to low or both directions
- MON inputs can also be evaluated with ADC in Active Mode, using adjustable threshold values
- Selectable pull-up and pull-down current sources available

25.2 Introduction

This module is dedicated to monitor external voltage levels above or below a specified threshold. Each MONx pin can further be used to detect a wake-up event by detecting a level change by crossing the selected threshold. This applies to any power mode. Further more each MONx pin can be sampled by the ADC as analog input.

25.2.1 Block Diagram

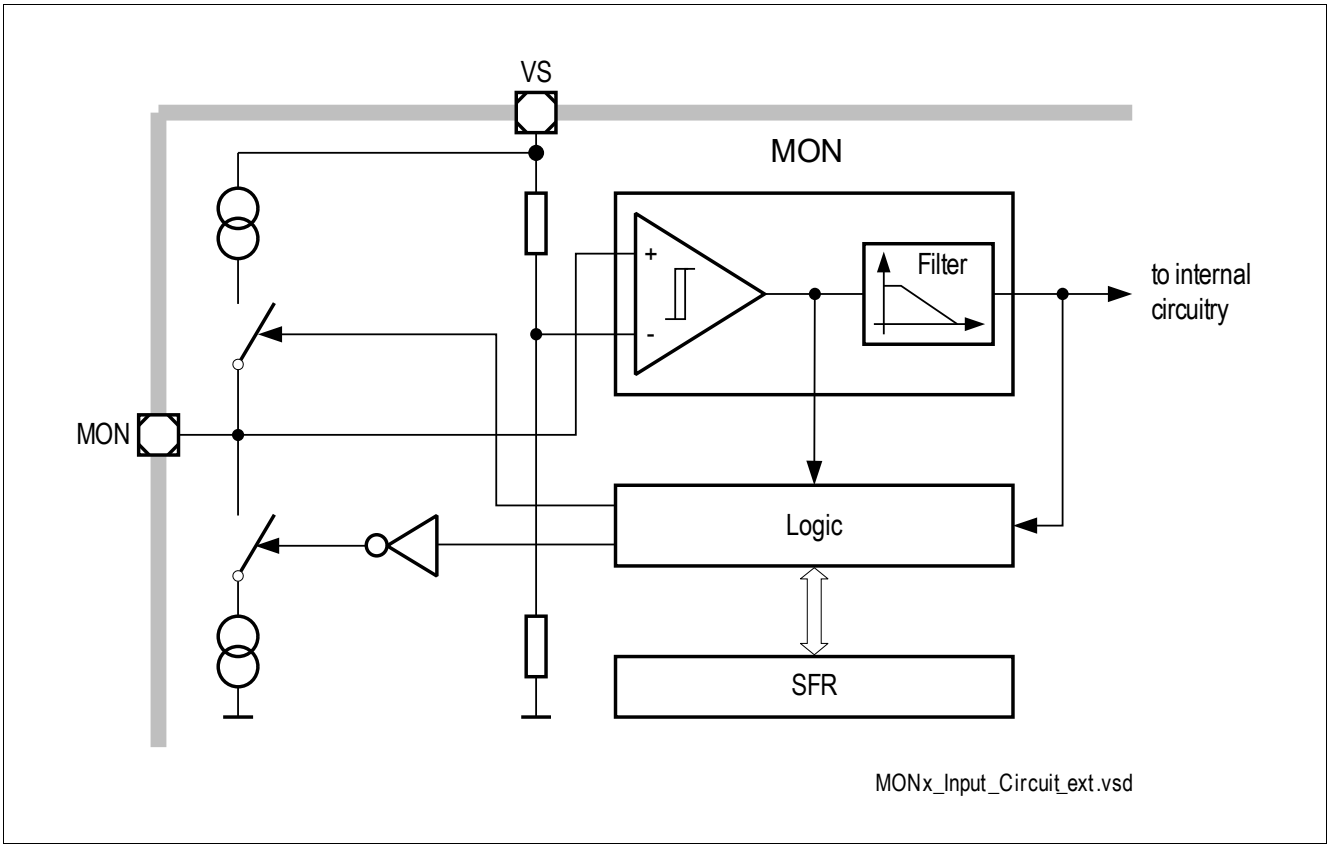


Figure 212 Monitoring Input Block Diagram

25.2.2 Functional Description

For a wake-up on a positive voltage transition, the **MONx_RISE** bit has to be configured. For a wake-up on a negative voltage transition, the corresponding bit **MONx_FALL** has to be set. This configuration can also be used for an edge detection in active mode.

As the system provides the functionality of cyclic sense, the MONx can be configured as a wake-up source for this mode. This is done by setting the bit **MONx_CYC**.

The MONx also includes an input circuit with pull-up (can be activated by **MONx_PU** bit) and pull-down (can be activated by **MONx_PD** Bit) current sources to define a certain voltage level with open inputs and a filter function to avoid wake-up events caused by unwanted voltage transients at the module input.

When automatic current source selection is enabled, a voltage level at the MONx input of $V_{MON\_th} < V_{MONx} < V_s$-1V activates the pull-up current source. If the MONx voltage is between 1 V < V_{MONx} < V_{MON_th} the pull-down sink is activated, providing stable levels at the monitor inputs. Below and above these voltage ranges the current is minimized to a leakage current. This automatic activation of the current sources, has to be done by setting **MONx_PU** and **MONx_PD** bit to one at the same time.

Note: ***In case a Monitoring Input is deactivated by setting bit MONx_EN to zero, it can neither be used as a wake-up source nor can it be used to detect logic levels!***

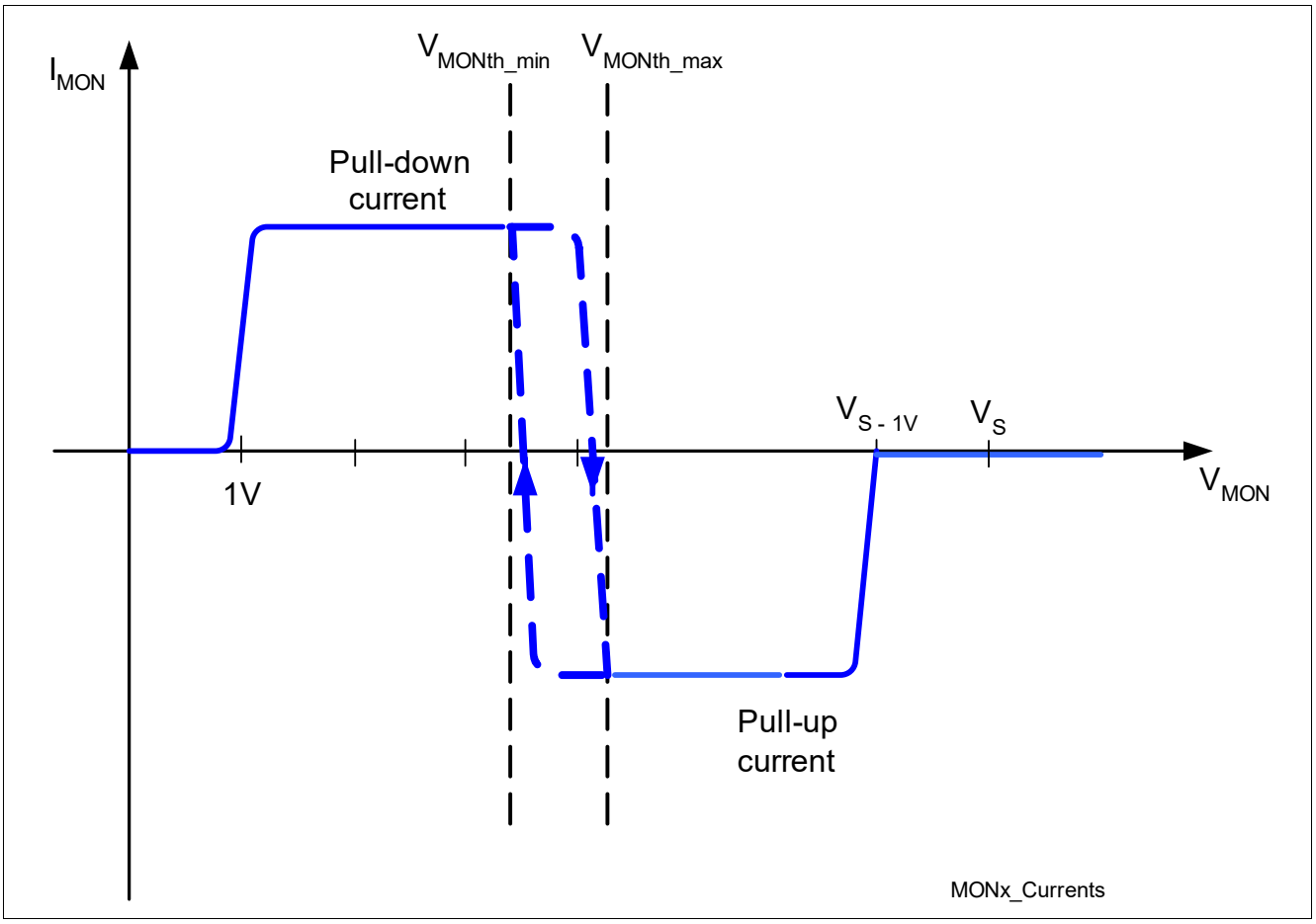


Figure 213 Module - HV_MON Input Characteristics for switchable pull current and static pull-down (on top) or pull-up

High-Voltage Monitor Input

The following tables provides an overview of the configuration possibilities on the MON_INs via **XSFR**.

Table 512 includes all pull-up and pull-down setup scenarios which can be chosen for one MONx. **Table 513** shows an overview of the available states of a MONx.

Table 512 Pull-Up / Pull-Down Input Current

MONx_PU	MONx_PD	Output Current	Description
0	0	leakage current[1)]	pull-up/down current source disabled
0	1	pull-down	pull-down current source enabled (for low active switches)
1	0	pull-up	pull-up current source enabled (for high active switches)
1	1	switchable[2)]	pull-up/down depending on input voltage

1) all current sources switched off.

2) will be automatically switched by the MONx circuit depending on level of input signal.

Table 513 MONx_EN MON Mode definition

MONx_EN	Mode	Description
0	disabled	Monitoring input is disabled (no wake-up possible!)
1	normal mode /power saving mode	Monitoring input is active during device Normal Mode Monitoring input automatically automatically enters power saving mode in device Sleep Mode and Stop Mode

25.3 Register Definition

This chapter describes the configuration registers for MON1-MON4.

Table 514 Register Address Space for PMU Registers

Module	Base Address	End Address	Note
PMU	50004000_H	$50004FFF_H$	Power Management Unit Registers

Table 515 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Register Definition, Monitor Input Registers			
PMU_MON_CNF1	Settings Monitor 1-4	034_H	$0707\ 0707_H$

The registers are addressed bytewise.

25.3.1 Monitor Input Registers

The monitor input registers are part of the PMU. This is due to the fact that this circuit requires supply (VDD1V5_PD_A) and clock, (LP_CLK) during system wide sleep and Stop modes.

Settings Monitor 1-4

PMU_MON_CNF1	Offset	Reset Value
Settings Monitor 1-4	034_H	see Table 516

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MON4_STS	RES	MON4_PU	MON4_PD	MON4_CYC	MON4_RI*	MON4_FA*	MON4_EN	MON3_STS	RES	MON3_PU	MON3_PD	MON3_CYC	MON3_RI*	MON3_FA*	MON3_EN
r	r	rw	rw	rw	rw	rw	rw	r	r	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MON2_STS	RES	MON2_PU	MON2_PD	MON2_CYC	MON2_RI*	MON2_FA*	MON2_EN	MON1_STS	RES	MON1_PU	MON1_PD	MON1_CYC	MON1_RI*	MON1_FA*	MON1_EN
r	r	rw	rw	rw	rw	rw	rw	r	r	rw	rw	rw	rw	rw	rw

Field	Bits	Type	Description
MON4_STS	31	r	**MON4 Status Input** *Note: the MON4 Status Input Bit is also updated in active mode of the device, when the HV MON4 input status changes. The user has to clear this flag before entering power saving modes otherwise the device will stay in active. Status is valid if MON4_EN is set and MON4_RISE or MON4_FALL is set.* 0_B **Low**, MON input has low status 1_B **High**, MON input has high status
RES	30	r	**Reserved** Always read as 1
MON4_PU	29	rw	**Pull-Up Current Source for MON4 Input Enable** *Note: Can only be enabled if MON4_EN is set.* 0_B **Disable**, Pull-up source disabled 1_B **Enable**, Pull-up source enabled
MON4_PD	28	rw	**Pull-Down Current Source for MON4 Input Enable** *Note: Can only be enabled if MON4_EN is set.* 0_B **Disable**, Pull-down source disabled 1_B **Enable**, Pull-down source enabled
MON4_CYC	27	rw	**MON4 for Cycle Sense Enable** *Note: Can only be enabled if MON4_EN is set. During reconfiguration of this bit, wake-up events can be lost.* 0_B **Disable**, Cycle Sense disabled 1_B **Enable**, Cycle Sense enabled
MON4_RISE	26	rw	**MON4 Wake-up on Rising Edge Enable** *Note: Can only be enabled if MON4_EN is set. During reconfiguration of this bit, wake-up events can be lost.* 0_B **Disable**, Wake-up disabled 1_B **Enable**, Wake-up enabled

Field	Bits	Type	Description
MON4_FALL	25	rw	**MON4 Wake-up on Falling Edge Enable** *Note: Can only be enabled if MON4_EN is set. During reconfiguration of this bit, wake-up events can be lost.* 0_B **Disable**, Wake-up disabled 1_B **Enable**, Wake-up enabled
MON4_EN	24	rw	**MON4 Enable** 0_B **Disable**, MON4 disabled 1_B **Enable**, MON4 enabled
MON3_STS	23	r	**MON3 Status Input** *Note: the MON3 Status Input Bit is also updated in active mode of the device, when the HV MON3 input status changes. The user has to clear this flag before entering power saving modes otherwise the device will stay in active. Status is valid if MON3_EN is set and MON3_RISE or MON3_FALL is set.* 0_B **Low**, MON input has low status 1_B **High**, MON input has high status
RES	22	r	**Reserved** Always read as 1
MON3_PU	21	rw	**Pull-Up Current Source for MON3 Input Enable** *Note: Can only be enabled if MON3_EN is set.* 0_B **Disable**, Pull-up source disabled 1_B **Enable**, Pull-up source enabled
MON3_PD	20	rw	**Pull-Down Current Source for MON3 Input Enable** *Note: Can only be enabled if MON3_EN is set.* 0_B **Disable**, Pull-down source disabled 1_B **Enable**, Pull-down source enabled
MON3_CYC	19	rw	**MON3 for Cycle Sense Enable** *Note: Can only be enabled if MON3_EN is set. During reconfiguration of this bit, wake-up events can be lost.* 0_B **Disable**, Cycle Sense disabled 1_B **Enable**, Cycle Sense enabled

Field	Bits	Type	Description
MON3_RISE	18	rw	**MON3 Wake-up on Rising Edge Enable** *Note: Can only be enabled if MON3_EN is set. During reconfiguration of this bit, wake-up events can be lost.* 0_B **Disable**, Wake-up disabled 1_B **Enable**, Wake-up enabled
MON3_FALL	17	rw	**MON3 Wake-up on Falling Edge Enable** *Note: Can only be enabled if MON3_EN is set. During reconfiguration of this bit, wake-up events can be lost.* 0_B **Disable**, Wake-up disabled 1_B **Enable**, Wake-up enabled
MON3_EN	16	rw	**MON3 Enable** 0_B **Disable**, MON3 disabled 1_B **Enable**, MON3 enabled
MON2_STS	15	r	**MON2 Status Input** *Note: the MON2 Status Input Bit is also updated in active mode of the device, when the HV MON2 input status changes. The user has to clear this flag before entering power saving modes otherwise the device will stay in active. Status is valid if MON2_EN is set and MON2_RISE or MON2_FALL is set.* 0_B **Low**, MON input has low status 1_B **High**, MON input has high status
RES	14	r	**Reserved** Always read as 1
MON2_PU	13	rw	**Pull-Up Current Source for MON2 Input Enable** *Note: Can only be enabled if MON2_EN is set.* 0_B **Disable**, Pull-up source disabled 1_B **Enable**, Pull-up source enabled
MON2_PD	12	rw	**Pull-Down Current Source for MON2 Input Enable** *Note: Can only be enabled if MON2_EN is set.* 0_B **Disable**, Pull-down source disabled 1_B **Enable**, Pull-down source enabled

Field	Bits	Type	Description
MON2_CYC	11	rw	**MON2 for Cycle Sense Enable** *Note: Can only be enabled if MON2_EN is set. During reconfiguration of this bit, wake-up events can be lost.* 0_B **Disable**, Cycle Sense disabled 1_B **Enable**, Cycle Sense enabled
MON2_RISE	10	rw	**MON2 Wake-up on Rising Edge Enable** *Note: Can only be enabled if MON2_EN is set. During reconfiguration of this bit, wake-up events can be lost.* 0_B **Disable**, Wake-up disabled 1_B **Enable**, Wake-up enabled
MON2_FALL	9	rw	**MON2 Wake-up on Falling Edge Enable** *Note: Can only be enabled if MON2_EN is set. During reconfiguration of this bit, wake-up events can be lost.* 0_B **Disable**, Wake-up disabled 1_B **Enable**, Wake-up enabled
MON2_EN	8	rw	**MON2 Enable** 0_B **Disable**, MON2 disabled 1_B **Enable**, MON2 enabled
MON1_STS	7	r	**MON1 Status Input** *Note: the MON1 Status Input Bit is also updated in active mode of the device, when the HV MON1 input status changes. The user has to clear this flag before entering power saving modes otherwise the device will stay in active. Status is valid if MON1_EN is set and MON1_RISE or MON1_FALL is set.* 0_B **Low**, MON input has low status 1_B **High**, MON input has high status
RES	6	r	**Reserved** Always read as 1
MON1_PU	5	rw	**Pull-Up Current Source for MON1 Input Enable** *Note: Can only be enabled if MON1_EN is set.* 0_B **Disable**, Pull-up source disabled 1_B **Enable**, Pull-up source enabled

Field	Bits	Type	Description
MON1_PD	4	rw	**Pull-Down Current Source for MON1 Input Enable** *Note: Can only be enabled if MON1_EN is set.* 0_B **Disable**, Pull-down source disabled 1_B **Enable**, Pull-down source enabled
MON1_CYC	3	rw	**MON1 for Cycle Sense Enable** *Note: Can only be enabled if MON1_EN is set. During reconfiguration of this bit, wake-up events can be lost.* 0_B **Disable**, Cycle Sense disabled 1_B **Enable**, Cycle Sense enabled
MON1_RISE	2	rw	**MON1 Wake-up on Rising Edge Enable** *Note: Can only be enabled if MON1_EN is set. During reconfiguration of this bit, wake-up events can be lost.* 0_B **Disable**, Wake-up disabled 1_B **Enable**, Wake-up enabled
MON1_FALL	1	rw	**MON1 Wake-up on Falling Edge Enable** *Note: Can only be enabled if MON1_EN is set. During reconfiguration of this bit, wake-up events can be lost.* 0_B **Disable**, Wake-up disabled 1_B **Enable**, Wake-up enabled
MON1_EN	0	rw	**MON1 Enable** 0_B **Disable**, MON1 disabled 1_B **Enable**, MON1 enabled

Table 516 RESET of PMU_MON_CNF1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_2	07070707_H	RESET_TYPE_2		

26 High-Side Switch

26.1 Features

The high-side switch is optimized for driving resistive loads. Only small line inductances are allowed. Typical applications are single or multiple LEDs of a dashboard, switch illumination or other loads that require a high-side switch.

A cyclic switch activation during Sleep Mode or Stop Mode of the system is also available.

Functional Features

- Multi purpose high-side switch for resistive load connections (only small line inductances are allowed)
- Overcurrent limitation
- Selectable current capability (25 mA/50 mA/100 mA/150 mA) by adjustable overcurrent detection with automatic shutdown
- Overtemperature detection and automatic shutdown
- Open load detection in on mode with open load current of typ. 1.4 mA
- Interrupt signalling of overcurrent, overtemperature and open load condition
- Cyclic switch activation in Sleep Mode and Stop Mode with cyclic sense support and reduced driver capability: max. 40 mA
- PWM capability up to 25 kHz
- Internal connection to System-PWM Generator (CCU6)
- Slew rate control for low EMI characteristic

Applications hints

- The voltage at HSx must not exceed the supply voltage by more than 0.3V to prevent a reverse current from HSx to VS.

26.2 Introduction

26.2.1 Block Diagram

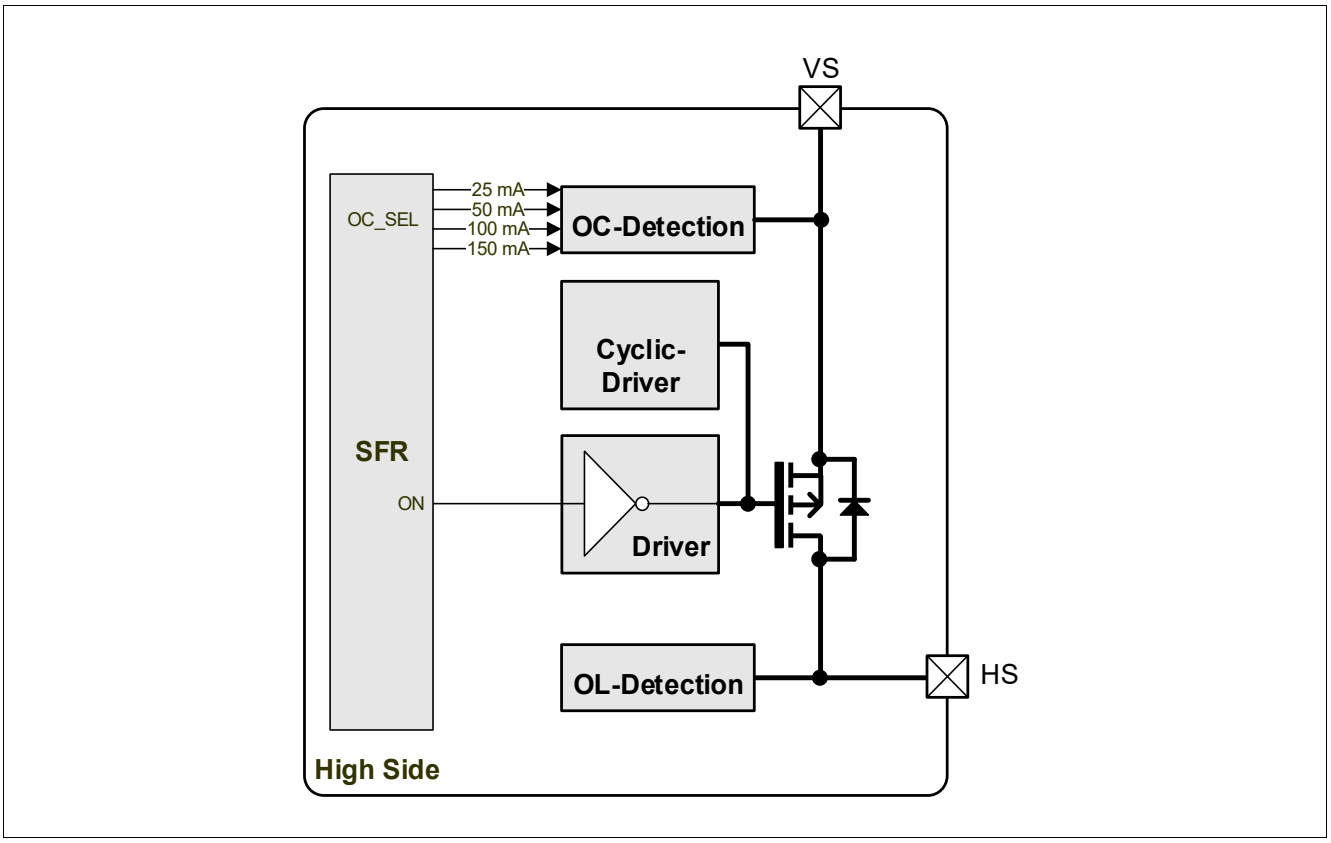


Figure 214 High-Side Module Block Diagram (incl. subblocks)

26.2.2 General

The high-side switch can generally be controlled in three different ways:

- In normal mode the output stage is fully controllable through the **SFR** Registers **HSx_CTRL**. Protection functions as overcurrent, overtemperature and open load detection are available.
- The PWM Mode can also be enabled by a **HSx_CTRL - SFR** bit. The PWM configuration has to be done in the corresponding PWM Module. All protection functions are also available in this mode. The maximum PWM frequency must not exceed 25 kHz (disabled slew rate control only).
- The high-side switch provides also the possibility of cyclic switch activation in all low power modes (Sleep Mode and Stop Mode). In this configuration it has limited functionality with limited current capability. Diagnostic functions are not available in this mode.

26.3 Functional Description

26.3.1 Normal Operation

In normal operation mode (CPU normal mode, CPU slow down mode) the high-side switch provides functionalities and protection functions which are:

- **selectable Slew Rate Control** for improved EMI behavior.
- **Overcurrent Detection** with four different thresholds (min.): 26 mA, 51 mA, 101 mA and 151 mA.
- **Overtemperature Protection**, to protect the switch against overtemperature.
- **On-State Open Load Detection** with threshold lower than 1.4 mA typ .

In device stop mode and device sleep mode, the high-side driver is switched off and disabled unless it is in cyclic sense mode. The user software does not need to take care about the proper power down sequence of this module. This is done by hardware.

In stop mode the configuration of the driver is kept inside the corresponding sfrs. If the driver was switched on before entering stop mode, after a wakeup its status is restored automatically.

26.3.1.1 Slew Rate Configuration

The high-side switch provides 3 slew rate configuration possibilities:

- slow, 3V/µs (up to 5 kHz PWM frequency).
- fast, 40V/µs (above 5 kHz PWM frequency).
- low EMC, 1V/µs (for low EMC emissions).

The configuration can be done by flag HSx_SR_CTRL_SEL. The slew rate configuration is also taken for the pwm mode.

26.3.1.2 Overcurrent Detection

To configure the proper overcurrent threshold the corresponding bits **HSx_OC_SEL** in the **HSx_CTRL - SFR** have to be set. If an overcurrent condition is present, the high-side switch will be automatically turned off. In parallel the flag **HSx_OC_IS** is set and the HSx_ON flag and HSx_PWM flag are cleared. To enable the high-side switch again, it is recommended to clear the **HSx_OC_IS** flag and then set the **HSx_ON** bit to reactivate the switch. Clearing only the **HSx_OC_IS** flag would not turn the switch automatically on. If the overcurrent condition is still present, the switch will be disabled once again.

26.3.1.3 Overtemperature Detection

If overtemperature condition appears, the switch will shutdown and the corresponding bit **HSx_OT_STS** is set. To reenable the high-side switch, the same procedure as for the overcurrent condition has to be applied. Due to the fact that overtemperature condition is removed very slowly (device has to cool down) in comparison to the CPU time base, it is recommended to clear the status flag and to check if it is set again immediately after clearing, before trying to switch the driver on again.

26.3.1.4 ON-State Open Load Detection

The high-side open load detection in ON State is mainly performed by the overcurrent detection and its fixed threshold of typ. 1.4 mA. If the current flowing through the output stage of the high-side switch falls below the value of typ. 1.4 mA, the corresponding status flag **OL_STS** is set. The open load detection has no influence on operation of the high-side switch.

The open load condition will cause an interrupt if enabled by the user.

26.3.2 PWM Operation

In PWM mode the high-side switch has to be first enabled by the corresponding bit in the **HSx_CTRL** register. The related bit is described below.

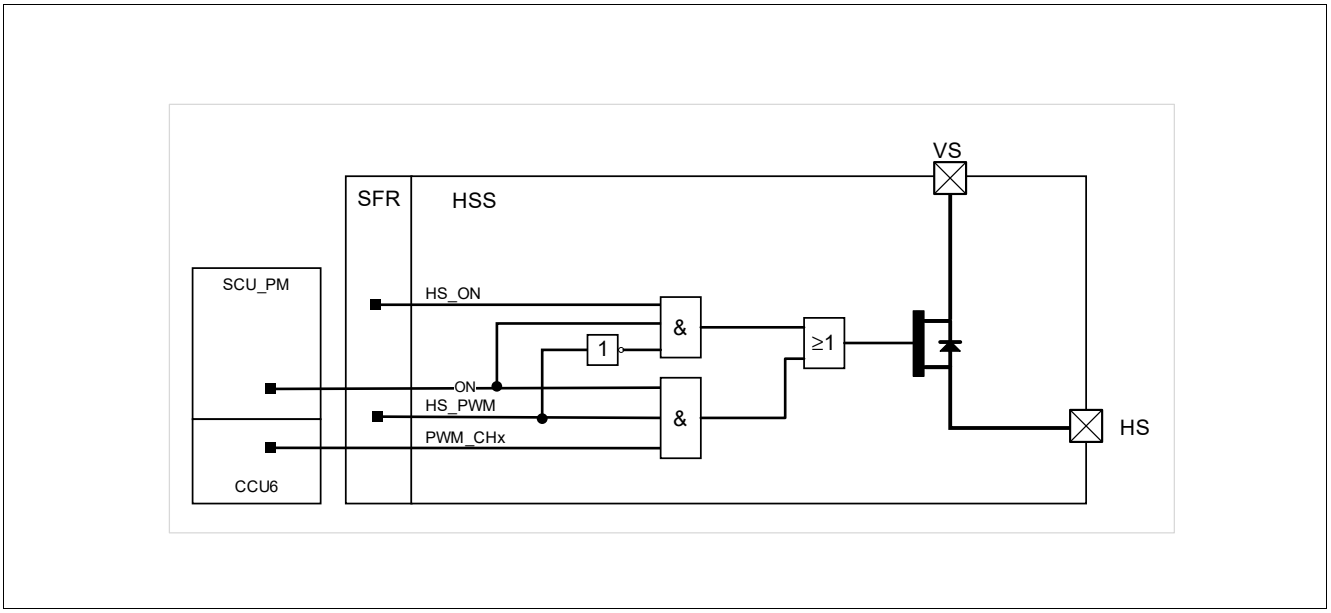


Figure 215 Combinatorial Control of High-Side Switch in PWM Mode

To avoid any output glitches on the HSx output, the **HSx_PWM** bit should be set first. After the function is enabled for PWM operation the corresponding PWM unit can be enabled.

For frequencies higher than 5 kHz, the slew rate setting has to be set to 30V/µs. Otherwise the internal power dissipation of the switch might damage the device.

In PWM mode all protection functions are available.

26.3.3 Cyclic Switching in Low Power Mode

In the cyclic sense power-saving mode the high-side switch cyclically supplies an external switch arrangement for a short time (see PMU_SLEEP.CYC_SENSE_S_DEL), just long enough to detect the position of the switch. The configuration procedure to use the high-side switch for cyclic sense operation is described in the chapter **Power Management Unit**.

26.4 Register Definition

This chapter describes all necessary registers to control the high-side module and monitor its operation status.

Table 517 Register Address Space

Module	Base Address	End Address	Note
HS	40024000_H	$40027FFF_H$	High-Side Switch

Table 518 Register Overview

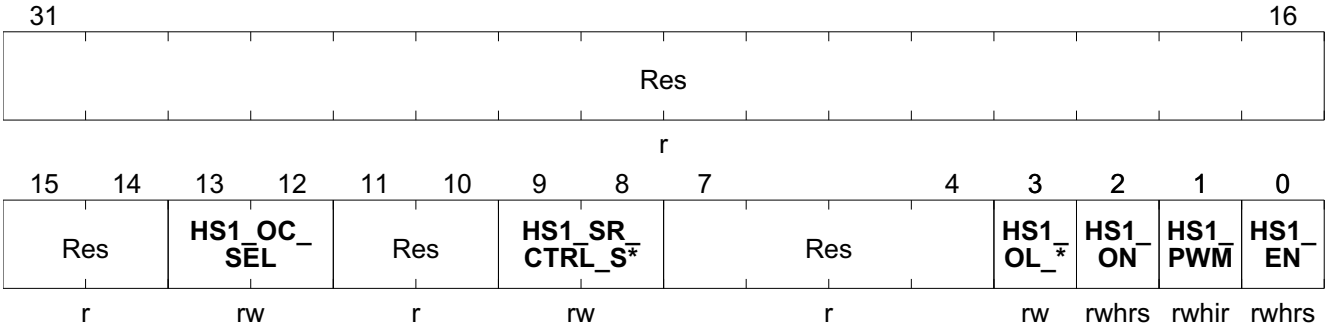
Register Short Name	Register Long Name	Offset Address	Reset Value
Register Definition, High-Side Switch Register			
HS_CTRL	High-Side Driver Control	04_H	$0000\ 3000_H$
HS_IRQS	High-Side Driver Interrupt Status	08_H	$0000\ 0000_H$
HS_IRQCLR	High-Side Driver Interrupt Status Clear Register	$0C_H$	$0000\ 0000_H$
HS_IRQEN	High-Side Driver Interrupt Enable Register	10_H	$0000\ 0000_H$
HS_PWMSRCSEL	High-Side PWM Source Selection Register	24_H	$0000\ 0000_H$
HS_TRIM	High-Side Driver 1 TRIM	$1C_H$	$0000\ 0000_H$

The registers are addressed bytewise.

26.4.1 High-Side Switch Register

High-Side Control Register

HS_CTRL	Offset	Reset Value
High Side Driver Control	**04_H**	**see Table 519**



Field	Bits	Type	Description
Res	31:14	r	**Reserved** Always read as 0
HS1_OC_SEL	13:12	rw	**High Side 1 Overcurrent Threshold Selection** 0_H **IOCTH0**, 26 mA min. 1_H **IOCTH1**, 51 mA min. 2_H **IOCTH2**, 101 mA min. 3_H **IOCTH3**, 151 mA min.
Res	11:10	r	**Reserved** Always read as 0
HS1_SR_CTRL_SEL	9:8	rw	**High Side 1 Slew Rate Control select** 00_B **Slew Rate 1**, Slow Slew Rate 3V/µs is enabled 01_B **Slew Rate 2**, Fast Slew Rate 40V/µs is enabled 10_B **Slew Rate 3**, Low EMC Slew Rate 1V/µs is enabled (for low EMC emissions) 11_B **Slew Rate 3**, Low EMC Slew Rate 1V/µs is enabled (for low EMC emissions)
Res	7:4	r	**Reserved** Always read as 0
HS1_OL_EN	3	rw	**High Side 1 Open Load Detection Enable** 0_B **DISABLE**, disable open load detection 1_B **ENABLE**, enable open load detection
HS1_ON	2	rwhrs	**High Side 1 On** 0_B **OFF**, HS driver off 1_B **ON**, HS driver on
HS1_PWM	1	rwhir	**High Side 1 PWM Enable** *Note: this flag has higher priority then HS1_ON* 0_B **DISABLE**, disables control by PWM input 1_B **ENABLE**, enables control by PWM input
HS1_EN	0	rwhrs	**High Side 1 Enable** 0_B **DISABLE**, HS circuit power off 1_B **ENABLE**, HS circuit power on

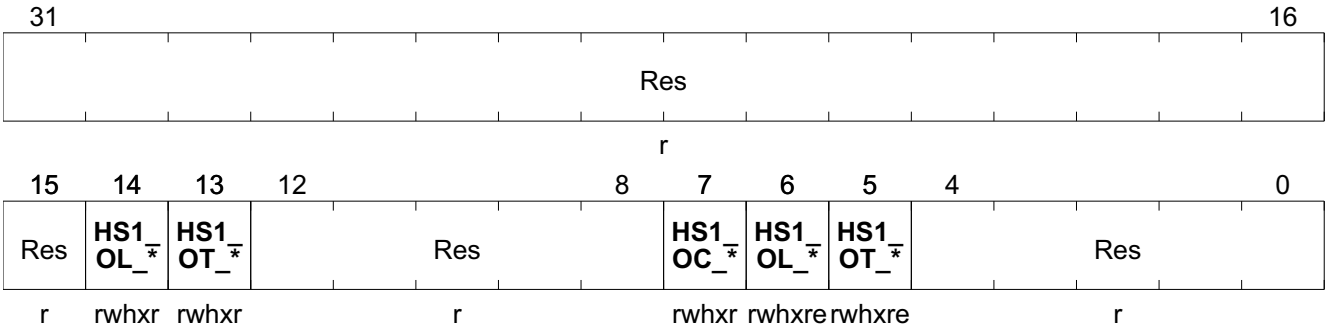
Table 519 RESET of HS_CTRL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00003000_H	RESET_TYPE_3		

High-Side Interrupt Status Register

HS_IRQS	Offset	Reset Value
High Side Driver Interrupt Status	08_H	see Table 520

High-Side Switch



Field	Bits	Type	Description
Res	31:15	r	**Reserved** Always read as 0
HS1_OL_STS	14	rwhxr	**High Side 1 Open Load Status** 0_B **no Open Load**, no open load Condition occurred. 1_B **Open Load**, open load occurred; switch is not automatically shutdown. Write sets status.
HS1_OT_STS	13	rwhxr	**High Side 1 Overtemperature Status** 0_B **no Overtemperature** , no overtemperature occurred. 1_B **Overtemperature**, overtemperature occurred; switch is automatically shutdown. Write sets status.
Res	12:8	r	**Reserved** Always read as 0
HS1_OC_IS	7	rwhxr	**High Side 1 Overcurrent Interrupt Status** 0_B **no Overcurrent**, no overcurrent Condition occurred. 1_B **Overcurrent**, overcurrent occurred; switch is automatically shutdown. Write sets status.
HS1_OL_IS	6	rwhxre	**High Side 1 Open Load Interrupt Status** 0_B **NORMAL**, normal load 1_B **OPEN LOAD**, open load detected, write sets status
HS1_OT_IS	5	rwhxre	**High Side 1 Overtemperature Interrupt Status** 0_B **no Overtemperature** , no overtemperature occurred. 1_B **Overtemperature**, overtemperature occurred; switch is automatically shutdown. Write sets status
Res	4:0	r	**Reserved** Always read as 0

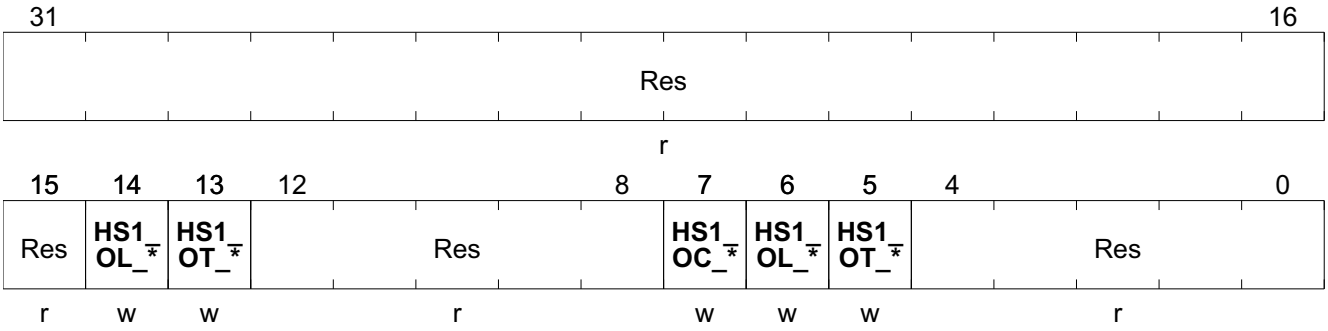
Table 520 RESET of HS_IRQS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

High-Side Switch

High-Side Interrupt Status Register Clear Register

HS_IRQCLR	Offset	Reset Value
High Side Driver Interrupt Status Clear Register	$0C_H$	see Table 521



Field	Bits	Type	Description
Res	31:15	r	**Reserved** Always read as 0
HS1_OL_SC	14	w	**High Side 1 Open Load Status Clear** 0_B **no Clear**, 1_B **Clear**,
HS1_OT_SC	13	w	**High Side 1 Overtemperature Status Clear** 0_B **no Clear** , 1_B **Clear**,
Res	12:8	r	**Reserved** Always read as 0
HS1_OC_ISC	7	w	**High Side 1 Overcurrent Interrupt Status Clear** 0_B **no Clear**, 1_B **Clear**,
HS1_OL_ISC	6	w	**High Side 1 Open Load Interrupt Status Clear** 0_B **no Clear**, 1_B **Clear**,
HS1_OT_ISC	5	w	**High Side 1 Overtemperature Interrupt Status Clear** 0_B **no Clear** , 1_B **Clear**,
Res	4:0	r	**Reserved** Always read as 0

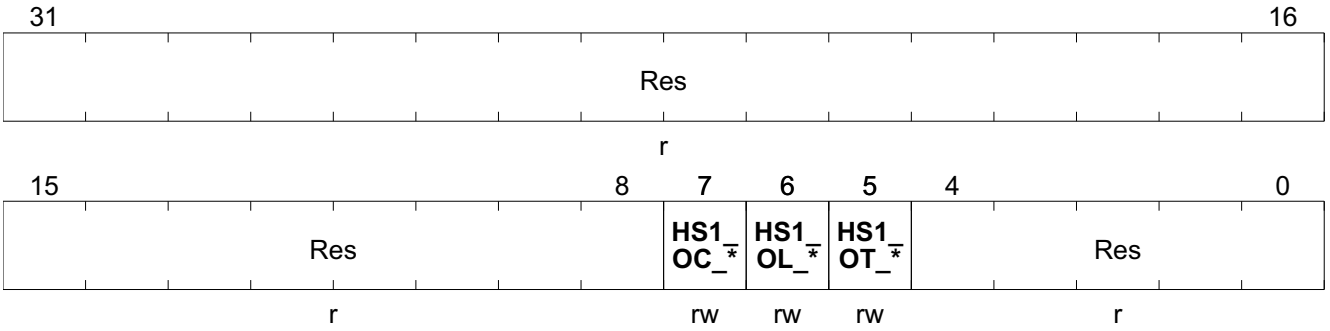
Table 521 RESET of HS_IRQCLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

High-Side Switch

High-Side Interrupt Enable Register

HS_IRQEN	Offset	Reset Value
High Side Driver Interrupt Enable Register	10_H	see Table 522



Field	Bits	Type	Description
Res	31:8	r	**Reserved** Always read as 0
HS1_OC_IEN	7	rw	**High Side 1 Overcurrent Interrupt Enable** 0_B **disable**, 1_B **enable**,
HS1_OL_IEN	6	rw	**High Side 1 Open Load Interrupt Enable** 0_B **disable**, 1_B **enable**,
HS1_OT_IEN	5	rw	**High Side 1 Overtemperature Interrupt Enable** 0_B **disabled**, 1_B **enable**,
Res	4:0	r	**Reserved** Always read as 0

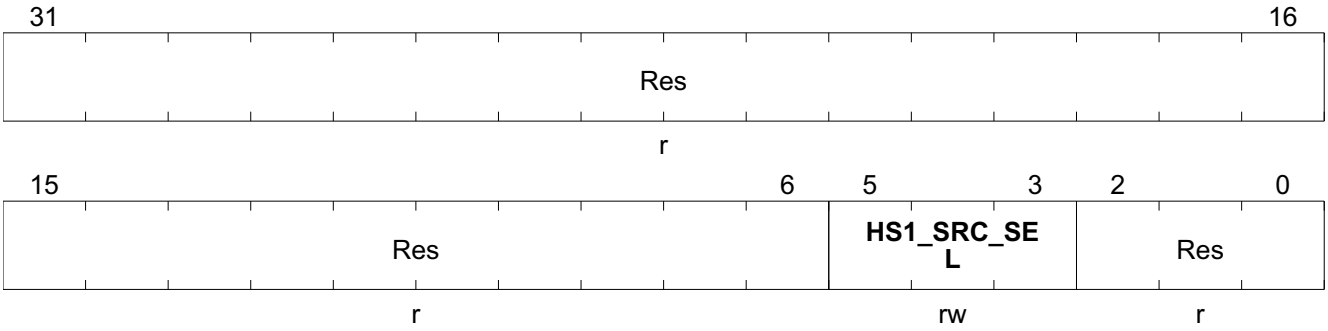
Table 522 RESET of HS_IRQEN

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

High-Side PWM Source Selection Register

HS_PWMSRCSEL	Offset	Reset Value
High Side PWM Source Selection Register	24_H	see Table 523

High-Side Switch



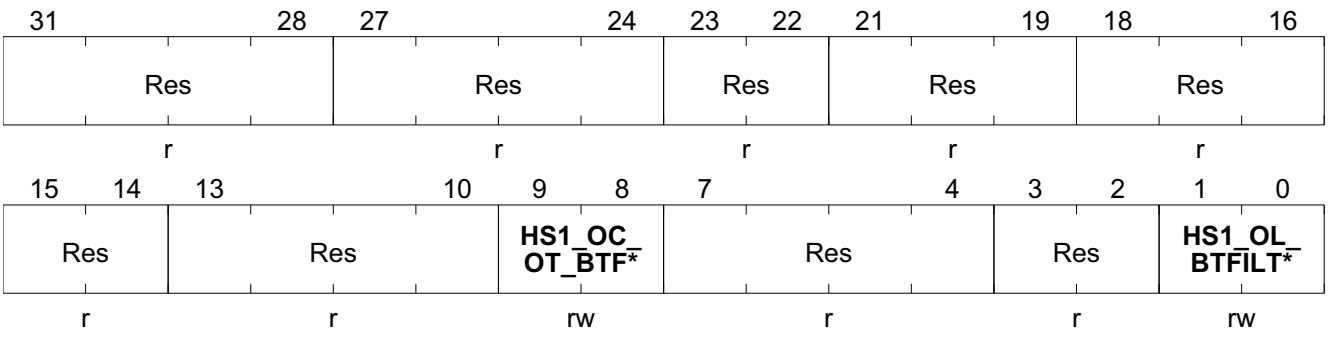
Field	Bits	Type	Description
Res	31:6	r	**Reserved** Always read as 0
HS1_SRC_SEL	5:3	rw	**HS1 PWM Source Selection** **Note: Can only be written when HS_CTRL.HS1_PWM = 0** 000_B **CC60**, PWM output of CCU6 001_B **CC61**, PWM output of CCU6 010_B **CC62**, PWM output of CCU6 011_B **COUT60**, PWM output of CCU6 100_B **COUT61**, PWM output of CCU6 101_B **COUT62**, PWM output of CCU6 110_B **T3OUT**, PWM output of GPT12 111_B **rfu**, same as 110_B
Res	2:0	r	**Reserved** Always read as 0

Table 523 RESET of HS_PWMSRCSEL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

High-Side 1 TRIM Register

HS_TRIM	Offset	Reset Value
High Side Driver 1 TRIM	$1C_H$	see Table 524



Field	Bits	Type	Description
Res	31:28	r	**Reserved** Always read as 0
Res	27:24	r	**Reserved** Always read as 0
Res	23:22	r	**Reserved** Always read as 0
Res	21:19	r	**Reserved** Always read as 0
Res	18:16	r	**Reserved** Always read as 0
Res	15:14	r	**Reserved** Always read as 0
Res	13:10	r	**Reserved** Always read as 0
HS1_OC_OT_BTFILT_SEL	9:8	rw	**Blanking Time Filter Select for HS1 overcurrent / overtemperature detection** 00_B **4_us**, 4 µs filter time 01_B **8_us**, 8 µs filter time 10_B **16_us**, 16 µs filter time 11_B **32_us**, 32 µs filter time
Res	7:4	r	**Reserved** Always read as 0
Res	3:2	r	**Reserved** Always read as 0
HS1_OL_BTFILT_SEL	1:0	rw	**Blanking Time Filter Select for HS1 open Load detection** 00_B **4_us**, 4 µs filter time 01_B **8_us**, 8 µs filter time 10_B **16_us**, 16 µs filter time 11_B **32_us**, 32 µs filter time

Table 524 RESET of HS_TRIM

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	$CAC00001_H$	RESET_TYPE_4		
TRIM_1	$CAC00001_H$	TRIM		

26.5 Interrupt Generation - and Status Bit Logic

The interrupt flags of the high-side module have the following behaviour:

Overcurrent detection: the overcurrent detection interrupt flag is a level sensitive interrupt flag. This flag is set when the overcurrent condition occurs and stays persistent as long as the condition is present.

Overtemperature detection: the overtemperature detection interrupt flag is an edge sensitive interrupt flag. This flag is set when the overtemperature condition occurs, but can be cleared immediately. The

overtemperature status of the overtemperature condition can then still be monitored in the dedicated status register, which is placed in the same interrupt status register.

Open Load detection: the open load detection interrupt flag is an edge sensitive interrupt flag. This flag is set when the open condition occurs, but can be cleared immediately. The open load status of the open load condition can then still be monitored in the dedicated status register, which is placed in the same interrupt status register.

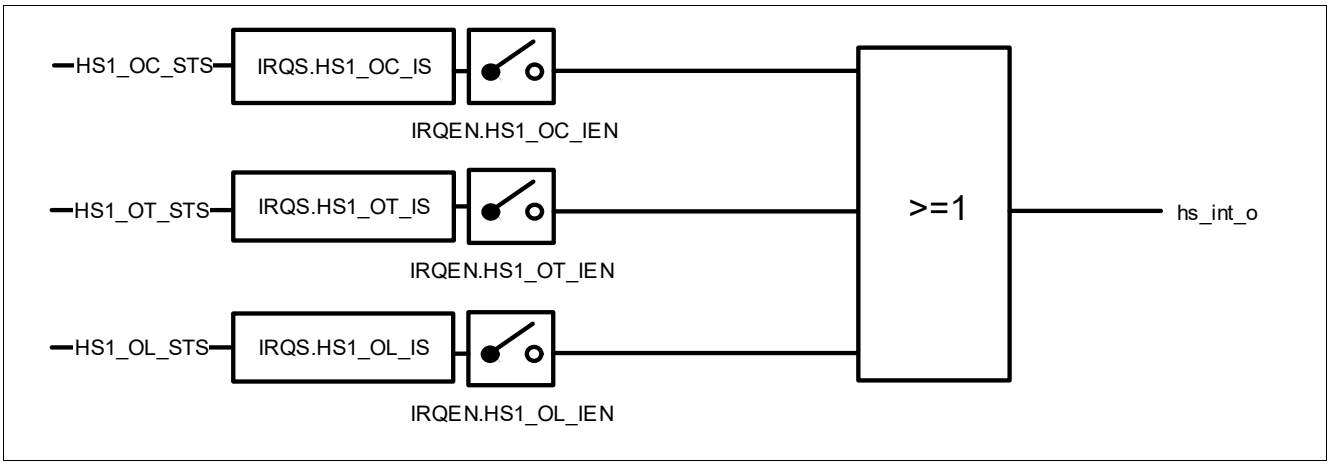


Figure 216 High-Side Switch Interrupt Generation

26.6 Application Information

If the high-side module is used as offboard pin the following external circuitry is mandatory:

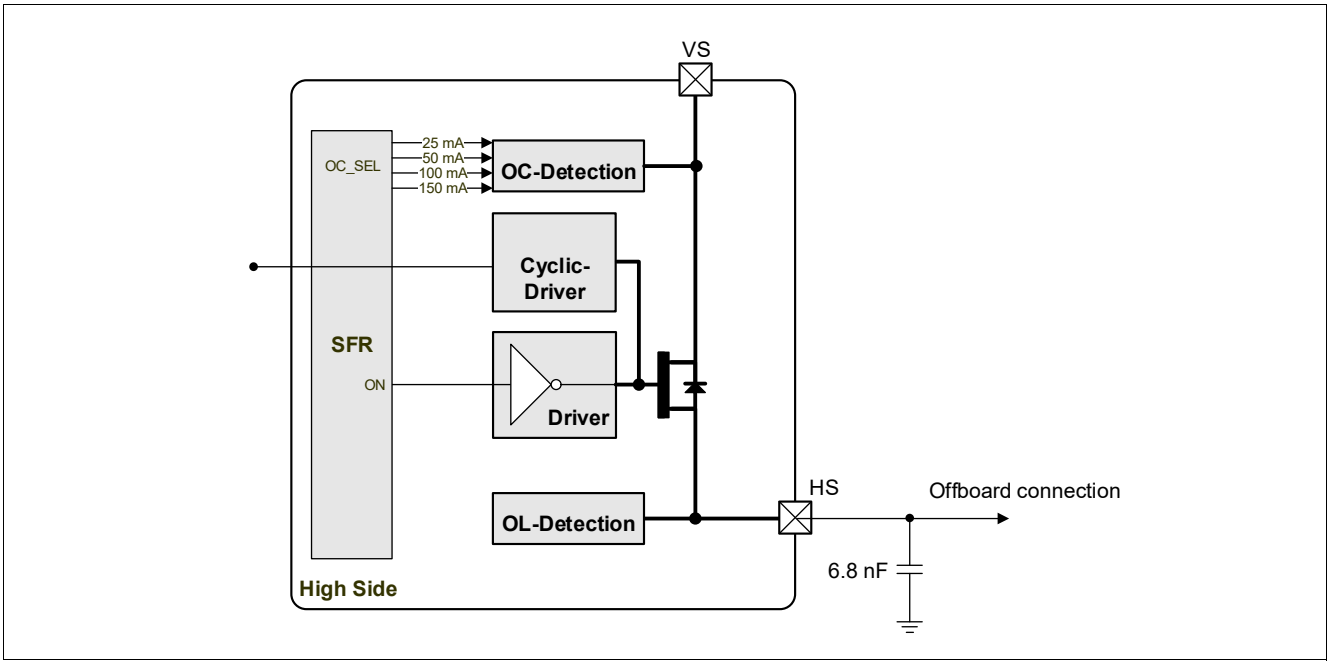


Figure 217 Circuitry Mandatory for use as Offboard Pin

If the high-side module is used as offboard pin a 6.8 nF is needed as buffer capacitor.

27 Bridge Driver (incl. Charge Pump)

27.1 Features

The Bridge Driver is intended to drive external normal-level MOSFETs in bridge configuration and provides many diagnostic possibilities to detect faults.

Functional Features

- **Flexible control** by SFRs of Bridge Driver module or PWM output signals of CCU6 module
- **Current-driven output stages** to control external n-channel MOSFET gates with flexibly programmable gate current profile
- **Adjustable cross-conduction protection**
- **High-current discharge mode** to reduce dead times and to keep external MOSFETs off during fast transients
- **Passive pull-down mode** to keep external MOSFETs off if the Bridge Driver is disabled
- **Brake mode** with reduced current consumption to statically switch on external MOSFETs
- **Hold mode** with low current consumption to switch on external low-side MOSFETs if the Bridge Driver is disabled
- **Timing measurements** of on/off delays and on/off slope durations
- **Adaptive control mode** with automatic adjustment of gate current values
- **Integrated 2-stage charge pump** for low-voltage operation and statical MOSFET gate control
- **Adjustable voltage monitoring** of Bridge Driver supply voltage (VSD) and charge pump output voltage (VCP)
- **Adjustable short-circuit detection** in on and off state
- **Open-load detection** in off state
- **Overtemperature** detection and shutdown

27.2 Introduction

27.2.1 Block Diagram

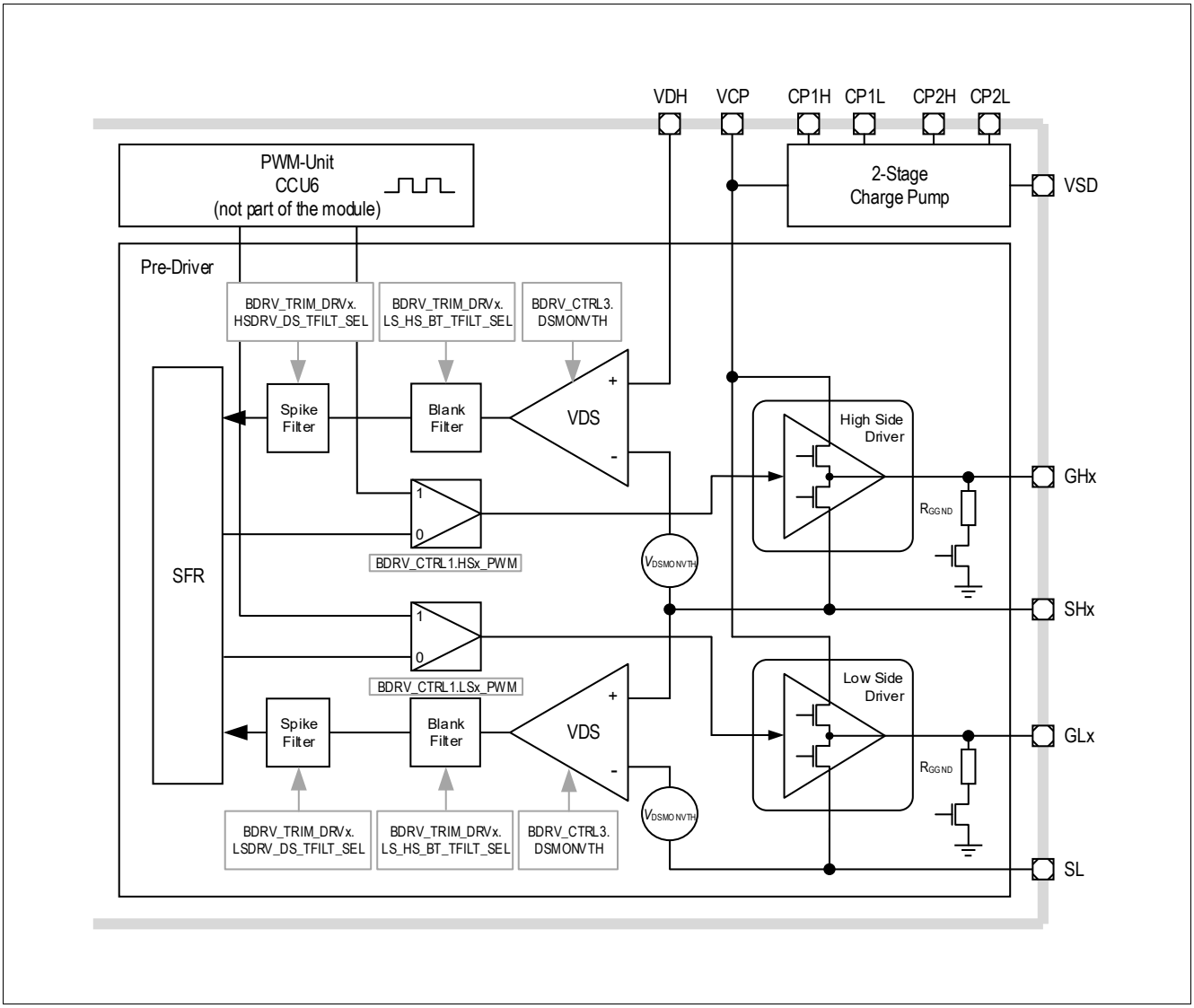


Figure 218 Driver Module Block Diagram (incl. System Connections)

27.2.2 Flexible Control

Each gate driver of the Bridge Driver module can be switched on and off in two different ways:

- Static Mode: The gate drivers are statically switched on or off by the Bridge Driver module SFRs.
- PWM Mode: The gate drivers are PWM-controlled by the System PWM Module (CCU6). The interconnection from the CCU6 output channels to the gate drivers is set up by the Bridge Driver module SFRs.

In both modes all diagnostic and protection functions (short-circuit, open-load, and overtemperature detection) are available.

27.2.3 Current-Driven Output Stages

The Bridge Driver output stages generate source and sink currents to charge and discharge the gates of the external n-channel MOSFETs. The gate current values are programmable to vary the slew rate at the bridge output.

27.2.3.1 Overview

Figure 219 shows an overview of one switching cylce of an external MOSFET.

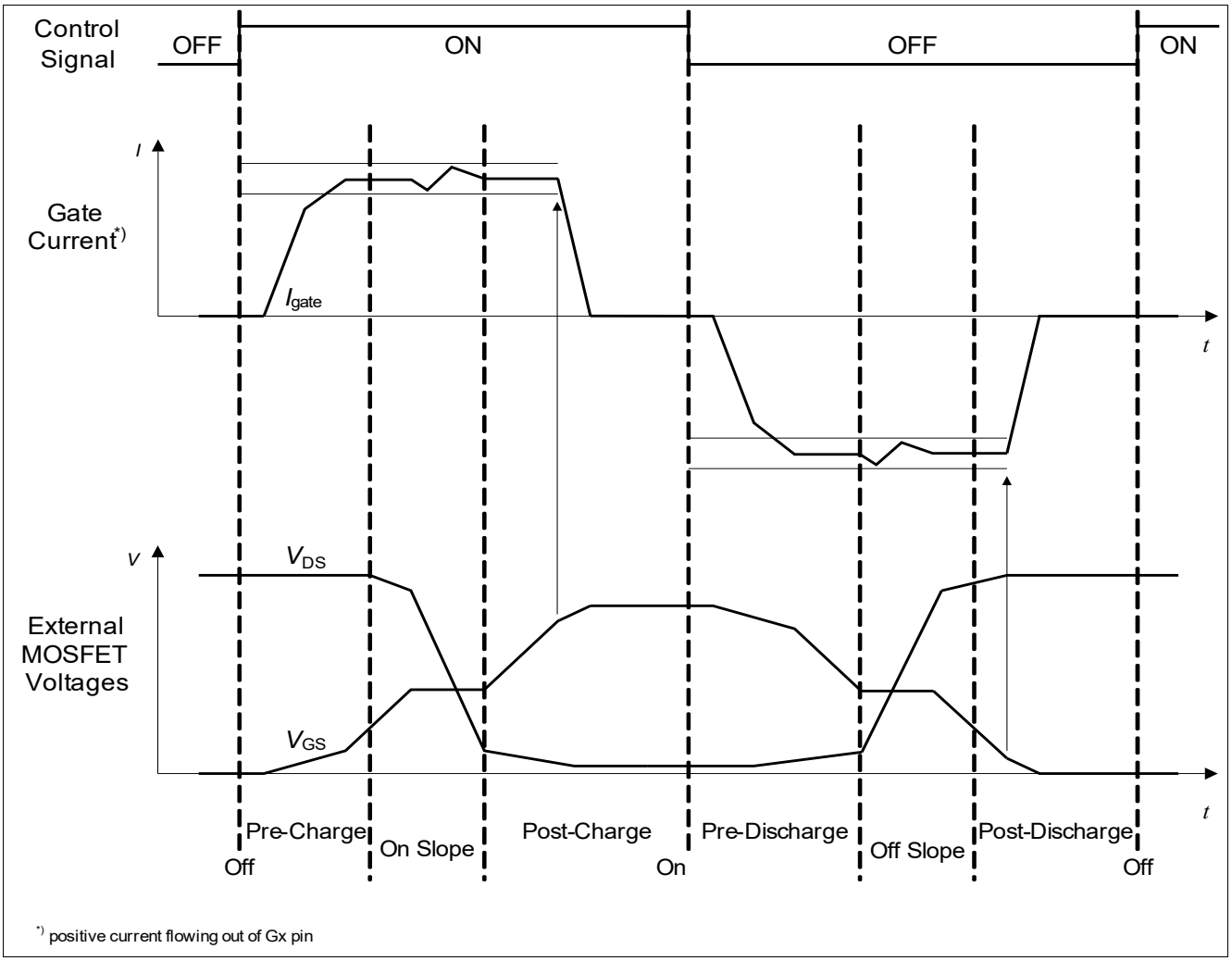


Figure 219 Switching cycle

The control input signal sets the gate driver either in charge or discharge mode, i.e. it generates a source current flowing out of the driver to charge the gate of an external MOSFET or a sink current flowing into the driver to discharge the gate of an external MOSFET.

Based on the changes on the drain-to-source voltage of the external MOSFET the charging and discharging phases can each be divided into three subphases.

Subphases of the charging phase:

- pre-charge subphase: the gate of the external MOSFET is pre-charged without change on V_{DS}; the external MOSFET is still off
- on slope subphase: the gate of the external MOSFET is further charged while the external MOSFET turns on and generates the on slope at V_{DS}

- post-charge subphase: the gate of the external MOSFET is post-charged until the maximum V_{GS} the gate driver is able to provide; the external MOSFET is on and its $R_{DS(on)}$ decreases to its minimum value

Subphases of the discharging phase:

- pre-discharge subphase: the gate of the external MOSFET is pre-discharged without significant change on V_{DS}; the external MOSFET is still on, but its $R_{DS(on)}$ increases
- off slope subphase: the gate of the external MOSFET is further discharged while the external MOSFET turns off and generates the off slope at V_{DS}
- post-discharge subphase: the gate of the external MOSFET is post-charged until V_{GS} is equal to 0V; the external MOSFET is off and is kept off

27.2.3.2 Switch-On

Figure 220 shows the detailed behavior of the gate driver output stage in the switch-on phase and the corresponding electrical characteristic parameters.

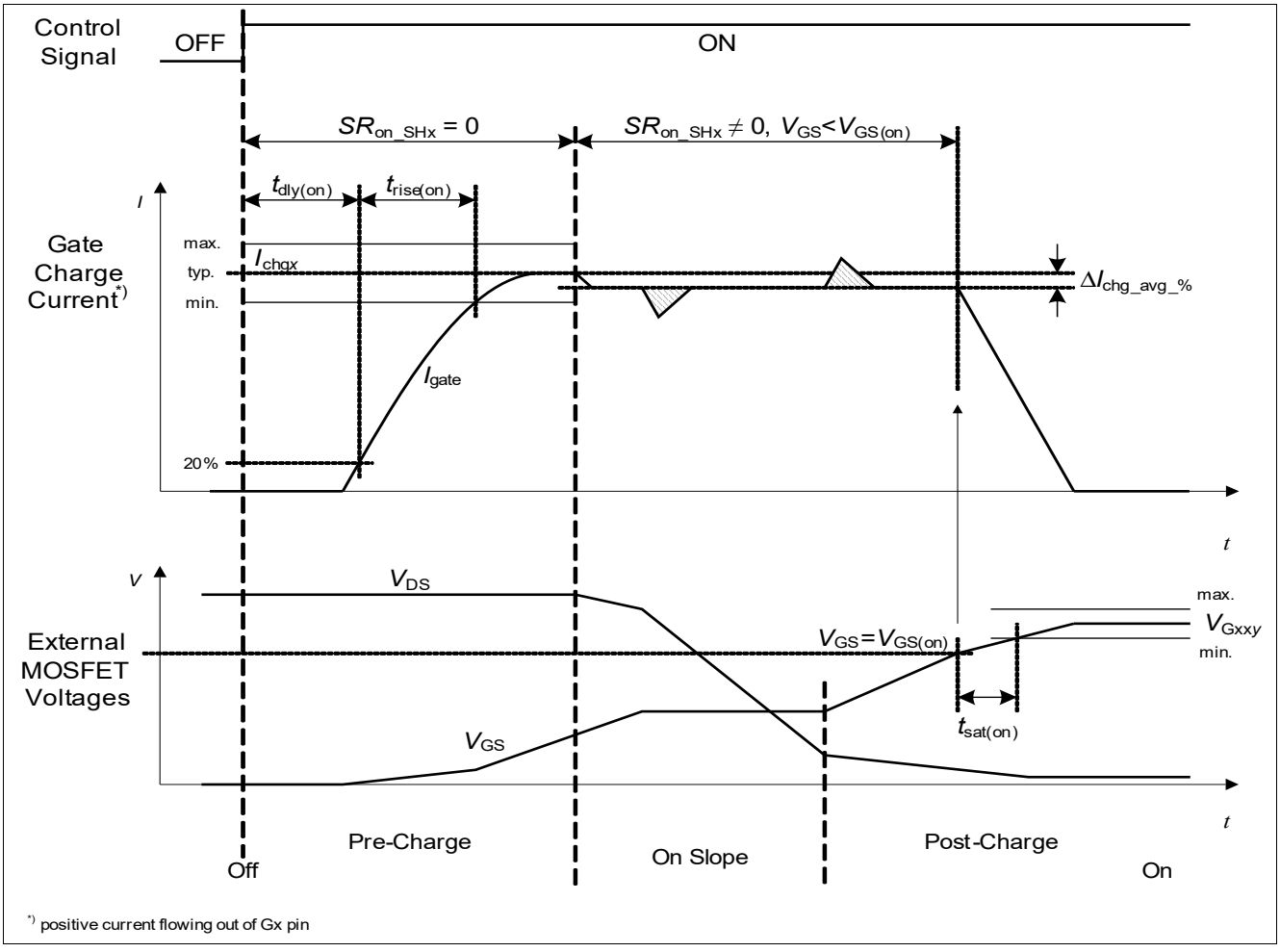


Figure 220 Detailed behavior of the gate driver output stage in the switch-on phase

After an initial turn-on delay time $t_{dly(on)}$ the gate charge current I_{gate} rises and after additional $t_{rise(on)}$ reaches its specified minimum limit $I_{chgx,min}$ and stays stable until the gate-to-source voltage of the external MOSFET reaches $V_{GS} = V_{GS(on)}$. During the slope at the corresponding SHx pin (i.e. $SR_{on\_SHx} \neq 0$) the average gate current deviates less than $\Delta I_{chg\_avg\_\%}$ from the original set point I_{chgx}. The gate of the external MOSFET is further charged to the high-level output voltage of the gate driver V_{Gxxy}. The time from exceeding $V_{GS} = V_{GS(on)}$ and reaching $V_{Gxxy,min}$ is defined by $t_{sat(on)}$.

27.2.3.3 Switch-Off

Figure 221 shows the detailed behavior of the gate driver output stage in the switch-off phase and the corresponding electrical characteristic parameters.

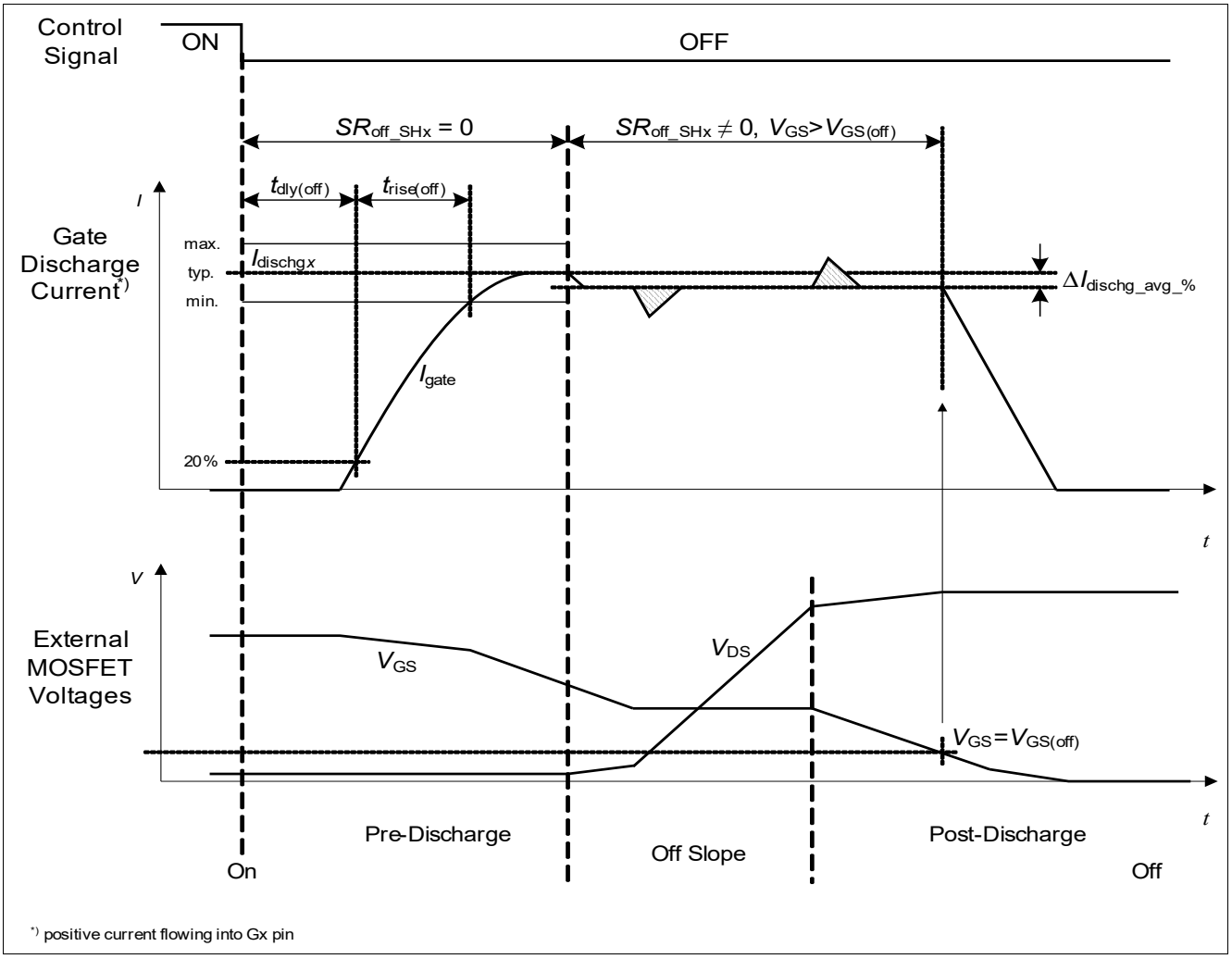


Figure 221 Detailed behavior of the gate driver output stage in the switch-off phase

After an initial turn-off delay time $t_{dly(off)}$ the gate discharge current I_{gate} rises and reaches its specified minimum limit $I_{dischgx,min}$ after $t_{rise(off)}$ and stays stable until the gate-to-source voltage of the external MOSFET reaches $V_{GS} = V_{GS(off)}$. During the slope at the corresponding SHx pin (i.e. $SR_{off\_SHx} \neq 0$) the average gate discharge current deviates less than $\Delta I_{dischg\_avg\_\%}$ from the original set point $I_{dischgx}$.

27.2.3.4 Control Modes

There are two basic modes to program the gate current set point values of the output stages: constant mode and sequencer mode.

- In constant mode a simple gate current profile is defined by SFRs where the gate charging phase and the gate discharging phase each have two current set point values and one duration value. The second current set point value (“clamping value”) remains valid until the driver changes from charge mode to discharge mode or vice versa and is intended to statically keep on or off the external MOSFET at a reduced gate current level to be robust against external shorts at the gate pin.

- In sequencer mode an advanced gate current profile is defined by SFRs where the gate charging phase and the gate discharging phase each are split into consecutive sub phases with individual current set point values and duration values (see **Figure 222**):
  - For the gate charging phase 5 current set point values ($i_{x(on)}$) and 4 duration values ($t_{x(on)}$) are defined by SFRs.
  - The fifth current setpoint value $I_{5(on)}$ (“clamping value”) remains valid until the driver changes to discharge mode. This charge current is intended to statically keep on the external MOSFET (e.g. driving an external R_{GS}) at a reduced gate current level to be robust against external gate-to-source shorts.
  - For the gate discharging phase 5 current set point values ($i_{x(off)}$) and 4 duration values ($t_{x(off)}$) are defined by SFRs.
  - The fifth current setpoint value $I_{5(off)}$ (“clamping value”) remains valid until the driver changes to charge mode. This discharge current is intended to statically keep off the external MOSFET (e.g. during fast voltage transients or EMI) at a reduced gate current level to be robust against external gate-to-drain shorts.
  - At the transition between two gate current set points the actual gate driver output current settles within $t_{set(seq)}$ to the new gate current set point (see **Figure 223**).

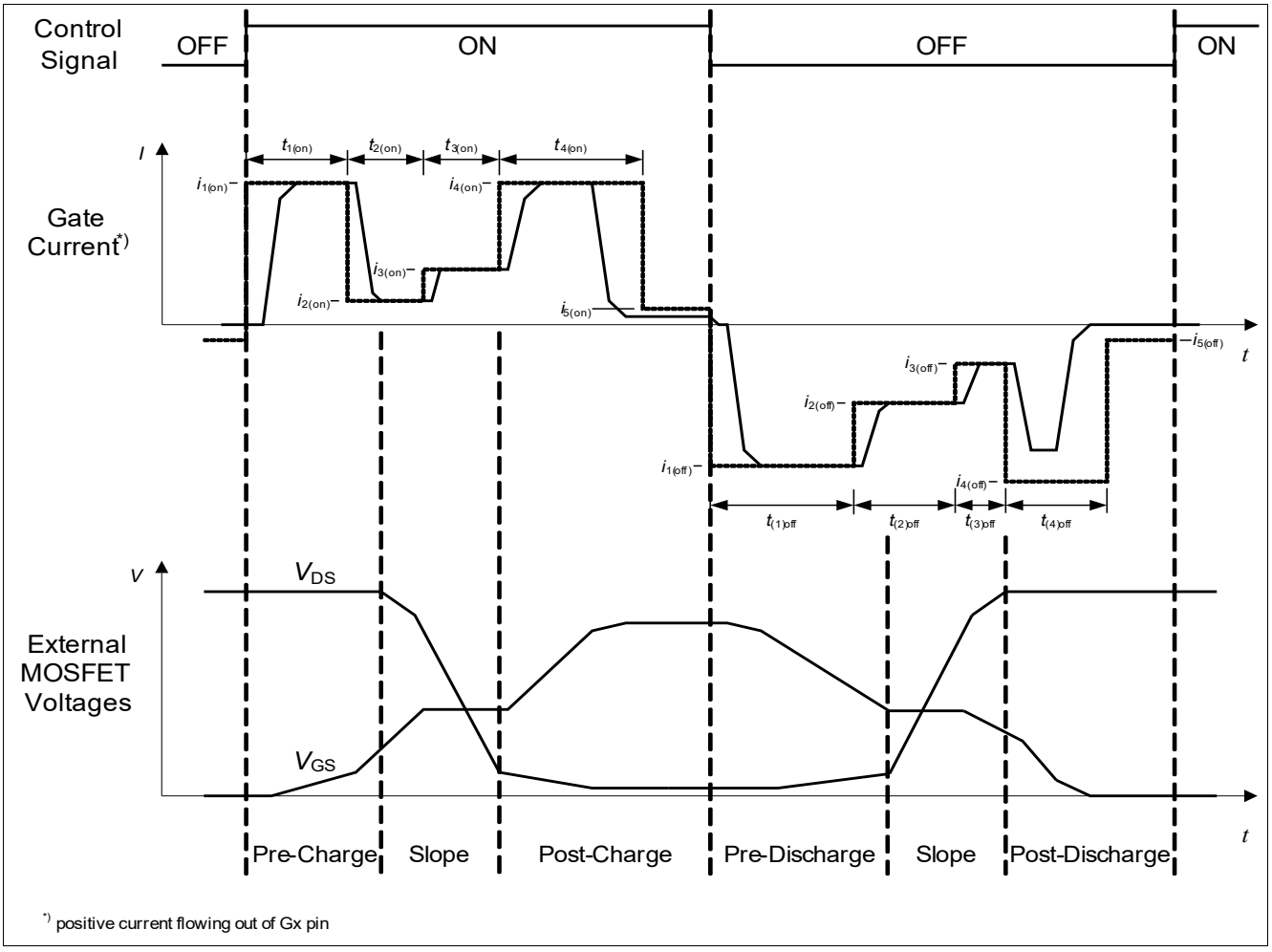


Figure 222 Gate current set point values generated by the sequencer

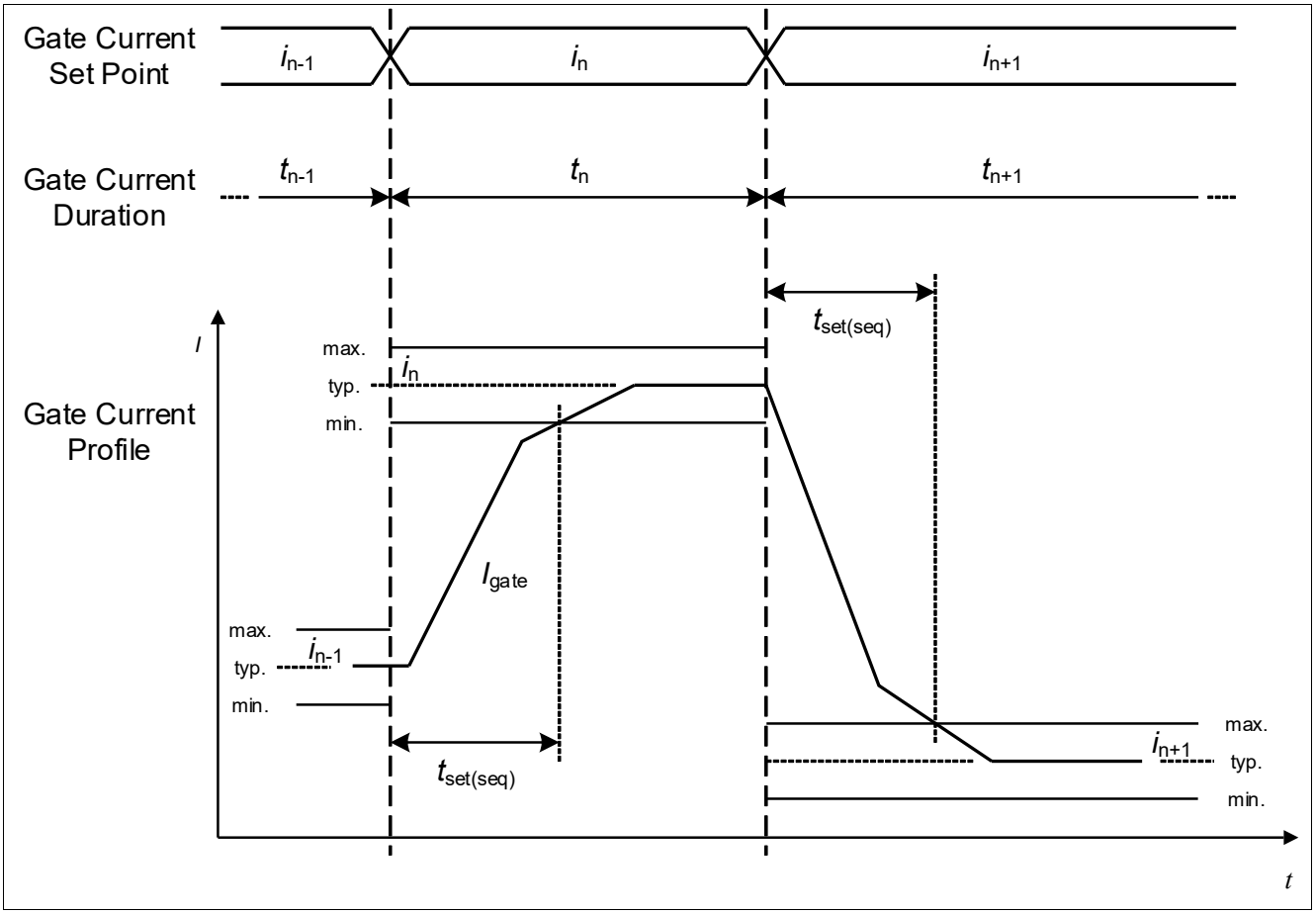


Figure 223 Gate current settling time

27.2.4 Adjustable Cross-Conduction Protection

The Bridge Driver protects half bridges of external MOSFETs against cross conduction. After switching off one of the MOSFETs of a half bridge the complementary MOSFET cannot be switched on for an optionally programmable time defined by SFRs.

27.2.5 High-Current Discharge Mode

The high-current discharge mode provides a low-ohmic path between the Gx and Sx pins to do a fast discharge of the external MOSFET gate and keep the external MOSFET off during fast voltage transients at its drain or source terminals.

The high-current discharge mode is activated in the following situations:

- in the case of an emergency shutdown after detection of an error condition,
- if the complementary external MOSFET is switched on to avoid cross conduction in the external half bridge.

If the adjustable cross-conduction protection feature is enabled the high-current discharge mode is delayed and activated at the same time than the switch-on control signal of the complementary MOSFET.

27.2.6 Passive Pull-Down Mode

If the Bridge Driver module is disabled the passive pull-down mode activates resistors R_{GGND} between the Gx pins and ground to passively keep discharged the gates of the external MOSFETs. During normal operation these pull-down resistors are switched off.

27.2.7 Brake Mode

In Brake Mode either both external high-side MOSFETs or both external low-side MOSFETs are statically switched on to short circuit the motor coil to brake the motor or keep it actively blocked during standstill. Since in brake mode no PWM capability is needed the charge pump is set into low-power mode to reduce the current consumption I_{VSD_BK} from the VSD pin.

27.2.8 Hold Mode

In Hold Mode the external low-side MOSFETs can be switched to an auxiliary gate voltage V_{Gxx_HM} to terminate the motor pins in all cases where the Bridge Driver and its charge pump is disabled (including stop mode and sleep mode where the Bridge Driver is disabled by default). This leads to low current consumption I_{VSD_SMHM} and I_{VSD_STPMHM} from the VSD pin. The Hold Mode is configured by SFRs in the Power Management Unit where the behavior during stop or sleep mode is defined. The configuration includes the channel-individual selection between static and cyclic activation of the Hold mode and programmable timing.

Note: In Hold Mode the monitoring and protection of the Bridge Driver is not available.

27.2.9 Timing Measurements

The Bridge Driver provides fast comparators with low propagation delay t_{cdly} at the SHx pins to measure on and off delays $t_{sdly(on)}$ and $t_{sdly(off)}$ between changes on the control signals and the corresponding slopes at the SHx pins. Additionally, these comparators are able to measure the on and off slope durations $t_{sdur(on)}$ and $t_{sdur(off)}$ at the SHx pins. The measured values are stored in SFRs for further evaluation by software or by the adaptive control mode (see **Chapter 27.2.10**).

Figure 224 shows the thresholds $V_{SH(high)}$ and $V_{SH(low)}$ and propagation delay t_{cdly} of the fast comparators and the measured slope timing parameters $t_{sdly(on)}$, $t_{sdur(on)}$, $t_{sdly(off)}$, and $t_{sdur(off)}$ during PWM actuation of the external low-side MOSFET.

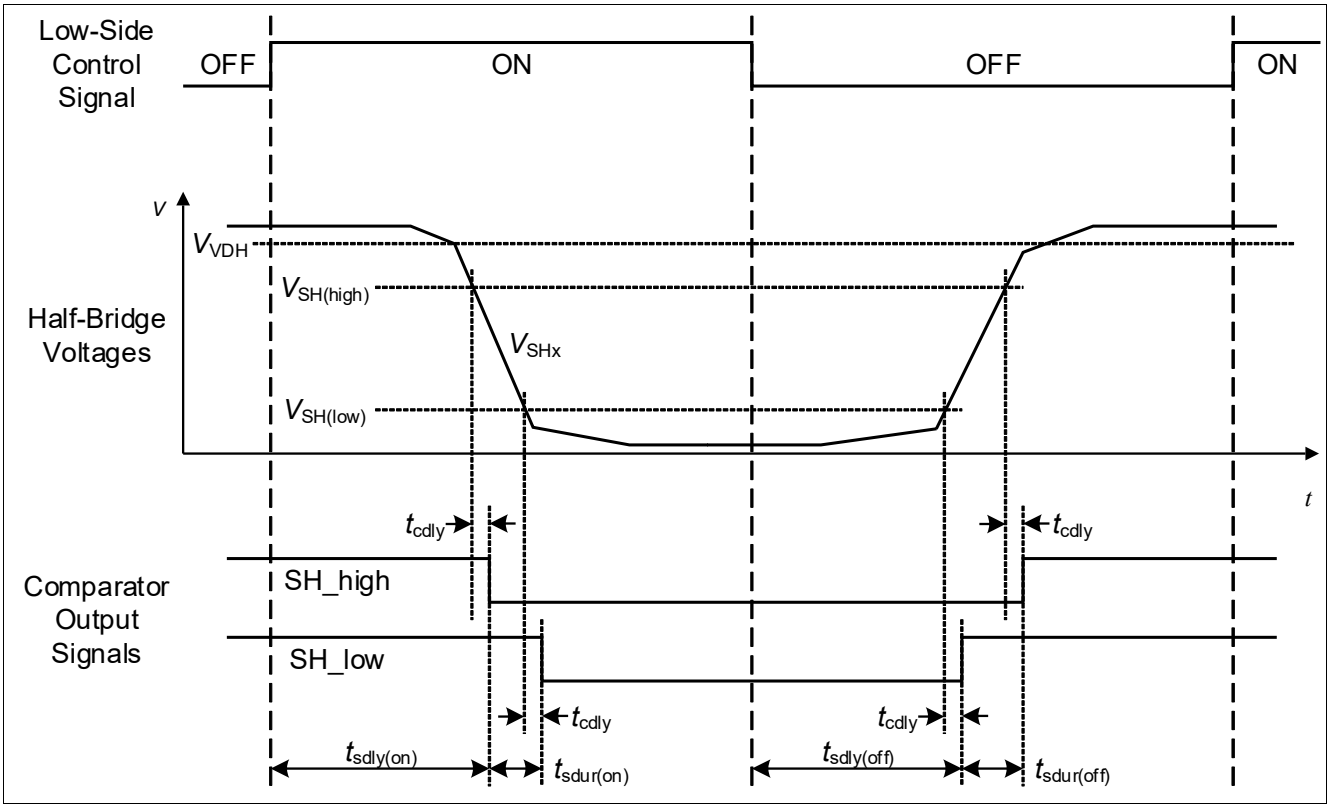


Figure 224 Comparator thresholds and timing parameters during low-side PWM

Figure 225 shows the thresholds $V_{SH(high)}$ and $V_{SH(low)}$ and propagation delay t_{cdly} of the fast comparators and the measured slope timing parameters $t_{sdly(on)}$, $t_{sdur(on)}$, $t_{sdly(off)}$, and $t_{sdur(off)}$ during PWM actuation of the external high-side MOSFET.

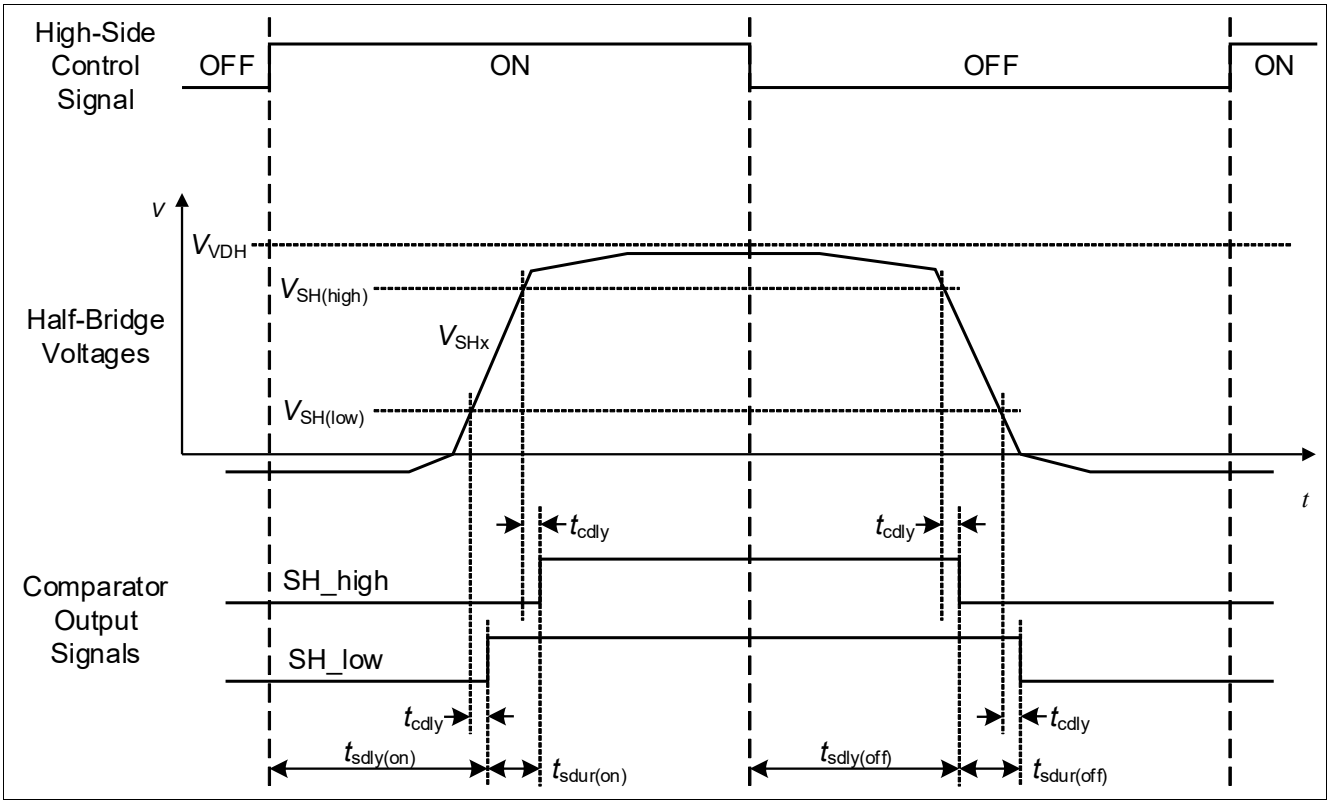


Figure 225 Comparator thresholds and timing parameters during high-side PWM

For plausibility checks of the fast comparators and the assigned timing measurement counters there is an additional channel on/off delay measurement counter which can be switched to each channel's control signals and the corresponding drain-to-source voltage monitoring comparator outputs.

27.2.10 Adaptive Control Mode

The Bridge Driver provides an optional adaptive control mode if the output stages are controlled by the sequencer mode. The target values for on and off delays are defined by SFRs. In order to reach these target values, the adaptive control algorithm reads the results from the timing measurement and adjusts the current set point of the first gate charging sub phase $i_{1(on)}$ and the first gate discharging sub phase $i_{1(off)}$ accordingly.

27.2.11 Integrated 2-Stage Charge Pump

The Bridge Driver is supplied by an integrated 2-stage charge pump which provides a stable voltage V_{CP} above the battery voltage. This enables the Bridge Driver to operate down to low battery voltage values and to statically switch on the external MOSFETs.

The charge pump output voltage is programmable by SFRs. The charge pump frequency is continously varied between two boundary frequencies defined by SFRs.

27.2.12 Adjustable Voltage Monitoring

The supply voltages of the Bridge Driver (VSD and VCP) are monitored by the Measurement Unit. The Bridge Driver including the charge pump can be optionally disabled at undervoltage or overvoltage of the monitored signals.

27.2.13 Adjustable Short Circuit Detection

For short circuit detection the drain-to-source comparators of the Bridge Driver are used to compare the voltage drops across the external MOSFETs to the programmable threshold voltage $V_{DSMONVTH}$. During transitions from off to on and vice versa the comparator output signals are ignored for a programmable blank time defined by SFRs.

In on state the external MOSFETs are switched off automatically if a stable short-circuit condition is detected for a SFR programmable filter time and an interrupt is generated. It can be selected by SFR if all MOSFETs are switched off or only the one where the short-circuit condition was detected.

In off state the motor phases can be pulled up or pulled down by the diagnostic currents I_{PUDiag} and I_{PDDiag}. The drain-to-source comparator output signals can be read by SFRs to check if the motor phase voltages change according to the activated diagnostic currents.

27.2.14 Open-Load Detection

For open-load detection in off state the pull-up diagnostic current I_{PUDiag} of one half bridge and the pull-down diagnostic current I_{PDDiag} of the other half bridge are activated. The pull-down diagnostic current I_{PDDiag} is able to overdrive the pull-up diagnostic current I_{PUDiag} (by I_{PDDiag_OD}). Therefore, in the case of a connected motor, both motor phase voltages are pulled-down. In the case of an disconnected motor, one motor phase voltage is pulled down while the other is pulled up according to the diagnostic current settings. The reaction of the motor phase voltages can be checked by the drain-to-source comparators of the Bridge Driver and their corresponding SFR status bits.

27.2.15 Overtemperature

The temperature of the Bridge Driver Charge Pump is monitored by a dedicated temperature sensor of the Measurement Unit for temperature warning signalling and overtemperature shutdown of the Bridge Driver.

27.3 Functional Description

27.3.1 Flexible Control

The source of the gate driver control signals are configured by the register **BDRV_CTRL1**. The two basic control modes “static mode” and “PWM mode” are selected by the bits LSx_PWM and HSx_PWM. In static mode the gate drivers are controlled by the bits LSx_ON or HSx_ON. In PWM mode the gate drivers are controlled by the CCU6 output signals. The assignment between CCU6 channels and gate drivers is done by the register **BDRV_PWMSRCSEL**.

Note: For PWM control all 4 gate drivers must be set to PWM mode. Static control of one of the half bridges must be done by proper CCU6 settings (e.g. by duty cycle = 0 or 100%).

*Note: The default/reset settings for the BDRV_PWMSRCSEL register assign all gate drivers to the same CCU6 channel. This leads to simultaneous control of low-side and high-side gate drivers and will be refused by the cross-conduction protection. The recommended settings can be found in **Table 525**.*

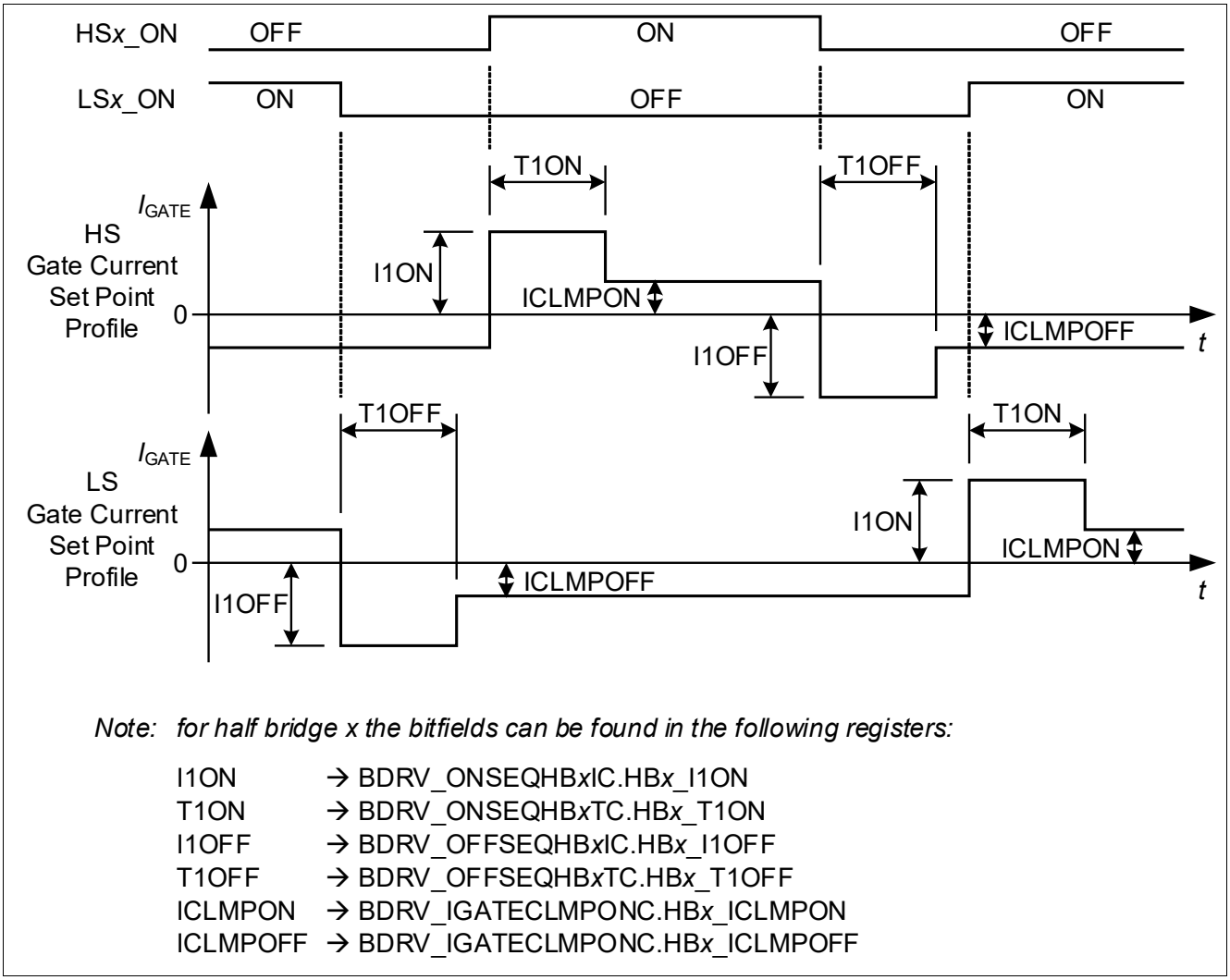
Table 525 Recommended settings for the BDRV_PWMSRCSEL register

BDRV_PWMSRCSEL bit field	Default/reset settings	Recommended settings
LS1_SRC_SEL	00_B: CC60	10_B: COUT60
LS2_SRC_SEL	00_B: CC60	11_B: COUT61
HS1_SRC_SEL	00_B: CC60	00_B: CC60
HS2_SRC_SEL	00_B: CC60	01_B: CC61

27.3.2 Current-Driven Output Stages

The charge and discharge currents of the gate drivers can be programmed. There are two modes, “constant mode” and “sequencer mode”, which are selected for each half bridge and individually for on and off phases by the bits HBxONSEQCNF and HBxOFFSEQCNF in the register **BDRV_CTRL2**.

The gate current set point values in constant mode are configured as shown in **Figure 226**.



Note: for half bridge x the bitfields can be found in the following registers:

I1ON	→ BDRV_ONSEQHBxIC.HBx_I1ON
T1ON	→ BDRV_ONSEQHBxTC.HBx_T1ON
I1OFF	→ BDRV_OFFSEQHBxIC.HBx_I1OFF
T1OFF	→ BDRV_OFFSEQHBxTC.HBx_T1OFF
ICLMPON	→ BDRV_IGATECLMPONC.HBx_ICLMPON
ICLMPOFF	→ BDRV_IGATECLMPONC.HBx_ICLMPOFF

Figure 226 Gate current set point profile in constant mode

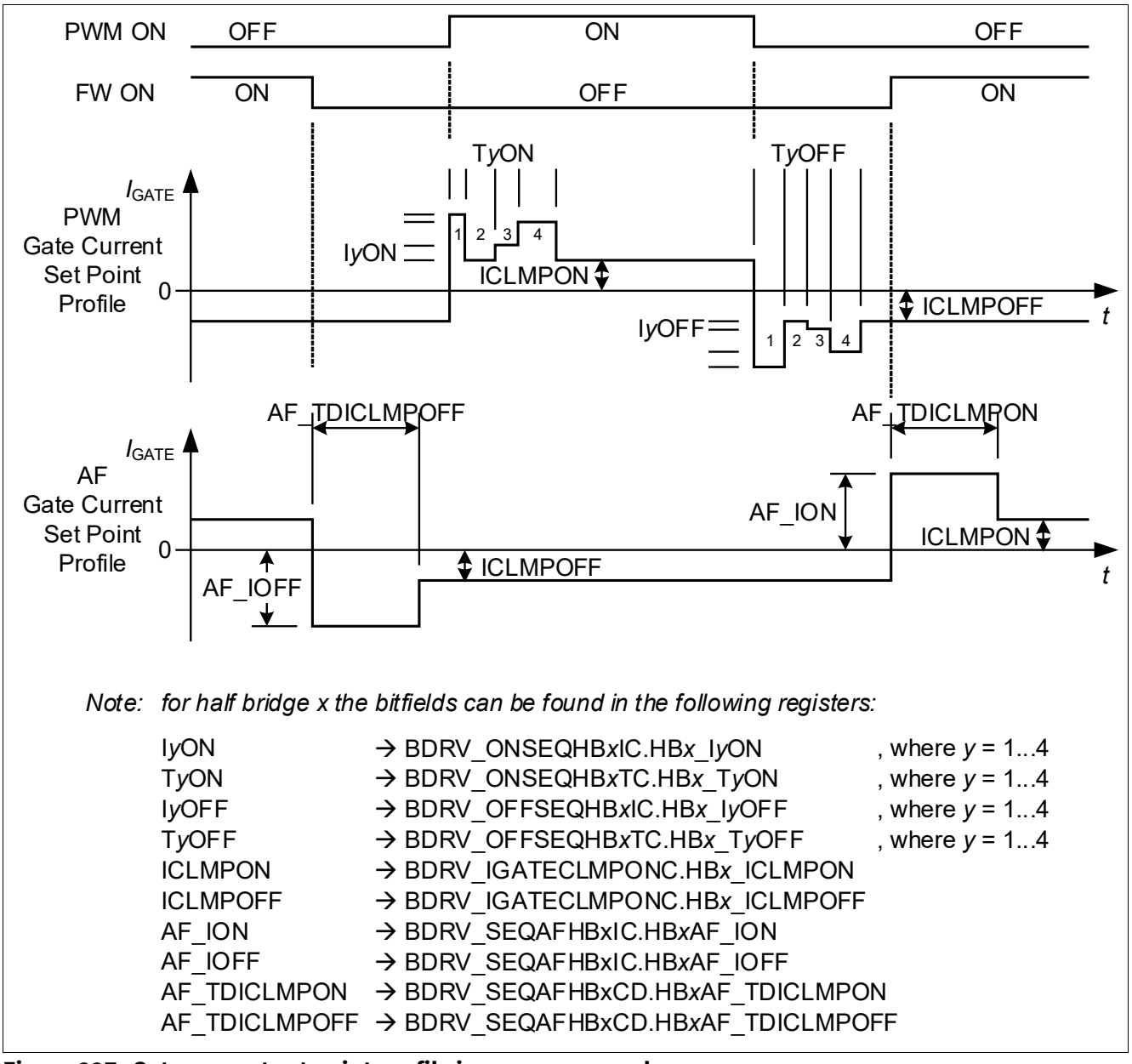
The constant mode provides a constant gate current set point I1ON/I1OFF for the time T1ON/T1OFF to switch on/off the external MOSFET. Then, it changes to a clamping current set point ICLMPON/ICLMPOFF during the rest of the on/off phase.

Depending on the application needs, the clamping current set point values can be programmed

- to low values in order to be robust against external shorts, or
- to high values in order to be robust against fast transients.

Note: The time values T1ON/T1OFF have to be chosen according to the actual entire MOSFET switching time in order to be able to control the switching slopes by I1ON/I1OFF.

The gate current set point values in sequencer mode are configured as shown in **Figure 227**.



Note: for half bridge x the bitfields can be found in the following registers:

I*y*ON	→ BDRV_ONSEQHB*x*IC.HB*x*_I*y*ON	, where *y* = 1...4
T*y*ON	→ BDRV_ONSEQHB*x*TC.HB*x*_T*y*ON	, where *y* = 1...4
I*y*OFF	→ BDRV_OFFSEQHB*x*IC.HB*x*_I*y*OFF	, where *y* = 1...4
T*y*OFF	→ BDRV_OFFSEQHB*x*TC.HB*x*_T*y*OFF	, where *y* = 1...4
ICLMPON	→ BDRV_IGATECLMPONC.HB*x*_ICLMPON	
ICLMPOFF	→ BDRV_IGATECLMPONC.HB*x*_ICLMPOFF	
AF_ION	→ BDRV_SEQAFHBxIC.HB*x*AF_ION	
AF_IOFF	→ BDRV_SEQAFHBxIC.HB*x*AF_IOFF	
AF_TDICLMPON	→ BDRV_SEQAFHBxCD.HB*x*AF_TDICLMPON	
AF_TDICLMPOFF	→ BDRV_SEQAFHBxCD.HB*x*AF_TDICLMPOFF	

Figure 227 Gate current set point profile in sequencer mode

The sequencer mode is able to provide 4 distinct gate current set points for the respective 4 MOSFET switching on/off phases shown in **Figure 228**.

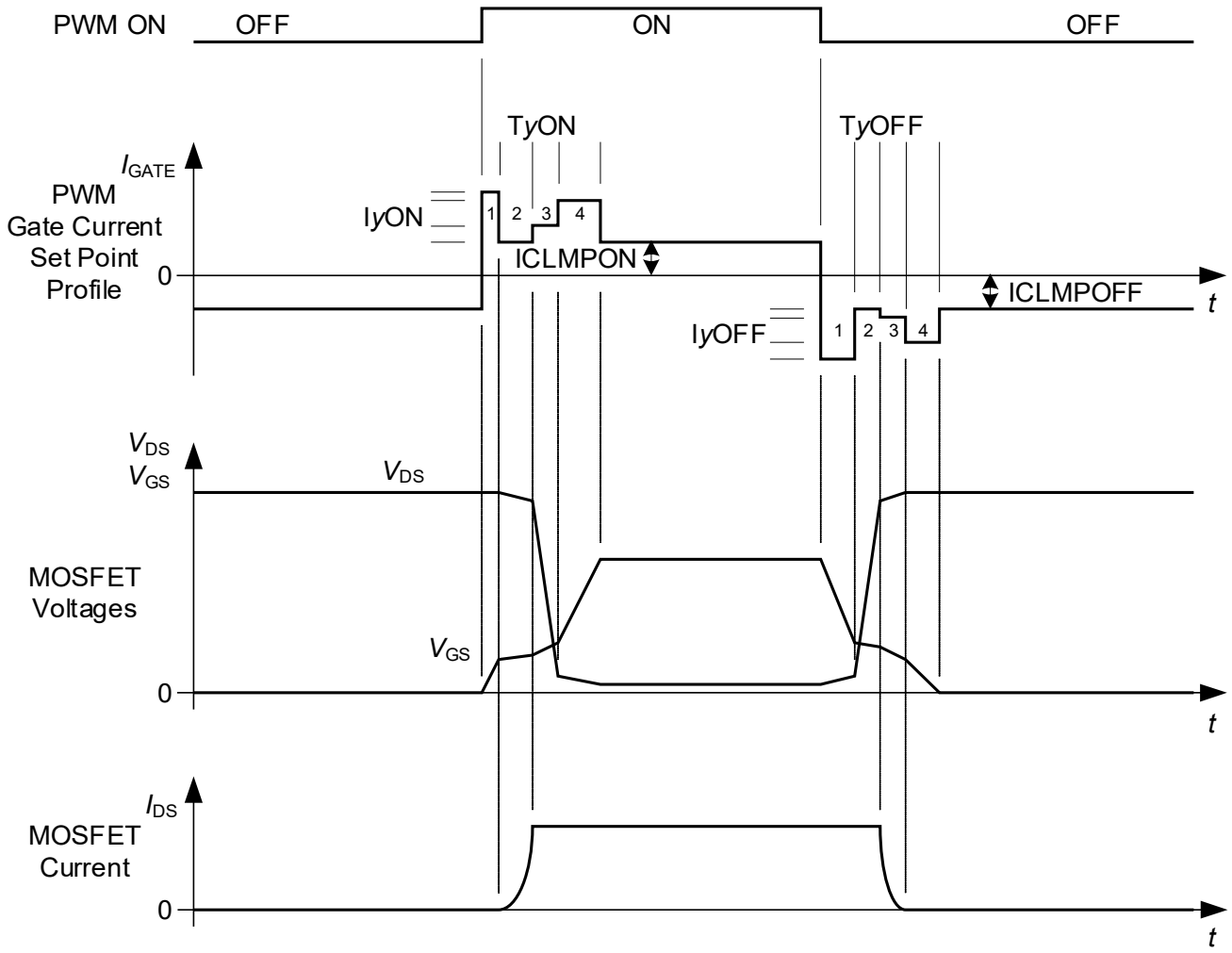


Figure 228 MOSFET switching phases and corresponding gate current set points

The 4 phases of switching on a MOSFET are:

- I1ON / T1ON: “pre-charge” phase (charge the gate until the threshold voltage)
- I2ON / T2ON: “d*I*/d*t*” phase (the MOSFET starts conducting until it takes the entire current)
- I3ON / T3ON: “d*V*/d*t*” phase (voltage slope)
- I4ON / T4ON: “post-charge” phase (the gate is charged to the maximum V_{GS} provided by the gate driver)

The 4 phases of switching off a MOSFET are:

- I1OFF / T1OFF: “pre-discharge” phase (discharge the gate until the voltage slope occurs)
- I2OFF / T2OFF: “d*V*/d*t*” phase (voltage slope)
- I3OFF / T3OFF: “d*I*/d*t*” phase (the MOSFET stops conducting until it takes no current)
- I4OFF / T4OFF: “post-discharge” phase (the gate is discharged to V_{GS} = 0 V)

*Note: The sequencer provides two gate current set points for the respective “slope” phases introduced in **Chapter 27.2.3** to individually control their “dI/dt” and “dV/dt” subphases.*

The nominal gate currents (I_{GATE}) for the respective set point values (x) can be estimated by

(27.1)

$$I_{GATE}(x) = 7\ mA + 313\ mA * (\frac{x}{63})^{1.35}$$

and are shown in **Table 526**.

Table 526 Nominal gate currents

x	I_{GATE}	x	I_{GATE}	x	I_{GATE}	x	I_{GATE}
0	7 mA	16	56 mA	32	132 mA	48	224 mA
1	8 mA	17	60 mA	33	138 mA	49	230 mA
2	10 mA	18	65 mA	34	143 mA	50	236 mA
3	12 mA	19	69 mA	35	149 mA	51	242 mA
4	15 mA	20	74 mA	36	154 mA	52	249 mA
5	17 mA	21	78 mA	37	160 mA	53	255 mA
6	20 mA	22	83 mA	38	165 mA	54	261 mA
7	23 mA	23	87 mA	39	171 mA	55	268 mA
8	26 mA	24	92 mA	40	177 mA	56	274 mA
9	30 mA	25	97 mA	41	182 mA	57	280 mA
10	33 mA	26	102 mA	42	188 mA	58	287 mA
11	37 mA	27	107 mA	43	194 mA	59	293 mA
12	40 mA	28	112 mA	44	200 mA	60	300 mA
13	44 mA	29	117 mA	45	206 mA	61	307 mA
14	48 mA	30	122 mA	46	212 mA	62	313 mA
15	52 mA	31	127 mA	47	218 mA	63	320 mA

27.3.3 Adjustable Cross-Conduction Protection

The cross-conduction protection feature can be enabled or disabled by the bit DRV_CCP_DIS of the register **BDRV_CTRL3**.

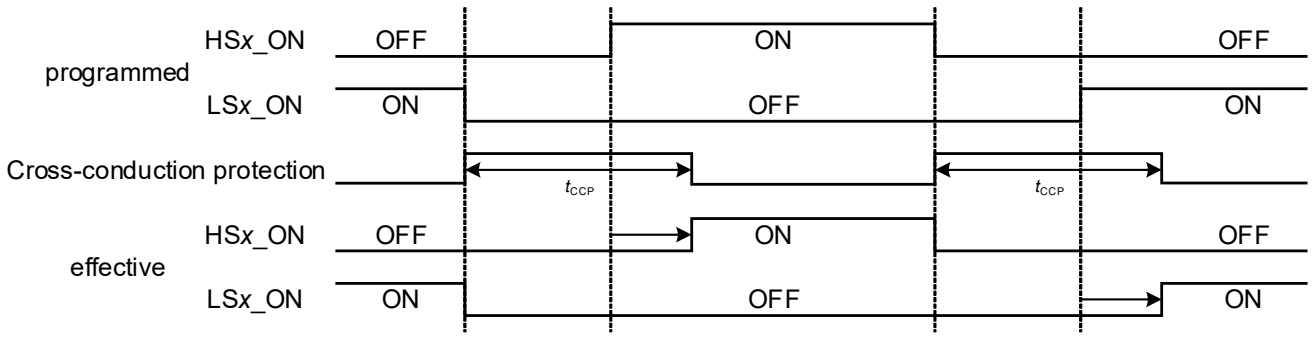


Figure 229 Cross-conduction protection

The minimum cross-conduction protection time t_{CCP} (see **Figure 229**) can be programmed by the bit fields DRV_CCP_TMUL and DRV_CCP_TIMSEL of the register **BDRV_CTRL3** according to **Table 527**:

Table 527 Cross-conduction protection time settings

	DRV_CCP_TMUL= 00_B	DRV_CCP_TMUL= 01_B	DRV_CCP_TMUL= 10_B	DRV_CCP_TMUL= 11_B
DRV_CCP_TIMSEL= 00_B	0.2 µs	0.4 µs	0.8 µs	1.6 µs
DRV_CCP_TIMSEL= 01_B	0.4 µs	0.8 µs	1.6 µs	3.2 µs
DRV_CCP_TIMSEL= 10_B	0.8 µs	1.6 µs	3.2 µs	6.4 µs
DRV_CCP_TIMSEL= 11_B	1.6 µs	3.2 µs	6.4 µs	12.8 µs

Note: The cross-conduction protection feature of the Bridge Driver is only able to increase the effective dead time between the complementary control signals of one half bridge (LSx_ON, HSx_ON) to the selected cross-conduction protection time. If the dead time of the programmed control signals (e.g. from CCU6) is already greater than the selected cross-conduction protection time this feature has no effect.

27.3.4 High-Current Discharge Mode

The high-current discharge mode can be enabled and disabled individually for each gate driver by the LSxDRV_FDISCHG_DIS and HSxDRV_FDISCHG_DIS bits of the register **BDRV_TRIM_DRVx**.

27.3.5 Passive Pull-Down Mode

The passive pull-down mode is automatically activated for a gate driver if it is not enabled in the register **BDRV_CTRL1** (bits LSx_EN, HSx_EN) and the hold mode is not active (see PMU_DRV_CTRL register).

27.3.6 Brake Mode

In brake mode the PWM capability of the bridge driver is not needed and the charge pump can be operated in a mode with lower current capability and therefore with lower current consumption. Depending on application needs one of the following setting can be chosen:

- enable low power mode by the bit CPLOPWRM_EN in register **BDRV_CP_CTRL** and optionally decrease the charge pump output voltage to nominal 9 V by the bit VCP9V_SET in register **BDRV_CP_CTRL**
- operate the charge pump in single stage mode by the bit field CP_STAGE_SEL in register **BDRV_CP_CTRL**
- switch off the charge pump by disabling its clock by the bit CPCLK_EN in register **BDRV_CP_CLK_CTRL**

27.3.7 Hold Mode

See PMU_DRV_CTRL register.

27.3.8 Timing Measurements

27.3.8.1 Fast Comparators

The results of the timing measurements are available in the following registers:

Table 528 Timing measurement results

	Half Bridge 1	Half Bridge 2
on delay	HB1_T12ONCNT in **BDRV_HB1ASEQONVAL**	HB2_T12ONCNT in **BDRV_HB2ASEQONVAL**
on slope	HB1_T3ONCNT in **BDRV_HB1ASEQONVAL**	HB2_T3ONCNT in **BDRV_HB2ASEQONVAL**
off delay	HB1_T1OFFCNT in **BDRV_HB1ASEQOFFVAL**	HB2_T1OFFCNT in **BDRV_HB2ASEQOFFVAL**
off slope	HB1_T2OFFCNT in **BDRV_HB1ASEQOFFVAL**	HB2_T2OFFCNT in **BDRV_HB2ASEQOFFVAL**

The fast comparator output signals (SH_low and SH_high as shown in **Figure 224** and **Figure 225**) are mapped to the HBx_T2CMP_STS[2)] and HBx_T3CMP_STS[3)] bits as follows depending on the mapping of the sequencer to LS or to HS by **BDRV_SEQMAP**.HBx_SEQMAP:

Table 529 Mapping of fast comparator output signals to the corresponding register status bits

	HBx_SEQMAP = ‘0’, i.e. observing LS[1)]				HBx_SEQMAP = ’1’, i.e. observing HS			
	SH_low		SH_high		SH_low		SH_high	
	‘0’	‘1’	‘0’	‘1’	‘0’	‘1’	‘0’	‘1’
HBx_T2CMP_STS [2)]	not mapped	not mapped	‘1’ =”LS on”	‘0’ =”LS off”	‘0’ =”HS off”	‘1’ =”HS on”	not mapped	not mapped
HBx_T3CMP_STS [3)]	‘1’ = “LS on”	‘0’ =“LS off”	not mapped	not mapped	not mapped	not mapped	‘0’ =”HS off”	‘1’ =”HS on”

1) if the sequencer is mapped to the low-side driver the status signals are inverted to the comparator output signals in order to have positive logic concerning the MOSFET status (‘0’ = “MOSFET off”, ‘1’ = “MOSFET on”)

2) i.e. **BDRV_HB1ASEQOFFVAL**.HB1_T2CMP_STS for half bridge 1 and **BDRV_HB2ASEQOFFVAL**.HB2_T2CMP_STS for half bridge 2

3) i.e. **BDRV_HB1ASEQONVAL**.HB1_T3CMP_STS for half bridge 1 and **BDRV_HB2ASEQONVAL**.HB2_T3CMP_STS for half bridge 2

27.3.8.2 Channel turn on/off delay measurement

For functional test and drive scheme timing optimization a dedicated timer is available to measure the delay between intended external MOS activation and actual turn on (VDS supervision). The principle function is shown in the following figure:

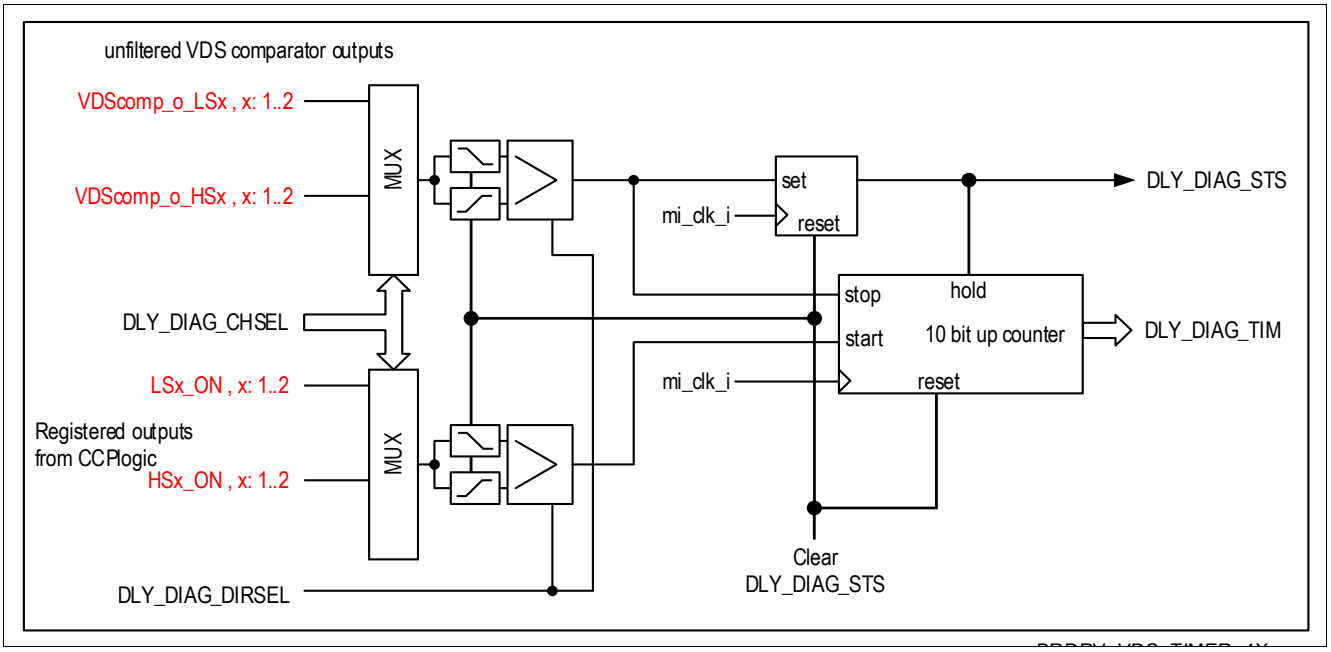


Figure 230 Principle of channel turn on/off timing measurement

27.3.9 Adaptive Control Mode

27.3.9.1 Introduction

The adaptive control mode uses the information from the timing measurement comparators (see **Chapter 27.3.8.1**) to optimize the settings for the pre-charge (“I1ON”) and pre-discharge (“I1OFF”) currents of the sequencer mode (see **Chapter 27.3.2**):

- If the measured time is longer than the target time then the respective gate current set point value is increased by one digit for the next switching event.
- If the measured time is shorter than the target time then the respective gate current set point value is decreased by one digit for the next switching event.

27.3.9.2 Target Settings

The target times for the optimizer of the adaptive contol mode are defined by the following timing values of the sequencer gate current set point profile:

- on delay target: BDRV_ONSEQHBxTC.HBx_T1ON + BDRV_ONSEQHBxTC.HBx_T2ON
- off delay target: BDRV_OFFSEQHBxTC.HBx_T1OFF + BDRV_OFFASEQTMIN.HBxT1OFFADDDLY

*Note: The above defined target times correspond to the delay between the change in the control signal and the respective dV/dt phase as shown in **Figure 228**. At the same time these delays correspond to the measured signals $t_{sdly(on)}$ and $t_{sdly(off)}$ shown in **Figure 224** and **Figure 225**.*

Note: For the off delay target an additional delay can be defined by BDRV_OFFASEQTMIN.HBxT1OFFADDDLY to let the gate driver settle to the target gate current value for the dV/dt phase.

27.3.9.3 Optimizer Activation

The adaptive control mode is individually set up for each half bridge and separately for on and off phases by the bits of the BDRV_ASEQC register according to **Table 530**:

Table 530 Adaptive Sequencer Mode control bits

Half bridge	**HB1**		**HB2**	
Switching phase	**On**	**Off**	**On**	**Off**
Enable	HB1ASMONEN	HB1ASMOFFEN	HB2ASMONEN	HB2ASMOFFEN
Optimizer activation	HB1OPTONACT	HB1OPTOFFACT	HB2OPTONACT	HB2OPTOFFACT
Hysteresis enable[1)]	HB1ONHYSTEN	HB1OFFHYSTEN	HB2ONHYSTEN	HB2OFFHYSTEN

1) If the hysteresis is enabled the optimizer adapts the gate current set point value only after the target time was missed 3 times in a row in the same direction.

27.3.9.4 Monitoring

The optimizer is monitored

- by checking the resulting gate current set points against the limits defined in
  - BDRV_ASEQIONMIN.I1ONMIN
  - BDRV_ASEQIONMAX.I1ONMAX
  - BDRV_ASEQIOFFMIN.I1OFFMIN
  - BDRV_ASEQIOFFMAX.I1OFFMAX
- by checking the timing measurement results against the limits defined in
  - BDRV_ONASEQTMIN.T12ONMIN
  - BDRV_ONASEQTMAX.T12ONMAX
  - BDRV_OFFASEQTMIN.T1OFFMIN
  - BDRV_OFFASEQTMAX.T1OFFMAX

If one of these limits is exceeded in the respective direction often enough (as defined in BDRV_ASEQERRCNT) the corresponding status flag is set in the BDRV_ASEQSTS register and an interrupt can be triggered (see BDRV_IRQEN, BDRV_IRQS, BDRV_IRQCLR).

27.3.10 Integrated 2-Stage Charge Pump

The driver is supplied by a 2-Stage Charge Pump. The charge pump enables a duty cycle range from 0 - 100% at the external MOSFETs. The regulated output voltage is nominally VS + 14V. The charge pump output VCP is monitored by the Measurement Unit. The Bridge Driver including the charge pump can be optionally disabled at undervoltage and/or overvoltage events at VCP.

27.3.10.1 Clock Generator of Driver Supply

The clock generator of the charge pump uses a spread spectrum technique to minimize emmission caused by the charge pump operation on the supply voltage VSD. The structure of the clock generation for the charge pump is shown in the figure below:

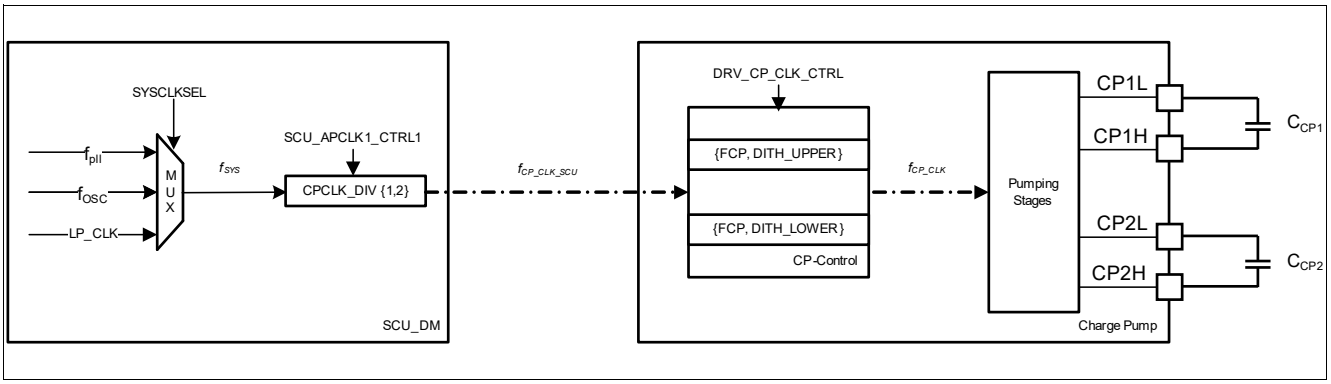


Figure 231 Clock Generation of Charge Pump Block

The charge pump clock f_{CP_CLK} is derived from the system clock f_{sys}. Inside the SCU_DM the system clock is divided by a configurable value and provided as $f_{CP\_CLK\_SCU}$ to the charge pump control block. During operation the frequency of the charge pump is varied between two frequency boundaries. These boundaries are defined by the concatenation of {FCP, DITH_UPPER} bits for the upper boundary and {FCP, DITH_LOWER} bits for the lower boundary (the concatenated bits represent a frequency divider value for $f_{CP\_CLK\_SCU}$).

27.3.11 Adjustable Voltage Monitoring

The voltages at the VSD and the VCP pins are monitored by the measurement unit (see “Chapter Measurement Unit”):

- VSD pin voltage is monitored by the ADC2 channel 1
- VCP pin voltage is monitored by the ADC2 channel 2

27.3.12 Adjustable Short-Circuit Detection

The Drain Souce Voltage (VDS) of each MOSFET is monitored by a comparator. In case the VDS voltage is higher then the limit set in DSMONVTH during the on phase of the MOSFET all drivers are switched off. The bit HSx_OC_STS or LSx_OC_STS is set. The feature can be disabled by bit HSx_OC_DIS / LSx_OC_DIS.

The filter time for the VDS measurements can be adjusted by the bits HSDRV_DS_TFILT_SEL and LSDRV_DS_TFILT_SEL.

The blank time for the VDS measurement can be adjusted by the bits LS_HS_BT_TFILT_SEL.

27.3.13 Overtemperature

The overtemperature detection and shutdown status can be monitored by the bit VCP_OTSD_STS in the register **BDRV_CP_IRQS**.

27.4 Register Definition

The Bridge Driver registers are located in the address space below.

Table 531 Register Address Space

Module	Base Address	End Address	Note
BDRV	40034000_H	$40037FFF_H$	Bridge Driver

Table 532 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Register Definition, Driver Register			
BDRV_CTRL1	H-Bridge Driver Control 1	00_H	see **Table 533**
BDRV_CTRL2	H-Bridge Driver Control 2	04_H	see **Table 534**
BDRV_CTRL3	H-Bridge Driver Control 3	08_H	see **Table 535**
BDRV_PWMSRCSEL	PWM Source Selection Register	$0C_H$	see **Table 536**
BDRV_IGATECLMPONC	Gate Current Clamping Value in ON State	30_H	see **Table 537**
BDRV_IGATECLMPOFFC	Gate Current Clamping Value in OFF State	34_H	see **Table 538**
BDRV_IRQS	H-Bridge Driver Interrupt Status	$F0_H$	see **Table 539**
BDRV_IRQCLR	H-Bridge Driver Interrupt Status Clear Register	$F4_H$	see **Table 540**
BDRV_IRQEN	H-Bridge Driver Control	$F8_H$	see **Table 541**
Register Definition, Sequencer Configuration Registers			
BDRV_SEQMAP	Slewrate Sequencer Mapping Register	10_H	see **Table 542**
Register Definition, Half Bridge 1 - Slew Rate Configuration Registers for Switch-Off/On.			
BDRV_OFFSEQHB1TC	Turn-off Slewrate Sequencer Half Bridge 1 Time Control	50_H	see **Table 543**
BDRV_OFFSEQHB1IC	Turn-off Slewrate Sequencer Half Bridge 1 Current Control	54_H	see **Table 544**
BDRV_ONSEQHB1TC	Turn-on Slewrate Sequencer Half Bridge 1 Time Control	58_H	see **Table 545**
BDRV_ONSEQHB1IC	Turn-on Slewrate Sequencer Half Bridge 1 Current Control	$5C_H$	see **Table 546**
Register Definition, Half Bridge 1 - Slew Rate Configuration Registers for Active Freewheeling.			
BDRV_SEQAFHB1IC	Slewrate Sequencer-Active Freewheeling-Half Bridge 1 Current Control	64_H	see **Table 547**
BDRV_SEQAFHB1CD	Slewrate Sequencer-Active Freewheeling- Half Bridge 1 Clamping Current Delay	68_H	see **Table 548**
Register Definition, Half Bridge 2 - Slew Rate Configuration Registers for Switch-Off/On			
BDRV_OFFSEQHB2TC	Turn-off Slewrate Sequencer Half Bridge 2 Time Control	70_H	see **Table 549**
BDRV_OFFSEQHB2IC	Turn-off Slewrate Sequencer Half Bridge 2 Current Control	74_H	see **Table 550**

Table 532 Register Overview (cont'd)

Register Short Name	Register Long Name	Offset Address	Reset Value
BDRV_ONSEQHB2TC	Turn-on Slewrate Sequencer Half Bridge 2 Time Control	78_H	see Table 551
BDRV_ONSEQHB2IC	Turn-on Slewrate Sequencer Half Bridge 2 Current Control	$7C_H$	see Table 552
Register Definition, Half Bridge 2 - Slew Rate Configuration Registers for Active Freewheeling			
BDRV_SEQAFHB2IC	Slewrate Sequencer-Active Freewheeling- Half Bridge 2 Current Control	84_H	see Table 553
BDRV_SEQAFHB2CD	Slewrate Sequencer-Active Freewheeling- Half Bridge 2 Clamping Current Delay	88_H	see Table 554
Register Definition, Adaptive Slew Rate Sequencer Control and Status Registers			
BDRV_ASEQC	Adaptive Slewrate Sequencer Control Register	90_H	see Table 555
BDRV_ASEQSTS	Adaptive Slewrate Sequencer Status Register	94_H	see Table 556
BDRV_ASEQERRCNT	Adaptive Slewrate Sequencer Error Counter Control Register	$D8_H$	see Table 557
Register Definition, Adaptive Slew Rate Sequencer Configuration Registers			
BDRV_ONASEQTMIN	Turn ON Adaptive Slewrate Sequencer Minimum Time Setting	98_H	see Table 558
BDRV_OFFASEQTMIN	Turn OFF Adaptive Slewrate Sequencer Minimum Time Setting	$9C_H$	see Table 559
BDRV_ASEQIONMIN	Adaptive Slewrate Sequencer On Phase Minimum Current Setting	$A0_H$	see Table 560
BDRV_ASEQIOFFMIN	Adaptive Slewrate Sequencer Off Phase Minimum Current Setting	$A4_H$	see Table 561
BDRV_ONASEQTMAX	Adaptive Slewrate On Sequencer Maximum Time Setting	$A8_H$	see Table 562
BDRV_OFFASEQTMAX	Adaptive Slewrate Off Sequencer Maximum Time Setting	AC_H	see Table 563
BDRV_ASEQIONMAX	Adaptive Slewrate Sequencer On Phase Maximum Current Setting	$B0_H$	see Table 564
BDRV_ASEQIOFFMAX	Adaptive Slewrate Sequencer Off Phase Maximum Current Setting	$B4_H$	see Table 565
BDRV_HB1ASEQONVAL	Half Bridge 1 Adaptive Sequencer On Values	$B8_H$	see Table 566
BDRV_HB1ASEQOFFVAL	Half Bridge 1 Adaptive Sequencer Off Values	BC_H	see Table 567
BDRV_HB2ASEQONVAL	Half Bridge 2 Adaptive Sequencer On Values	$D0_H$	see Table 568
BDRV_HB2ASEQOFFVAL	Half Bridge 2 Adaptive Sequencer Off Values	$D4_H$	see Table 569
Register Definition, Driver Trimming Register			
BDRV_TRIM_DRVx	Trimming of Driver	18_H	see Table 570

Table 532 Register Overview (cont'd)

Register Short Name	Register Long Name	Offset Address	Reset Value
Register Definition, Charge Pump Control and Status Register			
BDRV_CP_CTRL	Charge Pump Control and Status Register	20_H	see **Table 571**
BDRV_CP_CLK_CTRL	Charge Pump Clock Control Register	24_H	see **Table 572**
BDRV_CP_IRQS	Charge Pump Status Register	40_H	see **Table 573**
BDRV_CP_IRQCLR	Charge Pump Interrupt Status Clear Register	44_H	see **Table 574**
BDRV_CP_IRQEN	Charge Pump Interrupt Enable Register	48_H	see **Table 575**
Register Definition, Dynamic Compensation Trimming Register			
BDRV_DCTRIM_DRVx	Current Trimming of Driver	$E0_H$	see **Table 576**

The registers are addressed wordwise.

27.4.1 Driver Register

H-Bridge (Half Bridge) Driver Control Register 1

Attention: ***The Bridge Driver module can only be enabled when all FET drivers are enabled in the register below.***

BDRV_CTRL1	Offset	Reset Value
H-Bridge Driver Control 1	**00_H**	**see Table 533**

31	30	29	28	27	26	25	24
HS2_OC_DIS	RES	HS2_SUPERR_STS	RES	HS2_DCS_EN	HS2_ON	HS2_PWM	HS2_EN
rw	r	r	r	rw	rwhir	rwhir	rw
23	22	21	20	19	18	17	16
HS1_OC_DIS	RES	HS1_SUPERR_STS	RES	HS1_DCS_EN	HS1_ON	HS1_PWM	HS1_EN
rw	r	r	r	rw	rwhir	rwhir	rw
15	14	13	12	11	10	9	8
LS2_OC_DIS	RES	LS2_SUPERR_STS	RES		LS2_ON	LS2_PWM	LS2_EN
rw	r	r	r		rwhir	rwhir	rw
7	6	5	4	3	2	1	0
LS1_OC_DIS	RES	LS1_SUPERR_STS	RES		LS1_ON	LS1_PWM	LS1_EN
rw	r	r	r		rwhir	rwhir	rw

Field	Bits	Type	Description
HS2_OC_DIS	31	rw	**High Side Driver Overcurrent Shutdown Select** 0_H **Global Shutdown**, all bridges will be shut down in case of overcurrent 1_H **Local Shutdown**, only local driver will be shut down in case of overcurrent
RES	30	r	**Reserved** Always read as 0
HS2_SUPERR_STS	29	r	**High Side Driver 2 Supply Error Status** 0_B **NORMAL**, supply is in required range. 1_B **SUPPLY ERROR**, detected; this flag is an OR of the VSD_x_STS and VCP_x_STS flags.
RES	28	r	**Reserved** Always read as 0
HS2_DCS_EN	27	rw	**High Side Driver 2 Diagnosis Current Source Enable** *Note:* ***BDRV_IGATECLMPOFFC****.HB2_ICLMPOFF has to be programmed to 0h* 0_H **DISABLE**, disable current source 1_H **ENABLE**, enable current source; short diagnosis can be performed by evaluating the LSx/HSx_DS_STS Flag
HS2_ON	26	rwhir	**High Side Driver 2 On** 0_B **OFF**, Driver off 1_B **ON**, Driver on
HS2_PWM	25	rwhir	**High Side Driver 2 PWM Enable** *Note:* *This bit can only be set if HS2_ON and LS2_ON are 0.* 0_B **DISABLE**, disables control by PWM input 1_B **ENABLE**, enables control by PWM input
HS2_EN	24	rw	**High Side Driver 2 Enable** 0_B **DISABLE**, Driver circuit power off 1_B **ENABLE**, Driver circuit power on
HS1_OC_DIS	23	rw	**High Side Driver Overcurrent Shutdown Select** 0_H **Global Shutdown**, all bridges will be shut down in case of overcurrent 1_H **Local Shutdown**, only local driver will be shut down in case of overcurrent
RES	22	r	**Reserved** Always read as 0

Field	Bits	Type	Description
HS1_SUPERR_STS	21	r	**High Side Driver 1 Supply Error Status** 0_B **NORMAL**, supply is in required range. 1_B **SUPPLY ERROR**, detected; this flag is an OR of the VDS_x_STS and VCP_x_STS flags.
RES	20	r	**Reserved** Always read as 0
HS1_DCS_EN	19	rw	**High Side Driver 1 Diagnosis Current Source Enable** *Note:* ***BDRV_IGATECLMPOFFC****.HB1_ICLMPOFF has to be programmed to 0h* 0_H **DISABLE**, disable current source 1_H **ENABLE**, enable current source; short diagnosis can be performed by evaluating the LSx/HSx_DS_STS Flag
HS1_ON	18	rwhir	**High Side Driver 1 On** 0_B **OFF**, Driver off 1_B **ON**, Driver on
HS1_PWM	17	rwhir	**High Side Driver 1 PWM Enable** *Note:* *This bit can only be set if HS1_ON and LS1_ON are 0 and PWM enable only takes effect if the bits HS2_PWM and/or LS2_PWM are 1.* 0_B **DISABLE**, disables control by PWM input 1_B **ENABLE**, enables control by PWM input
HS1_EN	16	rw	**High Side Driver 1 Enable** 0_B **DISABLE**, Driver circuit power off 1_B **ENABLE**, Driver circuit power on
LS2_OC_DIS	15	rw	**Low Side Driver Overcurrent Shutdown Select** 0_H **Global Shutdown**, all bridges will be shut down in case of overcurrent 1_H **Local Shutdown**, only local driver will be shut down in case of overcurrent
RES	14	r	**Reserved** Always read as 0
LS2_SUPERR_STS	13	r	**Low Side Driver 2 Supply Error Status** 0_B **NORMAL**, supply is in required range. 1_B **SUPPLY ERROR**, detected; this flag is an OR of the VDS_x_STS and VCP_x_STS flags.
RES	12:11	r	**Reserved** Always read as 0
LS2_ON	10	rwhir	**Low Side Driver 2 On** 0_B **OFF**, Driver off 1_B **ON**, Driver on

Field	Bits	Type	Description
LS2_PWM	9	rwhir	**Low Side Driver 2 PWM Enable** *Note: This bit can only be set if HS2_ON and LS2_ON are 0 and PWM enable only takes effect if the bits HS2_PWM and/or LS2_PWM are 1.* 0_B **DISABLE**, disables control by PWM input 1_B **ENABLE**, enables control by PWM input
LS2_EN	8	rw	**Low Side Driver 2 Enable** 0_B **DISABLE**, Driver circuit power off 1_B **ENABLE**, Driver circuit power on
LS1_OC_DIS	7	rw	**Low Side Driver 1 Overcurrent Shutdown Select** 0_H **Global Shutdown**, all bridges will be shut down in case of overcurrent 1_H **Local Shutdown**, only local driver will be shut down in case of overcurrent
RES	6	r	**Reserved** Always read as 0
LS1_SUPERR_STS	5	r	**Low Side Driver 1 Supply Error Status** 0_B **NORMAL**, supply is in required range. 1_B **SUPPLY ERROR**, detected; this flag is an OR of the VDS_x_STS and VCP_x_STS flags.
RES	4:3	r	**Reserved** Always read as 0
LS1_ON	2	rwhir	**Low Side Driver 1 On** 0_B **OFF**, Driver off 1_B **ON**, Driver on
LS1_PWM	1	rwhir	**Low Side Driver 1 PWM Enable** *Note: This bit can only be set if HS1_ON and LS1_ON are 0.* 0_B **DISABLE**, disables control by PWM input 1_B **ENABLE**, enables control by PWM input
LS1_EN	0	rw	**Low Side Driver 1 Enable** 0_B **DISABLE**, Driver circuit power off 1_B **ENABLE**, Driver circuit power on

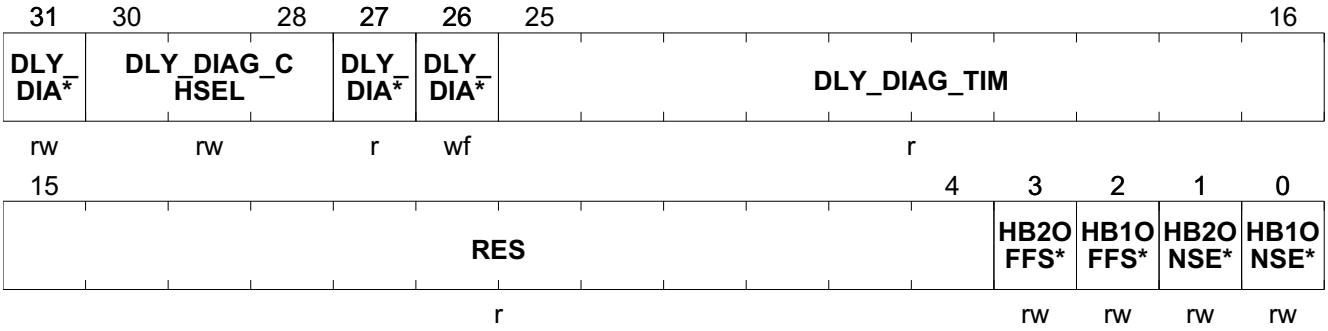
Table 533 RESET of BDRV_CTRL1

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	01010101_H	RESET_TYPE_3		

Bridge Driver (incl. Charge Pump)

H-Bridge Driver Control Register 2

BDRV_CTRL2	Offset	Reset Value
H-Bridge Driver Control 2	04_H	see Table 534



Field	Bits	Type	Description
DLY_DIAG_DIRSEL	31	rw	**Ext. power diag timer on / off select** 0_B **TURN OFF**, measure turn off time 1_B **TURN ON**, measure turn on time
DLY_DIAG_CHSEL	30:28	rw	**Ext. power diag timer channel select** 000_B **DISABLE**, diag timer deactivated. 001_B **HB1 LS select**, measure LS1 on/off delay time. 010_B **HB2 LS select**, measure LS2 on/off delay time. 011_B **DISABLE**, diag timer deactivated. 100_B **DISABLE**, diag timer deactivated. 101_B **HB1 HS select**, measure HS1 on/off delay time. 110_B **HB2 HS select**, measure HS2 on/off delay time. 111_B **DISABLE**, diag timer deactivated.
DLY_DIAG_STS	27	r	**Ext. power diag timer valid flag** *Note: Clear flag to start a measurement.* 0_B **Diag timer invalid**, diag timer measurement ongoing 1_B **Diag timer valid**, diag timer measurement finished
DLY_DIAG_SCLR	26	wf	**Ext. power diag timer valid flag clear** 0_B **Diag timer valid not clear**, 1_B **Diag timer valid clear**,
DLY_DIAG_TIM	25:16	r	**Ext. power diag timer result register**
RES	15:4	r	**Reserved** Always read as 0
HB2OFFSEQCNF	3	rw	**Half Bridge 2 Off Sequencer Configuration** 0_B **Normal Mode**, OFF-Sequencer is disabled and driver operates with constant current. 1_B **Sequencer Mode**, OFF-Sequencer is enabled.

Field	Bits	Type	Description
HB1OFFSEQCNF	2	rw	**Half Bridge 1 Off Sequencer Configuration** 0_B **Normal Mode**, OFF-Sequencer is disabled and driver operates with constant current. 1_B **Sequencer Mode**, OFF-Sequencer is enabled.
HB2ONSEQCNF	1	rw	**Half Bridge 2 On Sequencer Configuration** 0_B **Normal Mode**, ON-Sequencer is disabled and driver operates with constant current. 1_B **Sequencer Mode**, ON-Sequencer is enabled.
HB1ONSEQCNF	0	rw	**Half Bridge 1 On Sequencer Configuration** 0_B **Normal Mode**, ON-Sequencer is disabled and driver operates with constant current. 1_B **Sequencer Mode**, ON-Sequencer is enabled.

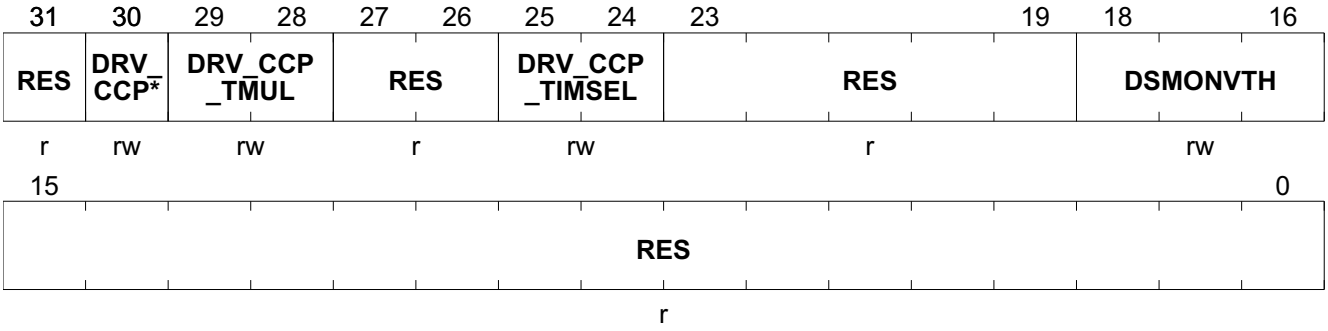
Table 534 RESET of BDRV_CTRL2

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Bridge Driver (incl. Charge Pump)

H-Bridge Driver Control 3

BDRV_CTRL3	Offset	Reset Value
H-Bridge Driver Control 3	08_H	see Table 535



Field	Bits	Type	Description
RES	31	r	**Reserved** Always read as 0
DRV_CCP_DIS	30	rw	**Dynamic cross conduction protection Disable** *Note: the cross condution protection consists of two stages. This flag disables the first stage which is the activation of the delayed gate clamp after the configured cross condution protection time. The second stage which is represented by the delayed gate clamp is still active and will be activated as soon as the opposite MOSFET within an inverter stage is activated.* 0_B **CCP Enable**, dynamic ccp is active. 1_B **CCP Disable**, dynamic ccp is disabled, delayed gate clamp remains active.
DRV_CCP_TMUL	29:28	rw	**Multiplier bits for cross conduction time settings in register DRV_CCP_TIMSEL** 00_B **MUL1**, DRV_CCP_TIMSEL value is multiplied by 1 01_B **MUL2**, DRV_CCP_TIMSEL value is multiplied by 2 10_B **MUL4**, DRV_CCP_TIMSEL value is multiplied by 4 11_B **MUL8**, DRV_CCP_TIMSEL value is multiplied by 8
RES	27:26	r	**Reserved** Always read as 0
DRV_CCP_TIMSEL	25:24	rw	**Minimum cross conduction protection time setting**[1)] 00_B **0.2us**, 200ns cross conduction protection time 01_B **0.4us**, 400ns cross conduction protection time 10_B **0.8us**, 800ns cross conduction protection time 11_B **1.6us**, 1.6us cross conduction protection time
RES	23:19	r	**Reserved** Always read as 0

Field	Bits	Type	Description
DSMONVTH	18:16	rw	**Voltage Threshold for Drain-Source Monitoring of external FETs** 000_B **0.125_V**, Threshold 0 for VDS at 0.125 V 001_B **0.25_V**, Threshold 1 for VDS at 0.25 V 010_B **0.50_V**, Threshold 2 for VDS at 0.50 V 011_B **0.75_V**, Threshold 3 for VDS at 0.75 V 100_B **1.00_V**, Threshold 4 for VDS at 1.00 V 101_B **1.25_V**, Threshold 5 for VDS at 1.25 V 110_B **1.50_V**, Threshold 6 for VDS at 1.50 V 111_B **1.75_V**, Threshold 7 for VDS at 1.75 V
RES	15:0	r	**Reserved** Always read as 0

1) if BRDRV_CLK = 20 MHz

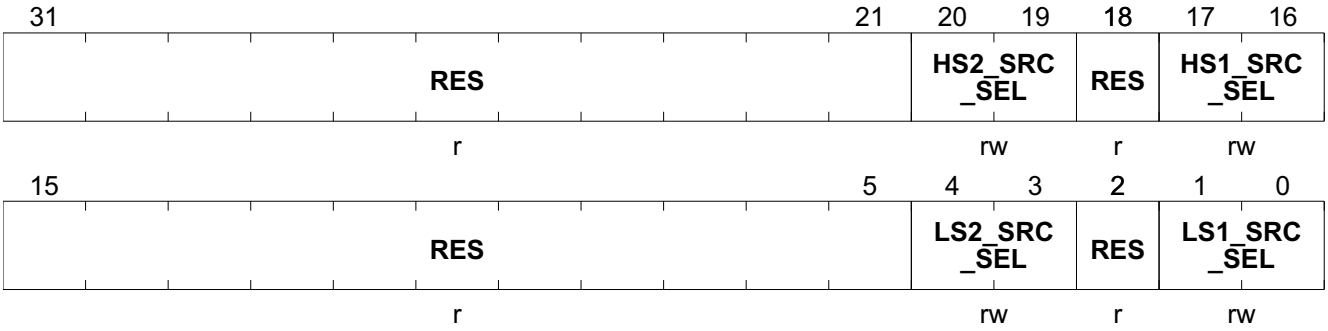
Table 535 RESET of BDRV_CTRL3

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00010000_H	RESET_TYPE_3		

Bridge Driver (incl. Charge Pump)

PWM Source Selection Register

BDRV_PWMSRCSEL	Offset	Reset Value
PWM Source Selection Register	$0C_H$	see Table 536



Field	Bits	Type	Description
RES	31:21	r	**Reserved** Always read as 0
HS2_SRC_SEL	20:19	rw	**HS2 PWM Source Selection** *Note: Can only be written if DRV_CTRL1.HS2_PWM=0.* 00_B **CC60**, PWM output of CCU6 01_B **CC61**, PWM output of CCU6 10_B **COUT60**, PWM output of CCU6 11_B **COUT61**, PWM output of CCU6
RES	18	r	**Reserved** Always read as 0
HS1_SRC_SEL	17:16	rw	**HS1 PWM Source Selection** *Note: Can only be written if DRV_CTRL1.HS1_PWM=0.* 00_B **CC60**, PWM output of CCU6 01_B **CC61**, PWM output of CCU6 10_B **COUT60**, PWM output of CCU6 11_B **COUT61**, PWM output of CCU6
RES	15:5	r	**Reserved** Always read as 0
LS2_SRC_SEL	4:3	rw	**LS2 PWM Source Selection** *Note: Can only be written if DRV_CTRL1.LS2_PWM=0.* 00_B **CC60**, PWM output of CCU6 01_B **CC61**, PWM output of CCU6 10_B **COUT60**, PWM output of CCU6 11_B **COUT61**, PWM output of CCU6

Field	Bits	Type	Description
RES	2	r	**Reserved** Always read as 0
LS1_SRC_SEL	1:0	rw	**LS1 PWM Source Selection** *Note: Can only be written if DRV_CTRL1.LS1_PWM=0.* 00_B **CC60**, PWM output of CCU6 01_B **CC61**, PWM output of CCU6 10_B **COUT60**, PWM output of CCU6 11_B **COUT61**, PWM output of CCU6

Table 536 RESET of BDRV_PWMSRCSEL

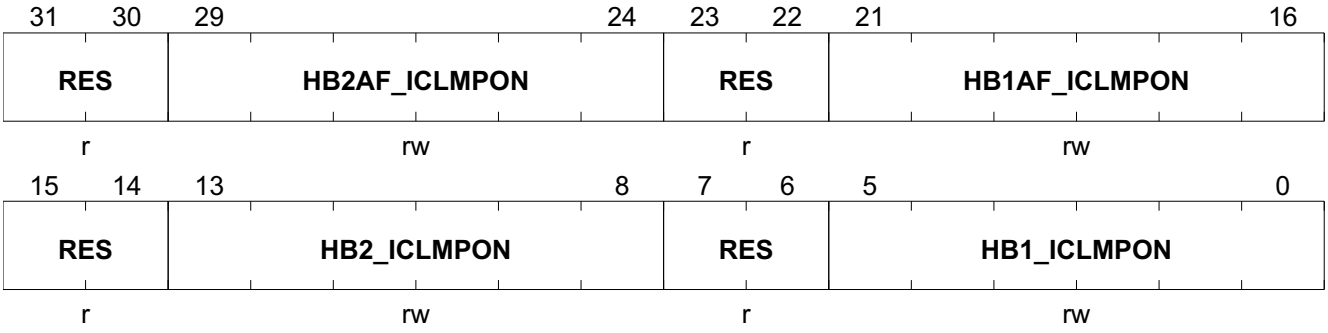
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Gate Current Clamping Value Control On State

BDRV_IGATECLMPONC — **Gate Current Clamping Value in ON State**

Offset 30_H

Reset Value see Table 537



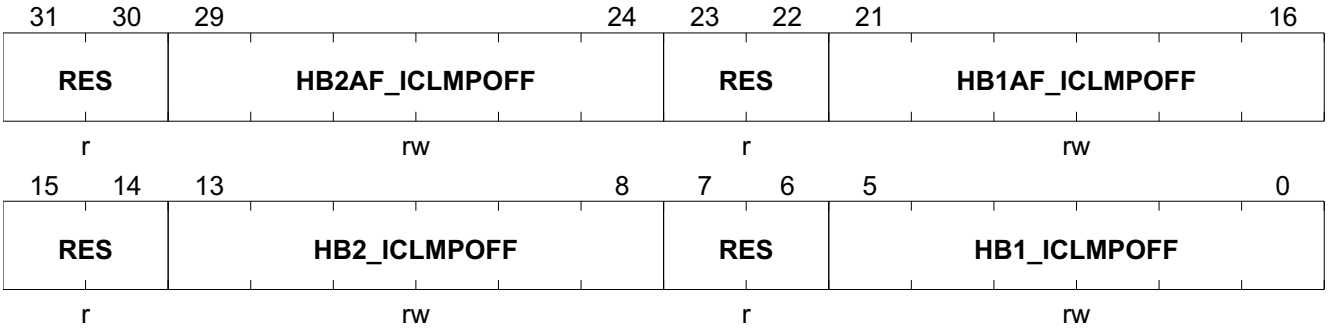
Field	Bits	Type	Description
RES	31:30	r	**Reserved** Always read as 0
HB2AF_ICLMPON	29:24	rw	**Half Bridge 2-active freewheeling-current clamping value for ON state** 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}
RES	23:22	r	**Reserved** Always read as 0
HB1AF_ICLMPON	21:16	rw	**Half Bridge 1-active freewheeling-current clamping value for ON state** 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}
RES	15:14	r	**Reserved** Always read as 0
HB2_ICLMPON	13:8	rw	**Half Bridge 2-current clamping value for ON state** 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}
RES	7:6	r	**Reserved** Always read as 0
HB1_ICLMPON	5:0	rw	**Half Bridge 1-current clamping value for ON state** 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}

Table 537 RESET of BDRV_IGATECLMPONC

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	07070707_H	RESET_TYPE_3		

Gate Current Clamping Value Control On State

BDRV_IGATECLMPOFFC	Offset	Reset Value
Gate Current Clamping Value in OFF State	34_H	see Table 538



Field	Bits	Type	Description
RES	31:30	r	**Reserved** Always read as 0
HB2AF_ICLMPOFF	29:24	rw	**Half Bridge 2-active freewheeling-current clamping value for OFF state** 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$
RES	23:22	r	**Reserved** Always read as 0
HB1AF_ICLMPOFF	21:16	rw	**Half Bridge 1-active freewheeling-current clamping value for OFF state** 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$
RES	15:14	r	**Reserved** Always read as 0
HB2_ICLMPOFF	13:8	rw	**Half Bridge 2-current clamping value for OFF state** 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$
RES	7:6	r	**Reserved** Always read as 0
HB1_ICLMPOFF	5:0	rw	**Half Bridge 1-current clamping value for OFF state** 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$

Table 538 RESET of BDRV_IGATECLMPOFFC

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	07070707_H	RESET_TYPE_3		

H-Bridge (Half Bridge) Driver Interrupt Status Register

BDRV_IRQS	Offset	Reset Value
H-Bridge Driver Interrupt Status	$F0_H$	see Table 539

31	30	29	28	27			24
SEQ_ERR_IS	HS2_OC_IS	HS2_DS_STS	HS2_DS_IS	RES			
r	r	r	r	r			
23	22	21	20	19			16
RES	HS1_OC_IS	HS1_DS_STS	HS1_DS_IS	RES			
r	r	r	r	r			
15	14	13	12	11			8
RES	LS2_OC_IS	LS2_DS_STS	LS2_DS_IS	RES			
r	r	r	r	r			
7	6	5	4	3	2	1	0
RES	LS1_OC_IS	LS1_DS_STS	LS1_DS_IS	RES		HB2_ASEQ_IS	HB1_ASEQ_IS
r	r	r	r	r		r	r

Field	Bits	Type	Description
SEQ_ERR_IS	31	r	**Driver Sequence Error Interrupt Status** 0_B **Driver Sequence ok**, no cross current 1_B **Driver Sequence fail**, HS and LS of same bridge concurrently activated, output protection activated
HS2_OC_IS	30	r	**External High Side 2 FET Over-current Interrupt Status** 0_B **no Over-current**, no over-current Condition occurred. 1_B **Over-current**, over-current occurred; switch is automatically shut down.
HS2_DS_STS	29	r	**High Side Driver 2 Drain Source Monitoring Status in OFF-State** 0_B **no short on external FET**, no short detected. 1_B **short on external FET detected**, short detected.
HS2_DS_IS	28	r	**High Side Driver 2 Drain Source Monitoring Interrupt Status in OFF-State** 0_B **no short on external FET**, no short detected. 1_B **short on external FET detected**, short detected.
RES	27:23	r	**Reserved** Always read as 0

Field	Bits	Type	Description
HS1_OC_IS	22	r	**External High Side 1 FET Over-current Interrupt Status** 0_B **no Over-current**, no over-current Condition occurred. 1_B **Over-current**, over-current occurred; switch is automatically shutdown.
HS1_DS_STS	21	r	**High Side Driver 1 Drain Source Monitoring Status in OFF-State** 0_B **no short on external FET**, no short detected. 1_B **short on external FET detected**, short detected.
HS1_DS_IS	20	r	**High Side Driver 1 Drain Source Monitoring Interrupt Status in OFF-State** 0_B **no short on external FET**, no short detected. 1_B **short on external FET detected**, short detected.
RES	19:15	r	**Reserved** Always read as 0
LS2_OC_IS	14	r	**External Low Side 2 FET Over-current Interrupt Status** 0_B **no Over-current**, no over-current Condition occurred. 1_B **Over-current**, over-current occurred; switch is automatically shutdown.
LS2_DS_STS	13	r	**Low Side Driver 2 Drain Source Monitoring Status in OFF-State** 0_B **no short on external FET**, no short detected. 1_B **short on external FET detected**, short detected.
LS2_DS_IS	12	r	**Low Side Driver 2 Drain Source Monitoring Interrupt Status in OFF-State** 0_B **no short on external FET**, no short detected. 1_B **short on external FET detected**, short detected.
RES	11:7	r	**Reserved** Always read as 0
LS1_OC_IS	6	r	**External Low Side 1 FET Over-current Interrupt Status** 0_B **no Over-current**, no over-current Condition occurred. 1_B **Over-current**, over-current occurred; switch is automatically shutdown.
LS1_DS_STS	5	r	**Low Side Driver 1 Drain Source Monitoring Status in OFF-State** 0_B **no short on external FET**, no short detected. 1_B **short on external FET detected**, short detected.
LS1_DS_IS	4	r	**Low Side Driver 1 Drain Source Monitoring Interrupt Status in OFF-State** 0_B **no short on external FET**, no short detected. 1_B **short on external FET detected**, short detected.
RES	3:2	r	**Reserved** Always read as 0

Field	Bits	Type	Description
HB2_ASEQ_IS	1	r	**Half Bridge 2 Adaptive Sequencer Interrupt Status** *Note: Interrupt is set on any HB2 Error reported in **BDRV_ASEQSTS*** 0_B **no error in SEQ**, no sequencer Error detected. 1_B **error in SEQ**, sequencer Error detected.
HB1_ASEQ_IS	0	r	**Half Bridge 1 Adaptive Sequencer Interrupt Status** *Note: Interrupt is set on any HB1 Error reported in **BDRV_ASEQSTS*** 0_B **no error in SEQ**, no sequencer Error detected. 1_B **error in SEQ**, sequencer Error detected.

Table 539 RESET of BDRV_IRQS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

H-Bridge (Half Bridge) Driver Interrupt Status Clear Register

BDRV_IRQCLR	Offset	Reset Value
H-Bridge Driver Interrupt Status Clear Register	$F4_H$	see Table 540

31	30	29	28	27	26	25	24
SEQ_ERR_ISC	HS2_OC_ISC	HS2_DS_SC	HS2_DS_ISC	RES			
w	w	w	w	r			
23	22	21	20	19	18	17	16
RES	HS1_OC_ISC	HS1_DS_SC	HS1_DS_ISC	RES			
r	w	w	w	r			
15	14	13	12	11	10	9	8
RES	LS2_OC_ISC	LS2_DS_SC	LS2_DS_ISC	RES			
r	w	w	w	r			
7	6	5	4	3	2	1	0
RES	LS1_OC_ISC	LS1_DS_SC	LS1_DS_ISC	RES		HB2_ASEQ_ISC	HB1_ASEQ_ISC
r	w	w	w	r		w	w

Field	Bits	Type	Description
SEQ_ERR_ISC	31	w	**Driver Sequence Error Status Clear** 0_B **no Clear**, 1_B **Clear**,
HS2_OC_ISC	30	w	**External High Side 2 FET Over-current Status Clear** 0_B **no Clear**, 1_B **Clear**,
HS2_DS_SC	29	w	**High Side Driver 2 Drain Source Monitoring Status Clear in OFF-State** 0_B **no Clear**, 1_B **Clear**,
HS2_DS_ISC	28	w	**High Side Driver 2 Drain Source Monitoring Interrupt Status Clear in OFF-State** 0_B **no Clear**, 1_B **Clear**,
RES	27:23	r	**Reserved** Always read as 0
HS1_OC_ISC	22	w	**External High Side 1 FET Over-current Status Clear** 0_B **no Clear**, 1_B **Clear**,
HS1_DS_SC	21	w	**High Side Driver 1 Drain Source Monitoring Status Clear in OFF-State** 0_B **no Clear**, 1_B **Clear**,
HS1_DS_ISC	20	w	**High Side Driver 1 Drain Source Monitoring Interrupt Status Clear in OFF-State** 0_B **no Clear**, 1_B **Clear**,
RES	19:15	r	**Reserved** Always read as 0
LS2_OC_ISC	14	w	**External Low Side 2 FET Over-current Status Clear** 0_B **no Clear**, 1_B **Clear**,
LS2_DS_SC	13	w	**Low Side Driver 2 Drain Source Monitoring Status Clear in OFF-State** 0_B **no Clear**, 1_B **Clear**,
LS2_DS_ISC	12	w	**Low Side Driver 2 Drain Source Monitoring Interrupt Status Clear in OFF-State** 0_B **no Clear**, 1_B **Clear**,
RES	11:7	r	**Reserved** Always read as 0

Field	Bits	Type	Description
LS1_OC_ISC	6	w	**External Low Side 1 FET Over-current Status Clear** 0_B **no Clear**, 1_B **Clear**,
LS1_DS_SC	5	w	**Low Side Driver 1 Drain Source Monitoring Status Clear in OFF-State** 0_B **no Clear**, 1_B **Clear**,
LS1_DS_ISC	4	w	**Low Side Driver 1 Drain Source Monitoring Interrupt Status Clear in OFF-State** 0_B **no Clear**, 1_B **Clear**,
RES	3:2	r	**Reserved** Always read as 0
HB2_ASEQ_ISC	1	w	**Half Bridge 2 Adaptive Sequencer Interrupt Status Clear** 0_B **no Clear**, 1_B **Clear**,
HB1_ASEQ_ISC	0	w	**Half Bridge 1 Adaptive Sequencer Interrupt Status Clear** 0_B **no Clear**, 1_B **Clear**,

Table 540 RESET of BDRV_IRQCLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

H-Bridge (Half Bridge) Driver Interrupt Enable Register

BDRV_IRQEN	Offset	Reset Value
H-Bridge Driver Control	$F8_H$	see Table 541

31	30	29	28	27			24
SEQ_ERR_IEN	HS2_OC_IEN	RES	HS2_DS_IEN	RES			
rw	rw	r	rw	r			
23	22	21	20	19			16
RES	HS1_OC_IEN	RES	HS1_DS_IEN	RES			
r	rw	r	rw	r			
15	14	13	12	11			8
RES	LS2_OC_IEN	RES	LS2_DS_IEN	RES			
r	rw	r	rw	r			
7	6	5	4	3	2	1	0
RES	LS1_OC_IEN	RES	LS1_DS_IEN	RES		HB2_ASEQ_IEN	HB1_ASEQ_IEN
r	rw	r	rw	r		rw	rw

Field	Bits	Type	Description
SEQ_ERR_IEN	31	rw	**Driver Sequence Error Interrupt Enable** 0_B **disable**, 1_B **enable**,
HS2_OC_IEN	30	rw	**External High Side 2 FET Over-current Interrupt Enable** 0_B **disable**, 1_B **enable**,
RES	29	r	**Reserved** Always read as 0
HS2_DS_IEN	28	rw	**High Side Driver 2 Drain Source Monitoring Interrupt Enable in OFF-State** 0_B **disable**, 1_B **enable**,
RES	27:23	r	**Reserved** Always read as 0
HS1_OC_IEN	22	rw	**External High Side 1 FET Over-current Interrupt Enable** 0_B **disable**, 1_B **enable**,
RES	21	r	**Reserved** Always read as 0
HS1_DS_IEN	20	rw	**High Side Driver 1 Drain Source Monitoring Interrupt Enable in OFF-State** 0_B **disable**, 1_B **enable**,
RES	19:15	r	**Reserved** Always read as 0

Field	Bits	Type	Description
LS2_OC_IEN	14	rw	**External Low Side 2 FET Over-current Interrupt Enable** 0_B **disable**, 1_B **enable**,
RES	13	r	**Reserved** Always read as 0
LS2_DS_IEN	12	rw	**Low Side Driver 2 Drain Source Monitoring Interrupt Enable in OFF-State** 0_B **disable**, 1_B **enable**,
RES	11:7	r	**Reserved** Always read as 0
LS1_OC_IEN	6	rw	**External Low Side 1 FET Over-current Interrupt Enable** 0_B **disable**, 1_B **enable**,
RES	5	r	**Reserved** Always read as 0
LS1_DS_IEN	4	rw	**Low Side Driver 1 Drain Source Monitoring Interrupt Enable in OFF-State** 0_B **disable**, 1_B **enable**,
RES	3:2	r	**Reserved** Always read as 0
HB2_ASEQ_IEN	1	rw	**Half Bridge 2 Adaptive Sequencer Interrupt Enable** 0_B **disable**, 1_B **enable**,
HB1_ASEQ_IEN	0	rw	**Half Bridge 1 Adaptive Sequencer Interrupt Enable** 0_B **disable**, 1_B **enable**,

Table 541 RESET of BDRV_IRQEN

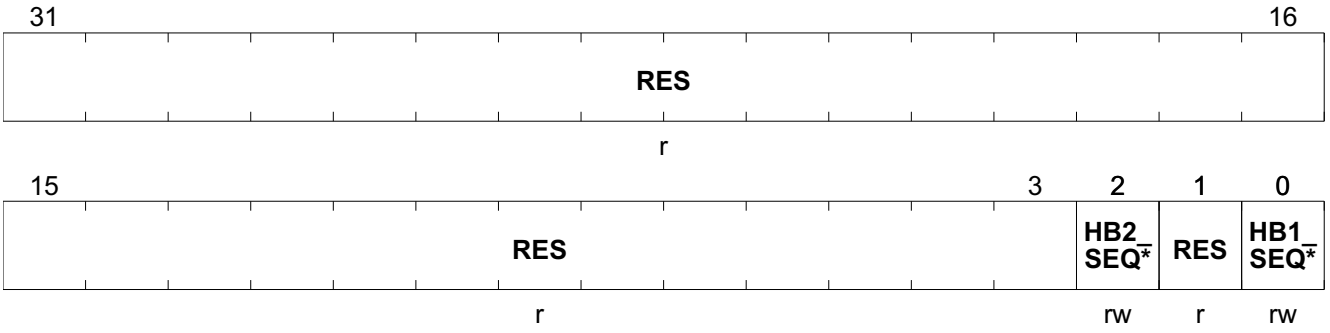
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

27.4.2 Sequencer Configuration Registers

As the sequencer functionality is only available per half bridge, the corresponding driver has to be assigned to it. The following register is used to map the sequencer functionality to the PWM operated driver.

Slewrate Sequencer Mapping Register

BDRV_SEQMAP	Offset	Reset Value
Slewrate Sequencer Mapping Register	10_H	see Table 542



Field	Bits	Type	Description
RES	31:3	r	**Reserved** Always read as 0
HB2_SEQMAP	2	rw	**Half Bridge 2 Sequencer Mapping** *Note: as the sequencer and adaptive sequencer driver functionality is only available per half bridge, this register is dedicated for the assignment to LS2 or HS2.* 0_H **LS2**, slew rate sequencer is mapped to LS2 1_H **HS2**, slew rate sequencer is mapped to HS2
RES	1	r	**Reserved** Always read as 0
HB1_SEQMAP	0	rw	**Half Bridge 1 Sequencer Mapping** *Note: as the sequencer and adaptive sequencer driver functionality is only available per half bridge, this register is dedicated for the assignment to LS1 or HS1.* 0_H **LS1**, slew rate sequencer is mapped to LS1 1_H **HS1**, slew rate sequencer is mapped to HS1

Table 542 RESET of BDRV_SEQMAP

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

27.4.3 Half Bridge 1 - Slew Rate Configuration Registers for Switch-Off/On.

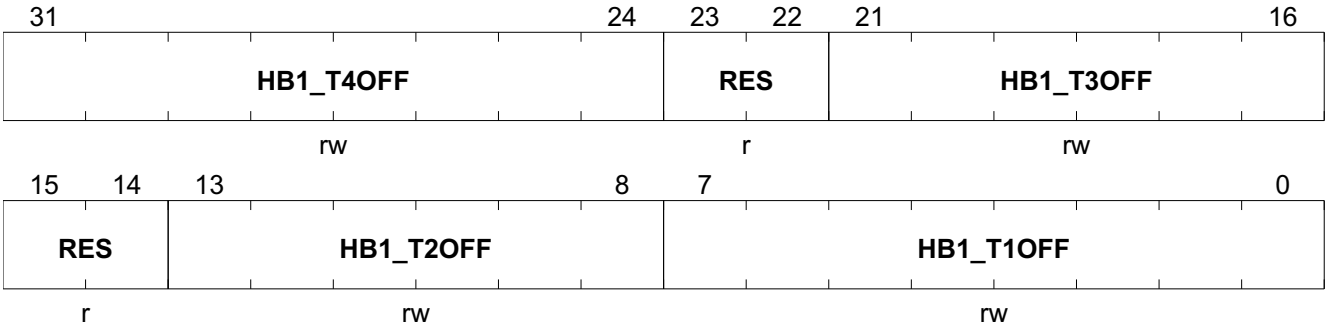
The switch off/on behaviour of the half bridge 1 drivers can be configured by two registers:

- configuration register for the discharge timing with four current settings for the four phases during switch-off
- configuration register for the discharge current with four current settings for the four phases during switch-off
- configuration register for the charge timing with four current settings for the four phases during switch-on
- configuration register for the charge current with four current settings for the four phases during switch-on

Note: time values are given for BRDRV_CLK = 20 MHz

Turn-off Slewrate Sequencer HB1 Time Control

BDRV_OFFSEQHB1TC	Offset	Reset Value
Turn-off Slewrate Sequencer Half Bridge 1 Time Control	50_H	see Table 543



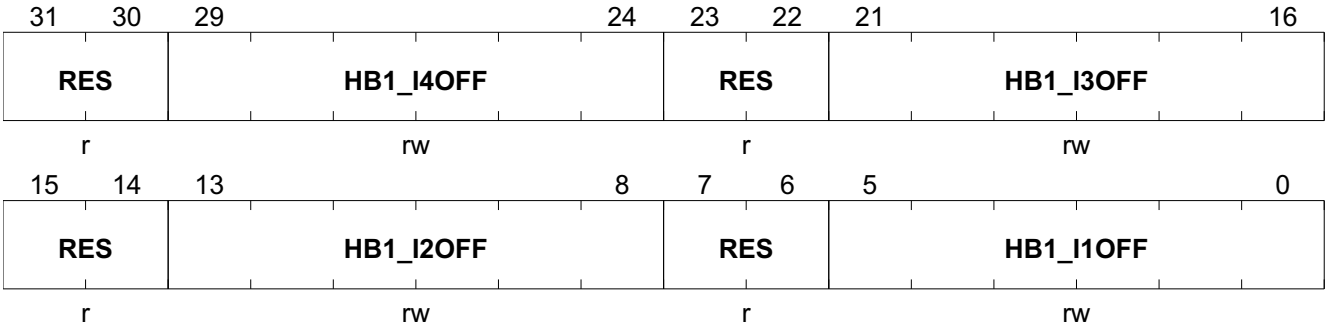
Field	Bits	Type	Description
HB1_T4OFF	31:24	rw	**Half Bridge 1-slew rate sequencer off-phase 4 time setting** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration
RES	23:22	r	**Reserved** Always read as 0
HB1_T3OFF	21:16	rw	**Half Bridge 1-slew rate sequencer off-phase 3 time setting** 0_H **50ns**, 50ns phase duration $3F_H$ **3.2us**, 3.2us phase duration
RES	15:14	r	**Reserved** Always read as 0
HB1_T2OFF	13:8	rw	**Half Bridge 1-slew rate sequencer off-phase 2 time setting** 0_H **50ns**, 50ns phase duration $3F_H$ **3.2us**, 3.2us phase duration
HB1_T1OFF	7:0	rw	**Half Bridge 1-slew rate sequencer off-phase 1 time setting** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration

Table 543 RESET of BDRV_OFFSEQHB1TC

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	03020206_H	RESET_TYPE_3		

Turn-off Slewrate Sequencer HB1 Current Control

BDRV_OFFSEQHB1IC	Offset	Reset Value
Turn-off Slewrate Sequencer Half Bridge 1 Current Control	54_H	see Table 544



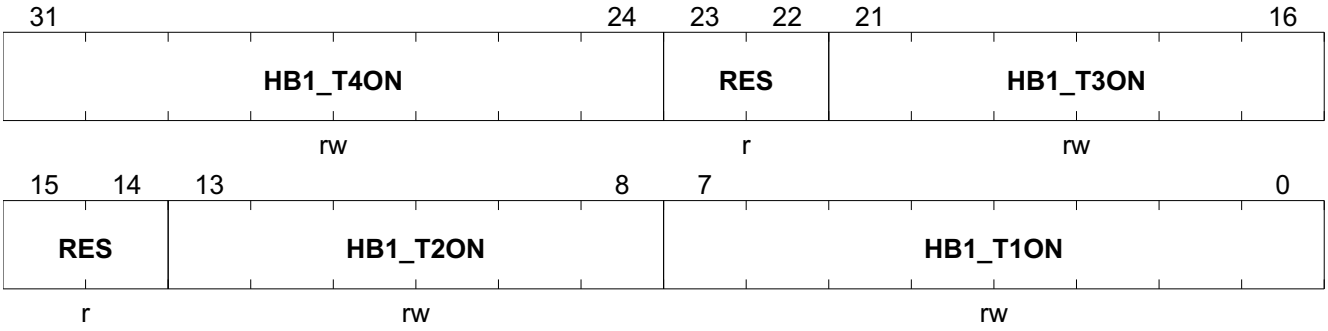
Field	Bits	Type	Description
RES	31:30	r	**Reserved** Always read as 0
HB1_I4OFF	29:24	rw	**Half Bridge 1-slew rate sequencer off-phase 4 current setting** 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$
RES	23:22	r	**Reserved** Always read as 0
HB1_I3OFF	21:16	rw	**Half Bridge 1-slew rate sequencer off-phase 3 current setting** 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$
RES	15:14	r	**Reserved** Always read as 0
HB1_I2OFF	13:8	rw	**Half Bridge 1-slew rate sequencer-off phase 2 current setting** 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$
RES	7:6	r	**Reserved** Always read as 0
HB1_I1OFF	5:0	rw	**Half Bridge 1-slew rate sequencer off-phase 1 current setting** 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$

Table 544 RESET of BDRV_OFFSEQHB1IC

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	20030320_H	RESET_TYPE_3		

Turn-on Slewrate Sequencer HB1 Time Control

BDRV_ONSEQHB1TC	Offset	Reset Value
Turn-on Slewrate Sequencer Half Bridge 1 Time Control	58_H	see Table 545



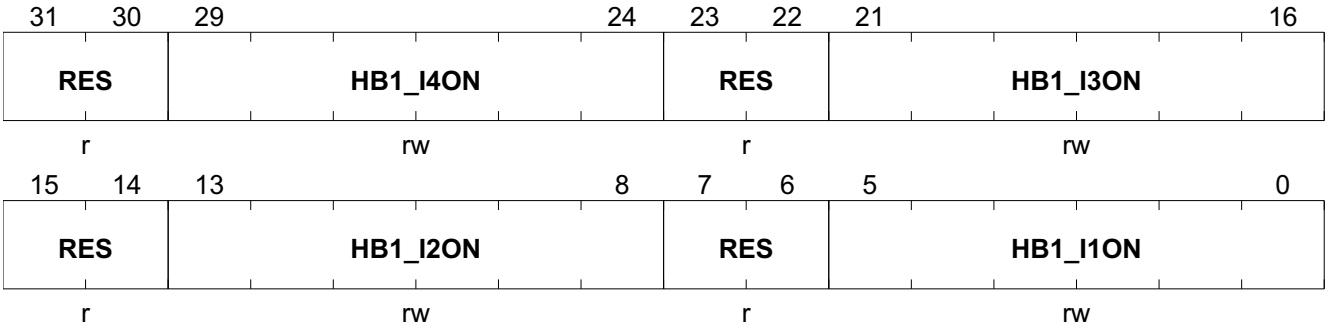
Field	Bits	Type	Description
HB1_T4ON	31:24	rw	**Half Bridge 1-slew rate sequencer on-phase 4 time setting** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration
RES	23:22	r	**Reserved** Always read as 0
HB1_T3ON	21:16	rw	**Half Bridge 1-slew rate sequencer on-phase 3 time setting** 0_H **50ns**, 50ns phase duration $3F_H$ **3.2us**, 3.2us phase duration
RES	15:14	r	**Reserved** Always read as 0
HB1_T2ON	13:8	rw	**Half Bridge 1-slew rate sequencer on-phase 2 time setting** 0_H **50ns**, 50ns phase duration $3F_H$ **3.2us**, 3.2us phase duration
HB1_T1ON	7:0	rw	**Half Bridge 1-slew rate sequencer on-phase 1 time setting** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration

Table 545 RESET of BDRV_ONSEQHB1TC

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	03020206_H	RESET_TYPE_3		

Turn-on Slewrate Sequencer HB1 Current Control

BDRV_ONSEQHB1IC	Offset	Reset Value
Turn-on Slewrate Sequencer Half Bridge 1 Current Control	$5C_H$	see Table 546



Field	Bits	Type	Description
RES	31:30	r	**Reserved** Always read as 0
HB1_I4ON	29:24	rw	**Half Bridge 1-slew rate sequencer on-phase 4 current setting** 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}
RES	23:22	r	**Reserved** Always read as 0
HB1_I3ON	21:16	rw	**Half Bridge 1-slew rate sequencer on-phase 3 current setting** 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}
RES	15:14	r	**Reserved** Always read as 0
HB1_I2ON	13:8	rw	**Half Bridge 1-slew rate sequencer on-phase 2 current setting** 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}
RES	7:6	r	**Reserved** Always read as 0
HB1_I1ON	5:0	rw	**Half Bridge 1-slew rate sequencer on-phase 1 current setting** 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}

Table 546 RESET of BDRV_ONSEQHB1IC

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	20030320_H	RESET_TYPE_3		

27.4.4 Half Bridge 1 - Slew Rate Configuration Registers for Active Freewheeling.

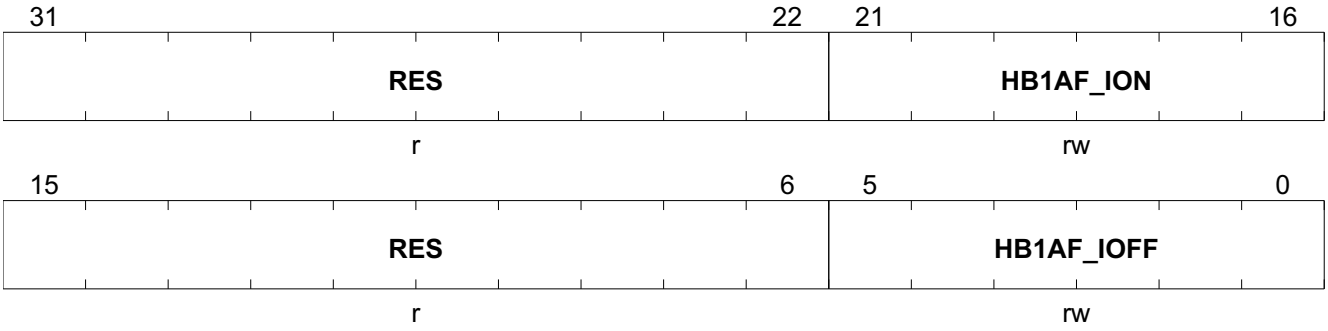
The switch off/on behaviour of the half bridge driver during active freewheeling can be configured by one dedicated gate current configuration register:

- configuration for the discharge current with one current settings for active freewheeling turn-off
- configuration for the charge current with one current settings for active freewheeling turn-on

Note: time values are given for BRDRV_CLK = 20 MHz

Slewrate Sequencer Active Freewheeling HB1 Current Control

BDRV_SEQAFHB1IC	Offset	Reset Value
Slewrate Sequencer-Active Freewheeling-Half Bridge 1 Current Control	64_H	see Table 547



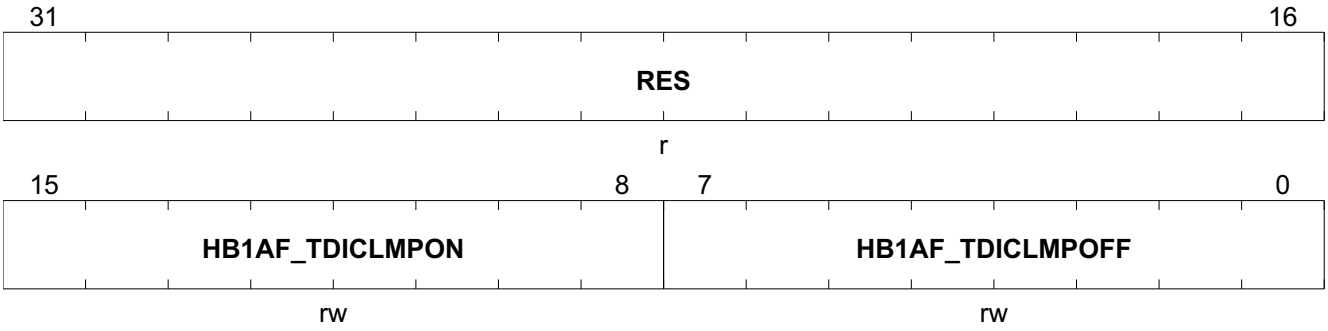
Field	Bits	Type	Description
RES	31:22	r	**Reserved** Always read as 0
HB1AF_ION	21:16	rw	**Half Bridge 1-active freewheeling-slew rate sequencer on-phase current setting** *Note: when a MOSFET operates in active freewheeling the corresponding driver is operated by a constant charge current with the value configured within this register.* 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$
RES	15:6	r	**Reserved** Always read as 0
HB1AF_IOFF	5:0	rw	**Half Bridge 1-active freewheeling-slew rate sequencer off-phase current setting** *Note: when a MOSFET operates in active freewheeling the corresponding driver is operated by a constant discharge current with the value configured within this register.* 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$

Table 547 RESET of BDRV_SEQAFHB1IC

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00200020_H	RESET_TYPE_3		

Slewrate Sequencer Active Freewheeling HB1 Clamping Current Delay

BDRV_SEQAFHB1CD	Offset	Reset Value
Slewrate Sequencer-Active Freewheeling-Half Bridge 1 Clamping Current Delay	68_H	see Table 548



Field	Bits	Type	Description
RES	31:16	r	**Reserved** Always read as 0
HB1AF_TDICLMPON	15:8	rw	**Clamping current delay during active freewheeling for switch on** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration
HB1AF_TDICLMPOFF	7:0	rw	**Clamping current delay during active freewheeling for switch off** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration

Table 548 RESET of BDRV_SEQAFHB1CD

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00004040_H	RESET_TYPE_3		

27.4.5 Half Bridge 2 - Slew Rate Configuration Registers for Switch-Off/On

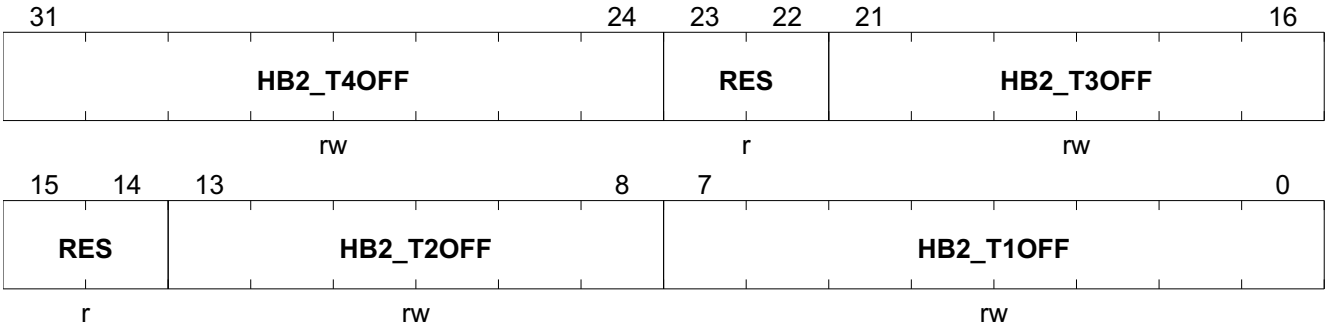
The switch off/on behaviour of the half bridge 2 driver can be configured by four registers:

- configuration register for the discharge timing with four current settings for the four phases during switch-off
- configuration register for the discharge current with four current settings for the four phases during switch-off
- configuration register for the charge current with four current settings for the four phases during switch-on
- configuration register for the charge current with four current settings for the four phases during switch-on

Note: time values are given for BRDRV_CLK = 20 MHz

Turn-off Slewrate Sequencer HB2 Time Control

BDRV_OFFSEQHB2TC	Offset	Reset Value
Turn-off Slewrate Sequencer Half Bridge 2 Time Control	70_H	see Table 549



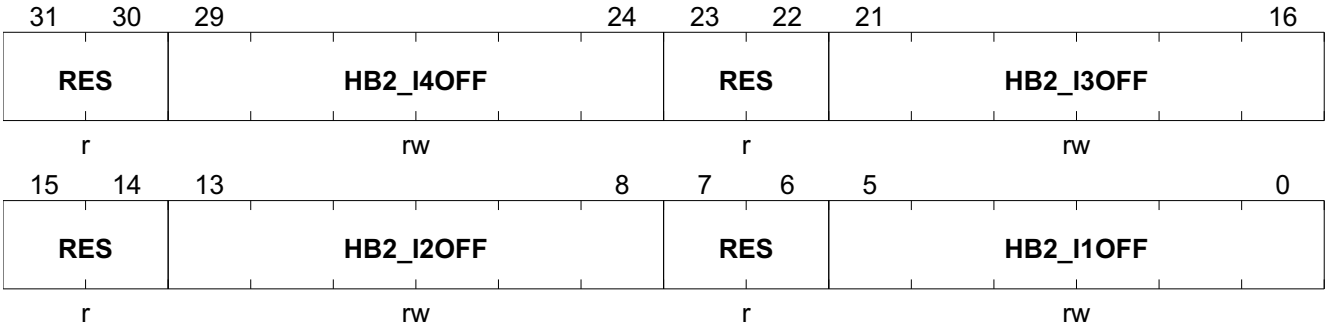
Field	Bits	Type	Description
HB2_T4OFF	31:24	rw	**Half Bridge 2-slew rate sequencer off-phase 4 time setting** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration
RES	23:22	r	**Reserved** Always read as 0
HB2_T3OFF	21:16	rw	**Half Bridge 2-slew rate sequencer off-phase 3 time setting** 0_H **50ns**, 50ns phase duration $3F_H$ **3.2us**, 3.2us phase duration
RES	15:14	r	**Reserved** Always read as 0
HB2_T2OFF	13:8	rw	**Half Bridge 2-slew rate sequencer off-phase 2 time setting** 0_H **50ns**, 50ns phase duration $3F_H$ **3.2us**, 3.2us phase duration
HB2_T1OFF	7:0	rw	**Half Bridge 2-slew rate sequencer off-phase 1 time setting** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration

Table 549 RESET of BDRV_OFFSEQHB2TC

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	03020206_H	RESET_TYPE_3		

Turn-off Slewrate Sequencer HB2 Current Control

BDRV_OFFSEQHB2IC	Offset	Reset Value
Turn-off Slewrate Sequencer Half Bridge 2 Current Control	74_H	see Table 550



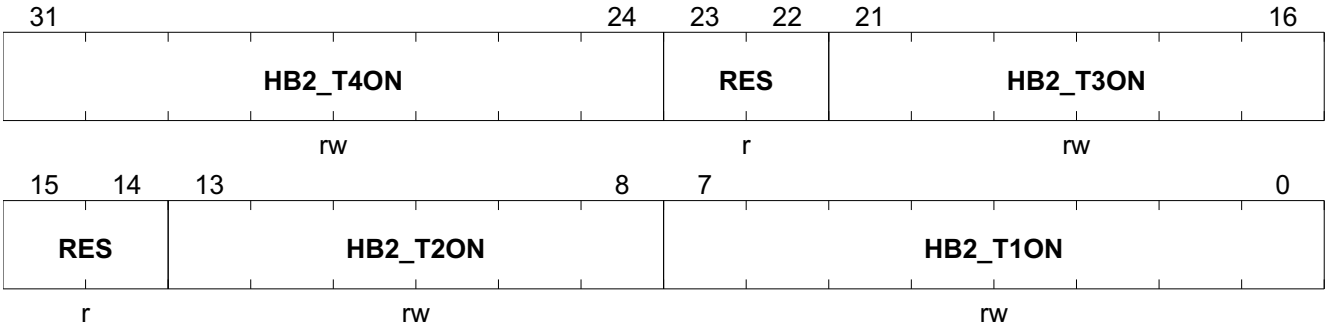
Field	Bits	Type	Description
RES	31:30	r	**Reserved** Always read as 0
HB2_I4OFF	29:24	rw	**Half Bridge 2-slew rate sequencer off-phase 4 current setting** 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$
RES	23:22	r	**Reserved** Always read as 0
HB2_I3OFF	21:16	rw	**Half Bridge 2-slew rate sequencer off-phase 3 current setting** 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$
RES	15:14	r	**Reserved** Always read as 0
HB2_I2OFF	13:8	rw	**Half Bridge 2-slew rate sequencer off-phase 2 current setting** 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$
RES	7:6	r	**Reserved** Always read as 0
HB2_I1OFF	5:0	rw	**Half Bridge 2-slew rate sequencer off-phase 1 current setting** 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$

Table 550 RESET of BDRV_OFFSEQHB2IC

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	20030320_H	RESET_TYPE_3		

Turn-on Slewrate Sequencer HB2 Time Control

BDRV_ONSEQHB2TC	Offset	Reset Value
Turn-on Slewrate Sequencer Half Bridge 2 Time Control	78_H	see Table 551



Field	Bits	Type	Description
HB2_T4ON	31:24	rw	**Half Bridge 2-slew rate sequencer on-phase 4 time setting** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration
RES	23:22	r	**Reserved** Always read as 0
HB2_T3ON	21:16	rw	**Half Bridge 2-slew rate sequencer on-phase 3 time setting** 0_H **50ns**, 50ns phase duration $3F_H$ **3.2us**, 3.2us phase duration
RES	15:14	r	**Reserved** Always read as 0
HB2_T2ON	13:8	rw	**Half Bridge 2-slew rate sequencer on-phase 2 time setting** 0_H **50ns**, 50ns phase duration $3F_H$ **3.2us**, 3.2us phase duration
HB2_T1ON	7:0	rw	**Half Bridge 2-slew rate sequencer on-phase 1 time setting** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration

Table 551 RESET of BDRV_ONSEQHB2TC

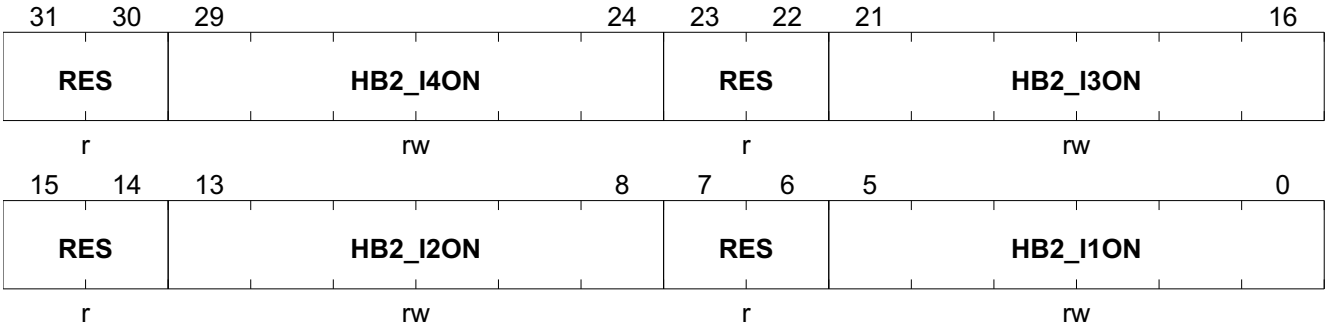
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	03020206_H	RESET_TYPE_3		

Turn-on Slewrate Sequencer HS1 Current Control

BDRV_ONSEQHB2IC — **Turn-on Slewrate Sequencer Half Bridge 2 Current Control**

Offset $7C_H$

Reset Value see Table 552



Field	Bits	Type	Description
RES	31:30	r	**Reserved** Always read as 0
HB2_I4ON	29:24	rw	**Half Bridge 2-slew rate sequencer on-phase 4 current setting** 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}
RES	23:22	r	**Reserved** Always read as 0
HB2_I3ON	21:16	rw	**Half Bridge 2-slew rate sequencer on-phase 3 current setting** 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}
RES	15:14	r	**Reserved** Always read as 0
HB2_I2ON	13:8	rw	**Half Bridge 2-slew rate sequencer on-phase 2 current setting** 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}
RES	7:6	r	**Reserved** Always read as 0
HB2_I1ON	5:0	rw	**Half Bridge 2-slew rate sequencer on-phase 1 current setting** 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}

Table 552 RESET of BDRV_ONSEQHB2IC

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	20030320_H	RESET_TYPE_3		

27.4.6 Half Bridge 2 - Slew Rate Configuration Registers for Active Freewheeling

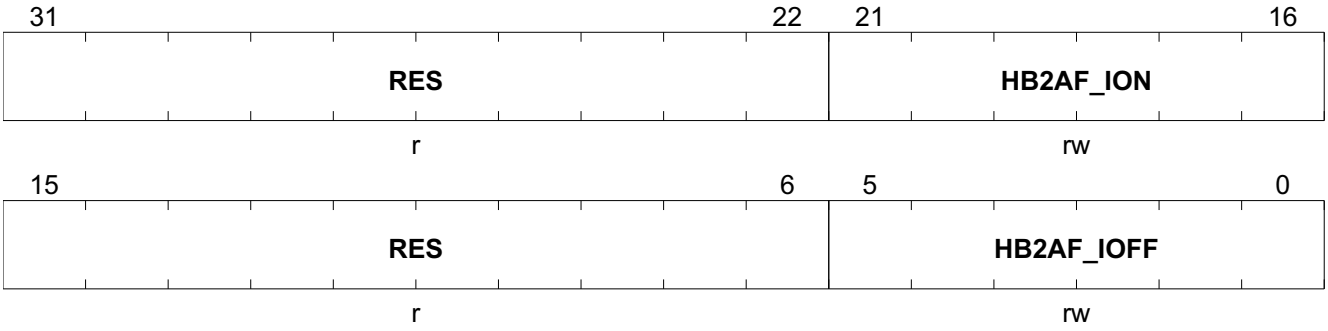
The switch off/on behaviour of the half bridge driver during active freewheeling can be configured by one dedicated gate current configuration register:

- configuration for the discharge current with one current settings for active freewheeling turn-off
- configuration for the charge current with one current settings for active freewheeling turn-on

Note: time values are given for BRDRV_CLK = 20 MHz

Slewrate Sequencer Active Freewheeling HB2 Current Control

BDRV_SEQAFHB2IC	Offset	Reset Value
Slewrate Sequencer-Active Freewheeling-Half Bridge 2 Current Control	84_H	see Table 553



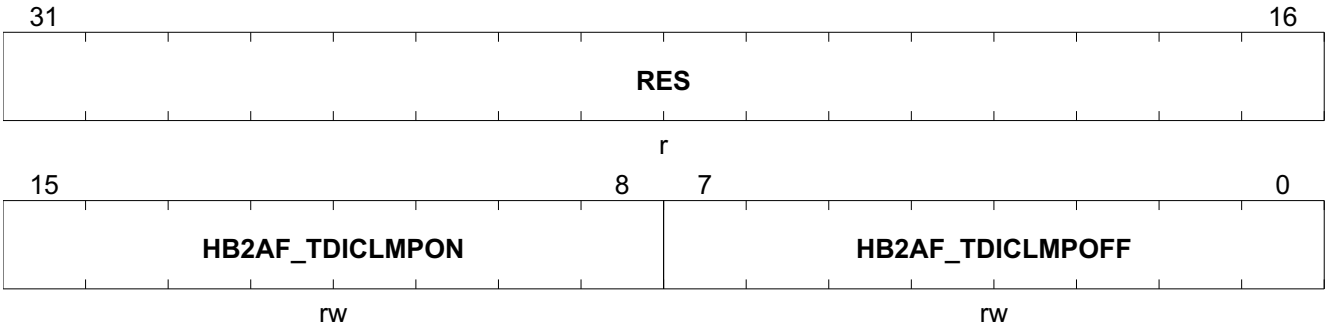
Field	Bits	Type	Description
RES	31:22	r	**Reserved** Always read as 0
HB2AF_ION	21:16	rw	**Half Bridge 2-active freewheeling-slew rate sequencer on-phase current setting** *Note: when a MOSFET operates in active freewheeling the corresponding driver is operated by a constant charge current with the value configured within this register.* 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$
RES	15:6	r	**Reserved** Always read as 0
HB2AF_IOFF	5:0	rw	**Half Bridge 2-active freewheeling-slew rate sequencer off-phase current setting** *Note: when a MOSFET operates in active freewheeling the corresponding driver is operated by a constant discharge current with the value configured within this register.* 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$

Table 553 RESET of BDRV_SEQAFHB2IC

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00200020_H	RESET_TYPE_3		

Slewrate Sequencer Active Freewheeling HB2 Clamping Current Delay

BDRV_SEQAFHB2CD	Offset	Reset Value
Slewrate Sequencer-Active Freewheeling-Half Bridge 2 Clamping Current Delay	88_H	see Table 554



Field	Bits	Type	Description
RES	31:16	r	**Reserved** Always read as 0
HB2AF_TDICLMPON	15:8	rw	**Clamping current delay during active freewheeling for switch on** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration
HB2AF_TDICLMPOFF	7:0	rw	**Clamping current delay during active freewheeling for switch off** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration

Table 554 RESET of BDRV_SEQAFHB2CD

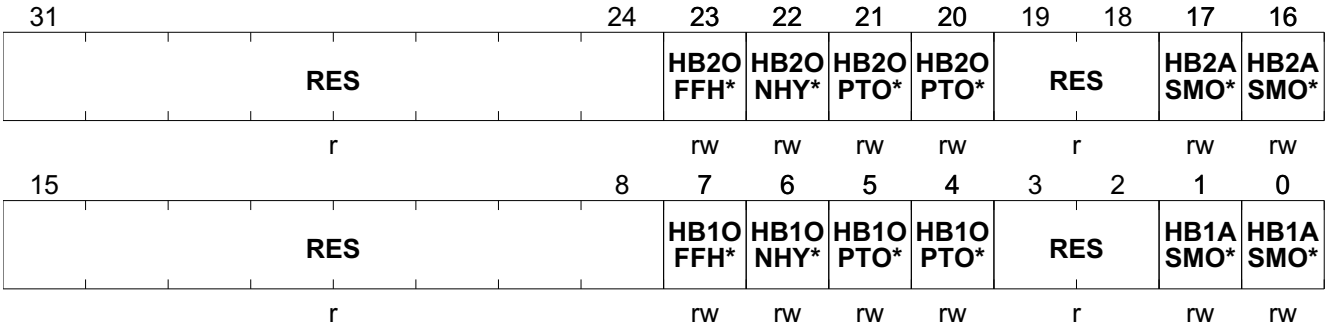
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00400040_H	RESET_TYPE_3		

27.4.7 Adaptive Slew Rate Sequencer Control and Status Registers

The following registers are used to control the adaptive sequencer mode and are reflecting the status of the adaptive sequencer.

Adaptive Slewrate Sequencer Control Register

BDRV_ASEQC	Offset	Reset Value
Adaptive Slewrate Sequencer Control Register	90_H	see Table 555



Field	Bits	Type	Description
RES	31:24	r	**Reserved** Always read as 0
HB2OFFHYSTEN	23	rw	**Half Bridge 2 Optimizer Hysteresis for Switch Off Enable Bit** 0_H **OFF**, current optimizer hysteresis is not enabled 1_H **ON**, current optimizer hysteresis is enabled
HB2ONHYSTEN	22	rw	**Half Bridge 2 Optimizer Hysteresis for Switch On Enable Bit** 0_H **OFF**, current optimizer hysteresis is not enabled 1_H **ON**, current optimizer hysteresis is enabled
HB2OPTOFFACT	21	rw	**Half Bridge 2 Optimizer for Switch Off Active Bit** 0_H **OFF**, current optimizer is not active 1_H **ON**, current optimizer is active
HB2OPTONACT	20	rw	**Half Bridge 2 Optimizer for Switch On Active Bit** 0_H **OFF**, current optimizer is not active 1_H **ON**, current optimizer is active
RES	19:18	r	**Reserved** Always read as 0
HB2ASMOFFEN	17	rw	**Half Bridge 2 Adaptive Sequencer Mode for Switch Off Enable** 0_H **Disable**, adaptive slew rate sequencer is disabled 1_H **Enable**, adaptive slew rate sequencer is enabled
HB2ASMONEN	16	rw	**Half Bridge 2 Adaptive Sequencer Mode for Switch On Enable** 0_H **Disable**, adaptive slew rate sequencer is disabled 1_H **Enable**, adaptive slew rate sequencer is enabled
RES	15:8	r	**Reserved** Always read as 0

Field	Bits	Type	Description
HB1OFFHYSTEN	7	rw	**Half Bridge 1 Optimizer Hysteresis for Switch Off Enable Bit** 0_H **OFF**, current optimizer hysteresis is not enabled 1_H **ON**, current optimizer hysteresis is enabled
HB1ONHYSTEN	6	rw	**Half Bridge 1 Optimizer Hysteresis for Switch On Enable Bit** 0_H **OFF**, current optimizer hysteresis is not enabled 1_H **ON**, current optimizer hysteresis is enabled
HB1OPTOFFACT	5	rw	**Half Bridge 1 Optimizer for Switch Off Active Bit** 0_H **OFF**, current optimizer is not active 1_H **ON**, current optimizer is active
HB1OPTONACT	4	rw	**Half Bridge 1 Optimizer for Switch On Active Bit** 0_H **OFF**, current optimizer is not active 1_H **ON**, current optimizer is active
RES	3:2	r	**Reserved** Always read as 0
HB1ASMOFFEN	1	rw	**Half Bridge 1 Adaptive Sequencer Mode for Switch Off Enable** 0_H **Disable**, adaptive slew rate sequencer is disabled 1_H **Enable**, adaptive slew rate sequencer is enabled
HB1ASMONEN	0	rw	**Half Bridge 1 Adaptive Sequencer Mode for Switch On Enable** 0_H **Disable**, adaptive slew rate sequencer is disabled 1_H **Enable**, adaptive slew rate sequencer is enabled

Table 555 RESET of BDRV_ASEQC

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Adaptive Slewrate Sequencer Status Register

BDRV_ASEQSTS	Offset	Reset Value
Adaptive Slewrate Sequencer Status Register	94_H	see Table 556

31	30	29					24	23	22	21	20	19	18	17	16
HB2ONMF	HB2OFFMF	RES						HB2I1ON*	HB2T12O*	HB2I1ON*	HB2T12O*	HB2I1OF*	HB2T1OF*	HB2I1OF*	HB2T1OF*
rc	rc	r						rc	rc	rc	rc	rc	rc	rc	rc
15	14	13					8	7	6	5	4	3	2	1	0
HB1ONMF	HB1OFFMF	RES						HB1I1ON*	HB1T12O*	HB1I1ON*	HB1T12O*	HB1I1OF*	HB1T1OF*	HB1I1OF*	HB1T1OF*
rc	rc	r						rc	rc	rc	rc	rc	rc	rc	rc

Field	Bits	Type	Description
HB2ONMF	31	rc	**Half Bridge 2- On Adaptive Mode Measurement Failure** 0_H **no Error**, No Measurement Failure 1_H **Error**, Measurement Failure
HB2OFFMF	30	rc	**Half Bridge 2- Off Adaptive Mode Measurement Failure** 0_H **no Error**, No Measurement Failure 1_H **Error**, Measurement Failure
RES	29:24	r	**Reserved** Always read as 0
HB2I1ONMIN	23	rc	**Half Bridge 2-I1 On Min Value reached** 0_H **no Error**, Min Value not reached 1_H **Error**, Min Value reached
HB2T12ONMIN	22	rc	**Half Bridge 2-T12 On Min Value reached** 0_H **no Error**, Min Value not reached 1_H **Error**, Min Value reached
HB2I1ONMAX	21	rc	**Half Bridge 2-I1 On Max Value reached** 0_H **no Error**, Max Value not reached 1_H **Error**, Max Value reached
HB2T12ONMAX	20	rc	**Half Bridge 2-T12 On Max Value reached** 0_H **no Error**, Max Value not reached 1_H **Error**, Max Value reached
HB2I1OFFMIN	19	rc	**Half Bridge 2-I1 Off Min Value reached** 0_H **no Error**, Min Value not reached 1_H **Error**, Min Value reached
HB2T1OFFMIN	18	rc	**Half Bridge 2-T1 Off Min Value reached** 0_H **no Error**, Min Value not reached 1_H **Error**, Min Value reached
HB2I1OFFMAX	17	rc	**Half Bridge 2-I1 Off Max Value reached** 0_H **no Error**, Max Value not reached 1_H **Error**, Max Value reached

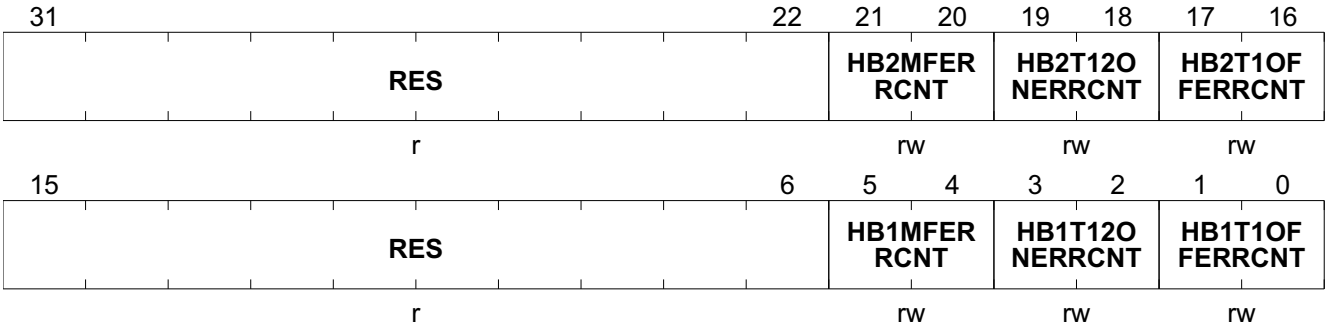
Field	Bits	Type	Description
HB2T1OFFMAX	16	rc	**Half Bridge 2-T1 Off Max Value reached** 0_H **no Error**, Max Value not reached 1_H **Error**, Max Value reached
HB1ONMF	15	rc	**Half Bridge 1-On Adaptive Mode Measurement Failure** 0_H **no Error**, No Measurement Failure 1_H **Error**, Measurement Failure
HB1OFFMF	14	rc	**Half Bridge 1-Off Adaptive Mode Measurement Failure** 0_H **no Error**, No Measurement Failure 1_H **Error**, Measurement Failure
RES	13:8	r	**Reserved** Always read as 0
HB1I1ONMIN	7	rc	**Half Bridge 1-I1 On Min Value reached** 0_H **no Error**, Min Value not reached 1_H **Error**, Min Value reached
HB1T12ONMIN	6	rc	**Half Bridge 1-T12 On Min Value reached** 0_H **no Error**, Min Value not reached 1_H **Error**, Min Value reached
HB1I1ONMAX	5	rc	**Half Bridge 1-I1 On Max Value reached** 0_H **no Error**, Max Value not reached 1_H **Error**, Max Value reached
HB1T12ONMAX	4	rc	**Half Bridge 1-T12 On Max Value reached** 0_H **no Error**, Max Value not reached 1_H **Error**, Max Value reached
HB1I1OFFMIN	3	rc	**Half Bridge 1-I1 Off Min Value reached** 0_H **no Error**, Min Value not reached 1_H **Error**, Min Value reached
HB1T1OFFMIN	2	rc	**Half Bridge 1-T1 Off Min Value reached** 0_H **no Error**, Min Value not reached 1_H **Error**, Min Value reached
HB1I1OFFMAX	1	rc	**Half Bridge 1-I1 Off Max Value reached** 0_H **no Error**, Max Value not reached 1_H **Error**, Max Value reached
HB1T1OFFMAX	0	rc	**Half Bridge 1-T1 Off Max Value reached** 0_H **no Error**, Max Value not reached 1_H **Error**, Max Value reached

Table 556 RESET of BDRV_ASEQSTS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Adaptive Slewrate Sequencer Error Counter Control Register

BDRV_ASEQERRCNT	Offset	Reset Value
Adaptive Slewrate Sequencer Error Counter Control Register	$D8_H$	see Table 557



Field	Bits	Type	Description
RES	31:22	r	**Reserved** Always read as 0
HB2MFERRCNT	21:20	rw	**Half Bridge 2-Measurement Failure Error Counter Setting** 00_B **2 Events**, Error Flag is set after 2 Events 01_B **4 Events**, Error Flag is set after 4 Events 10_B **8 Events**, Error Flag is set after 8 Events 11_B **15 Events**, Error Flag is set after 15 Events
HB2T12ONERRCNT	19:18	rw	**Half Bridge 2-T12 On Error Counter Setting** 00_B **2 Events**, Error Flag is set after 2 Events 01_B **4 Events**, Error Flag is set after 4 Events 10_B **8 Events**, Error Flag is set after 8 Events 11_B **15 Events**, Error Flag is set after 15 Events
HB2T1OFFERRCNT	17:16	rw	**Half Bridge 2-T1 Off Error Counter Setting** 00_B **2 Events**, Error Flag is set after 2 Events 01_B **4 Events**, Error Flag is set after 4 Events 10_B **8 Events**, Error Flag is set after 8 Events 11_B **15 Events**, Error Flag is set after 15 Events
RES	15:6	r	**Reserved** Always read as 0
HB1MFERRCNT	5:4	rw	**Half Bridge 1-Measurement Failure Error Counter Setting** 00_B **2 Events**, Error Flag is set after 2 Events 01_B **4 Events**, Error Flag is set after 4 Events 10_B **8 Events**, Error Flag is set after 8 Events 11_B **15 Events**, Error Flag is set after 15 Events

Field	Bits	Type	Description
HB1T12ONERRCNT	3:2	rw	**Half Bridge 1-T12 On Error Counter Setting** 00_B **2 Events**, Error Flag is set after 2 Events 01_B **4 Events**, Error Flag is set after 4 Events 10_B **8 Events**, Error Flag is set after 8 Events 11_B **15 Events**, Error Flag is set after 15 Events
HB1T1OFFERRCNT	1:0	rw	**Half Bridge 1-T1 Off Error Counter Setting** 00_B **2 Events**, Error Flag is set after 2 Events 01_B **4 Events**, Error Flag is set after 4 Events 10_B **8 Events**, Error Flag is set after 8 Events 11_B **15 Events**, Error Flag is set after 15 Events

Table 557 RESET of BDRV_ASEQERRCNT

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

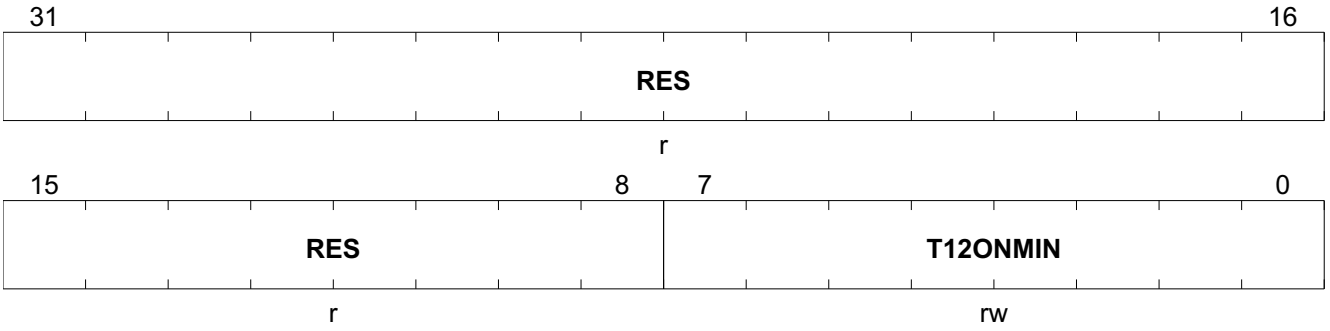
27.4.8 Adaptive Slew Rate Sequencer Configuration Registers

The following registers are used for configuration of the on and off slew rate of each MOSFET.

Note: time values are given for BRDRV_CLK = 20 MHz

Turn-on Adaptive Slewrate Sequencer Minimum Time Setting

BDRV_ONASEQTMIN	Offset	Reset Value
Turn ON Adaptive Slewrate Sequencer Minimum Time Setting	98_H	see Table 558



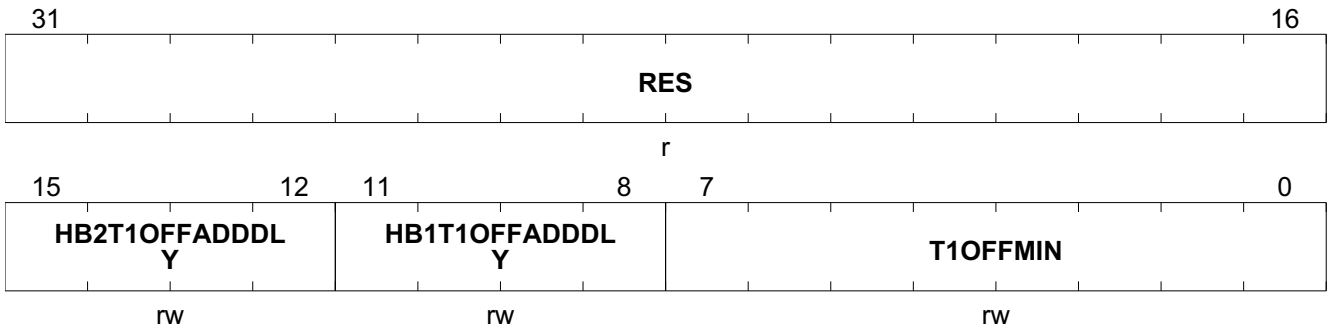
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Always read as 0
T12ONMIN	7:0	rw	**Slew rate sequencer on-phase 12 min. time setting** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration

Table 558 RESET of BDRV_ONASEQTMIN

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000001_H	RESET_TYPE_3		

Turn-off Adaptive Slewrate Sequencer Minimum Time Setting

BDRV_OFFASEQTMIN	Offset	Reset Value
Turn OFF Adaptive Slewrate Sequencer Minimum Time Setting	$9C_H$	see Table 559



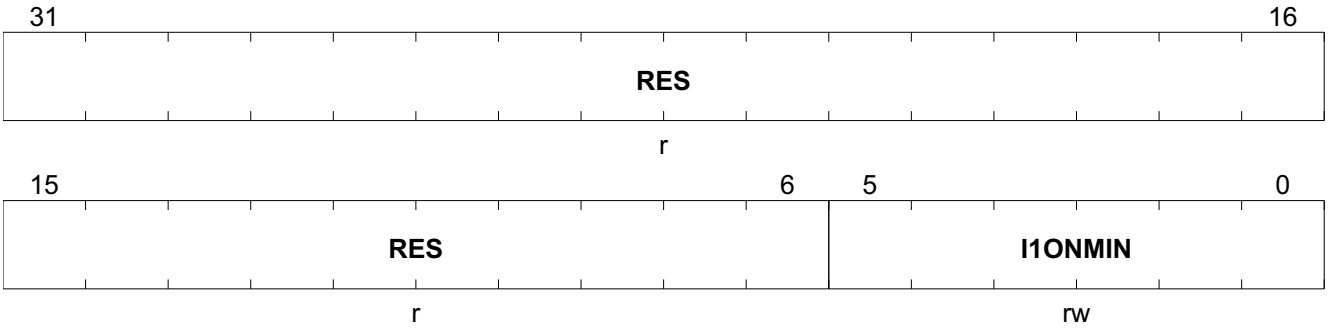
Field	Bits	Type	Description
RES	31:16	r	**Reserved** Always read as 0
HB2T1OFFADDDLY	15:12	rw	**HB2 adaptive sequencer T1OFF additional delay setting.** *Note: This setting is used in adaptive mode as additional delay to HB2T1OFF for adaptive regulation time. HB2I1OFF will be adapted to reach time measurement target of HB2T1OFF + HB2T1OFFADDDLY* 0_H **0ns**, 0ns added to HB2T1OFF F_H **750ns**, 750ns added to HB2T1OFF
HB1T1OFFADDDLY	11:8	rw	**HB1 adaptive sequencer T1OFF additional delay setting.** *Note: This setting is used in adaptive mode as additional delay to HB1T1OFF for adaptive regulation time. HB1I1OFF will be adapted to reach time measurement target of HB1T1OFF + HB1T1OFFADDDLY* 0_H **0ns**, 0ns added to HB1T1OFF F_H **750ns**, 750ns added to HB1T1OFF
T1OFFMIN	7:0	rw	**Slew rate sequencer off-phase 1 min. time setting** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration

Table 559 RESET of BDRV_OFFASEQTMIN

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000001_H	RESET_TYPE_3		

Turn-off Adaptive Slewrate Sequencer On Phase Minimum Current Setting

BDRV_ASEQIONMIN	Offset	Reset Value
Adaptive Slewrate Sequencer On Phase Minimum Current Setting	$A0_H$	see Table 560



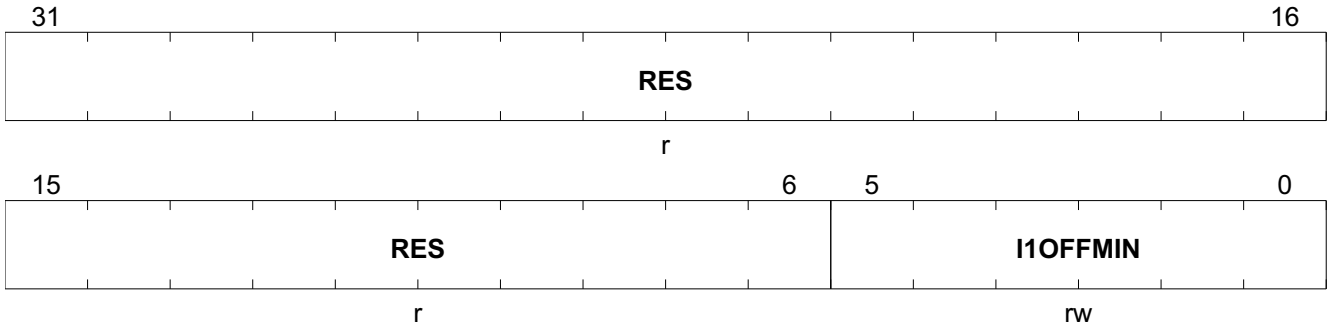
Field	Bits	Type	Description
RES	31:6	r	**Reserved** Always read as 0
I1ONMIN	5:0	rw	**Slew rate sequencer on-phase 1 min. current setting** *Note: this setting is used in adaptive mode as the lower limit for the optimization run.* 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}

Table 560 RESET of BDRV_ASEQIONMIN

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000001_H	RESET_TYPE_3		

Turn-off Adaptive Slewrate Sequencer Off Phase Minimum Current Setting

BDRV_ASEQIOFFMIN	Offset	Reset Value
Adaptive Slewrate Sequencer Off Phase Minimum Current Setting	$A4_H$	see Table 561



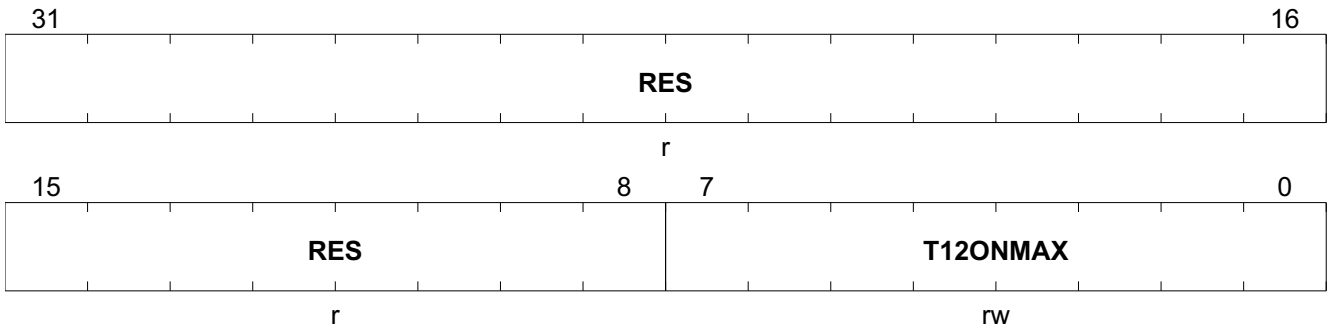
Field	Bits	Type	Description
RES	31:6	r	**Reserved** Always read as 0
I1OFFMIN	5:0	rw	**Slew rate sequencer off-phase 1 min. current setting** *Note: this setting is used in adaptive mode as the lower limit for the optimization run.* 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$

Table 561 RESET of BDRV_ASEQIOFFMIN

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000001_H	RESET_TYPE_3		

Turn-on Adaptive Slewrate Sequencer Maximum Time Setting

BDRV_ONASEQTMAX	Offset	Reset Value
Adaptive Slewrate On Sequencer Maximum Time Setting	$A8_H$	see Table 562



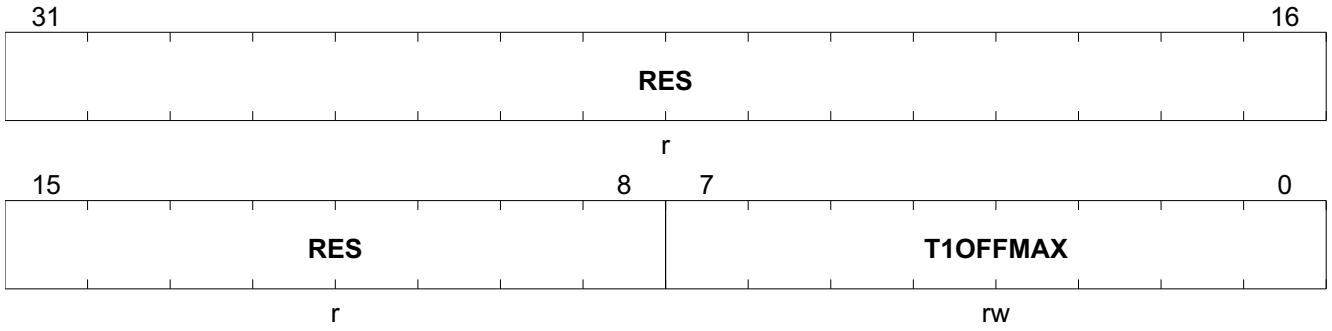
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Always read as 0
T12ONMAX	7:0	rw	**Slew rate sequencer on-phase 12 max. time setting** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration

Table 562 RESET of BDRV_ONASEQTMAX

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000040_H	RESET_TYPE_3		

Turn-off Adaptive Slewrate Sequencer Maximum Time Setting

BDRV_OFFASEQTMAX	Offset	Reset Value
Adaptive Slewrate Off Sequencer Maximum Time Setting	AC_H	see Table 563



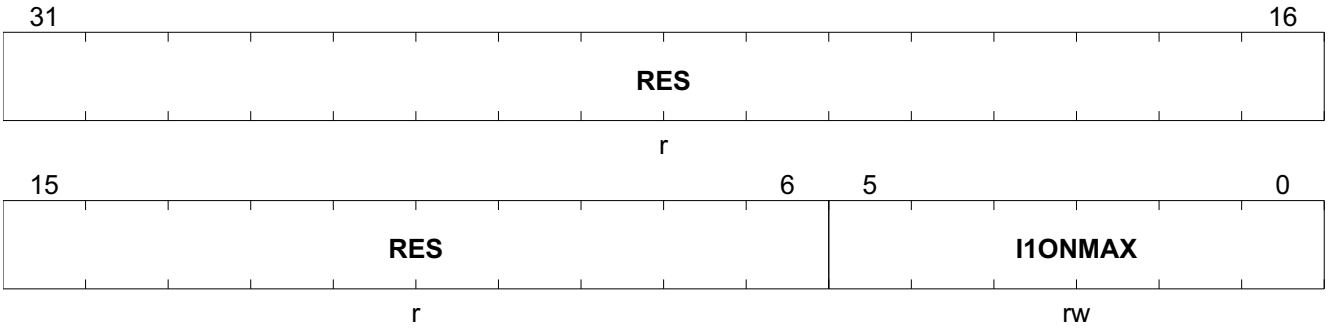
Field	Bits	Type	Description
RES	31:8	r	**Reserved** Always read as 0
T1OFFMAX	7:0	rw	**Slew rate sequencer off-phase 1 max. time setting** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration

Table 563 RESET of BDRV_OFFASEQTMAX

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000040_H	RESET_TYPE_3		

Turn-off Adaptive Slewrate Sequencer On Phase Minimum Current Setting

BDRV_ASEQIONMAX	Offset	Reset Value
Adaptive Slewrate Sequencer On Phase Maximum Current Setting	$B0_H$	see Table 564



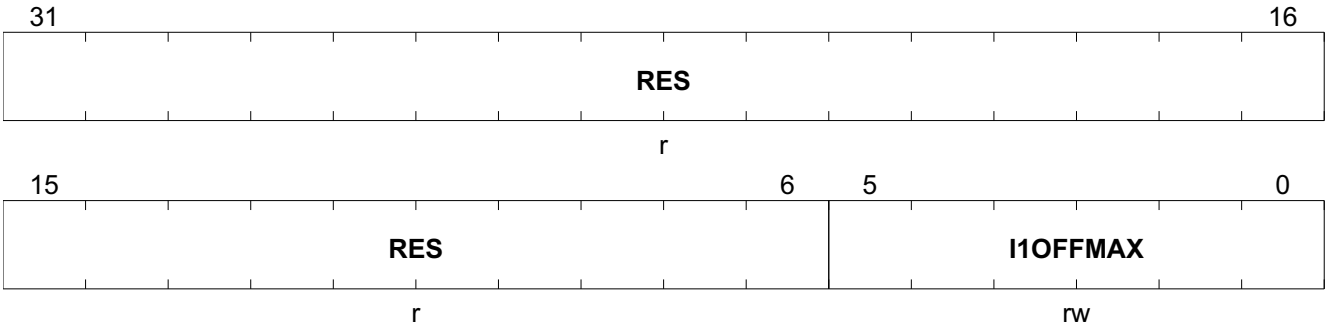
Field	Bits	Type	Description
RES	31:6	r	**Reserved** Always read as 0
I1ONMAX	5:0	rw	**Slew rate sequencer on-phase 1 max. current setting** *Note: this setting is used in adaptive mode as the upper limit for the optimization run.* 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}

Table 564 RESET of BDRV_ASEQIONMAX

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000030_H	RESET_TYPE_3		

Turn-off Adaptive Slewrate Sequencer Off Phase Minimum Current Setting

BDRV_ASEQIOFFMAX	Offset	Reset Value
Adaptive Slewrate Sequencer Off Phase Maximum Current Setting	$B4_H$	see Table 565



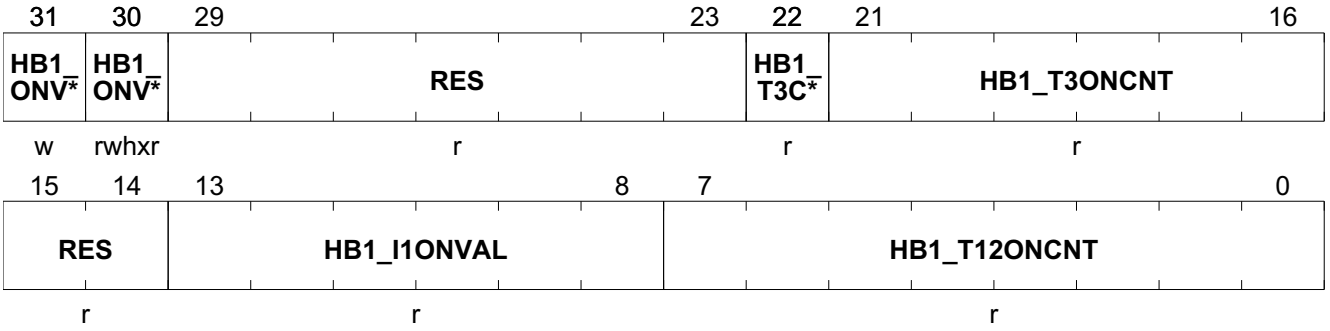
Field	Bits	Type	Description
RES	31:6	r	**Reserved** Always read as 0
I1OFFMAX	5:0	rw	**Slew rate sequencer off-phase 1 max. current setting** *Note: this setting is used in adaptive mode as the upper limit for the optimization run.* 0_H **min. current**, $I_{DISCHGmin}$ $3F_H$ **max. current**, $I_{DISCHGmax}$

Table 565 RESET of BDRV_ASEQIOFFMAX

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000030_H	RESET_TYPE_3		

Slewrate Adaptive Sequencers On Values for Half Bridge 1

BDRV_HB1ASEQONVAL	Offset	Reset Value
Half Bridge 1 Adaptive Sequencer On Values	$B8_H$	see Table 566



Field	Bits	Type	Description
HB1_ONVALVF_CLR	31	w	**Half Bridge 1-Turn on slew rate values Valid Flag - Clear.** 0_B **NOT CLEAR**, no clear of HB1_ONVALVF flag 1_B **CLEAR**, clear of HB1_ONVALVF flag
HB1_ONVALVF	30	rwhxr	**Half Bridge 1-Turn on slew rate values - Valid Flag.** 0_B **NOT VALID**, no new valid LS1/HS1_ON values available 1_B **VALID**, LS1/HS1_ON fields contain valid data
RES	29:23	r	**Reserved** Always read as 0
HB1_T3CMP_STS	22	r	**Half Bridge 1-Fast comparator status.** *Note: If HB1_SEQMAP = '0' the low-side (ground rail related) comparator is used to check if the low-side MOSFET is on. If HB1_SEQMAP = '1' the high-side (supply rail related) comparator is used to check if the high-side MOSFET is on.* 0_B **OFF**, selected MOSFET is off 1_B **ON**, selected MOSFET is on
HB1_T3ONCNT	21:16	r	**Half Bridge 1-Turn on slew rate-time value measured from beginning of phase 3 to end of phase 3.** 0_H **50ns**, 50ns phase duration $3E_H$ **3.15us**, 3.15us phase duration $3F_H$ **ERROR**, T3 value not valid - Measurement Error
RES	15:14	r	**Reserved** Always read as 0

Field	Bits	Type	Description
HB1_I1ONVAL	13:8	r	**Half Bridge 1-slew rate sequencer on-phase 1 current setting** *Note: the current used I1ON value for the adaptive mode is cleared by the bit HB1_T12ONCNTVF.* 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}
HB1_T12ONCNT	7:0	r	**Half Bridge 1-Turn on slew rate-time value measured from beginning of phase 1 to end of phase 2.** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration

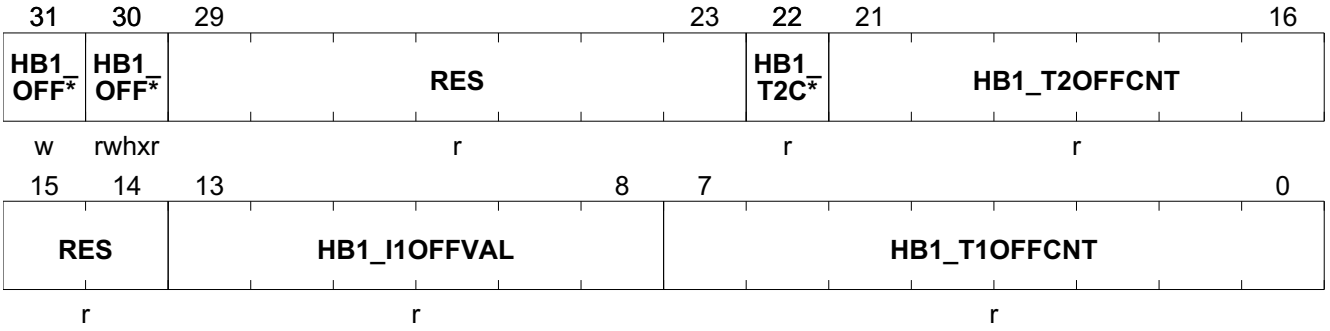
Table 566 RESET of BDRV_HB1ASEQONVAL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Bridge Driver (incl. Charge Pump)

Slewrate Adaptive Sequencers Off Values for Half Bridge 1

BDRV_HB1ASEQOFFVAL	Offset	Reset Value
Half Bridge 1 Adaptive Sequencer Off Values	BC_H	see Table 567



Field	Bits	Type	Description
HB1_OFFVALVF_CLR	31	w	**Half Bridge 1-Turn off slew rate values Valid Flag - Clear.** 0_B **NOT CLEAR**, no clear of HB1_OFFVALVF flag 1_B **CLEAR**, clear of HB1_OFFVALVF flag
HB1_OFFVALVF	30	rwhxr	**Half Bridge 1-Turn off slew rate values - Valid Flag.** 0_B **NOT VALID**, no new valid LS1/HS1_OFF values available 1_B **VALID**, LS1/HS1_OFF fields contain valid data
RES	29:23	r	**Reserved** Always read as 0
HB1_T2CMP_STS	22	r	**Half Bridge 1-Fast comparator status.** *Note: If HB1_SEQMAP = '0' the high-side (supply rail related) comparator is used to check if the low-side MOSFET is off. If HB1_SEQMAP = '1' the low-side (ground rail related) comparator is used to check if the high-side MOSFET is off.* 0_B **OFF**, selected MOSFET is off 1_B **ON**, selected MOSFET is on
HB1_T2OFFCNT	21:16	r	**Half Bridge 1-Turn off slew rate-time value measured from beginning of phase 2 to end of phase 2.** 0_H **50ns**, 50ns phase duration $3E_H$ **3.15us**, 3.15us phase duration $3F_H$ **ERROR**, T2 value not valid - Measurement Error
RES	15:14	r	**Reserved** Always read as 0

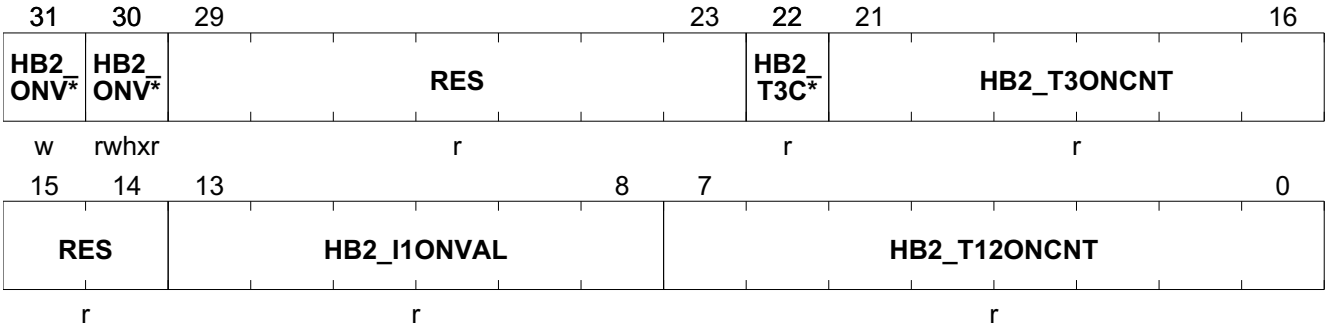
Field	Bits	Type	Description
HB1_I1OFFVAL	13:8	r	**Half Bridge 1-slew rate sequencer off-phase 1 current setting** *Note: the current used I1OFF value for the adaptive mode is cleared by the bit LS1_T1OFFCNTVF.* 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}
HB1_T1OFFCNT	7:0	r	**Half Bridge 1-Turn off slew rate-time value measured from beginning of phase 1 to end of phase 1.** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration

Table 567 RESET of BDRV_HB1ASEQOFFVAL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Slewrate Adaptive Sequencers On Values for Half Bridge 2

BDRV_HB2ASEQONVAL	Offset	Reset Value
Half Bridge 2 Adaptive Sequencer On Values	$D0_H$	see Table 568



Field	Bits	Type	Description
HB2_ONVALVF_CLR	31	w	**Half Bridge 2-Turn on slew rate values Valid Flag - Clear.** 0_B **NOT CLEAR**, no clear of HB2_ONVALVF flag 1_B **CLEAR**, clear of HB2_ONVALVF flag
HB2_ONVALVF	30	rwhxr	**Half Bridge 2-Turn on slew rate values - Valid Flag.** 0_B **NOT VALID**, no new valid LS2/HS2_ON values available 1_B **VALID**, LS2/HS2_ON fields contain valid data
RES	29:23	r	**Reserved** Always read as 0
HB2_T3CMP_STS	22	r	**Half Bridge 2-Fast comparator status.** *Note: If HB2_SEQMAP = '0' the low-side (ground rail related) comparator is used to check if the low-side MOSFET is on. If HB2_SEQMAP = '1' the high-side (supply rail related) comparator is used to check if the high-side MOSFET is on.* 0_B **OFF**, selected MOSFET is off 1_B **ON**, selected MOSFET is on
HB2_T3ONCNT	21:16	r	**Half Bridge 2-Turn on slew rate-time value measured from beginning of phase 3 to end of phase 3.** 0_H **50ns**, 50ns phase duration $3E_H$ **3.15us**, 3.15us phase duration $3F_H$ **ERROR**, T3 value not valid - Measurement Error
RES	15:14	r	**Reserved** Always read as 0

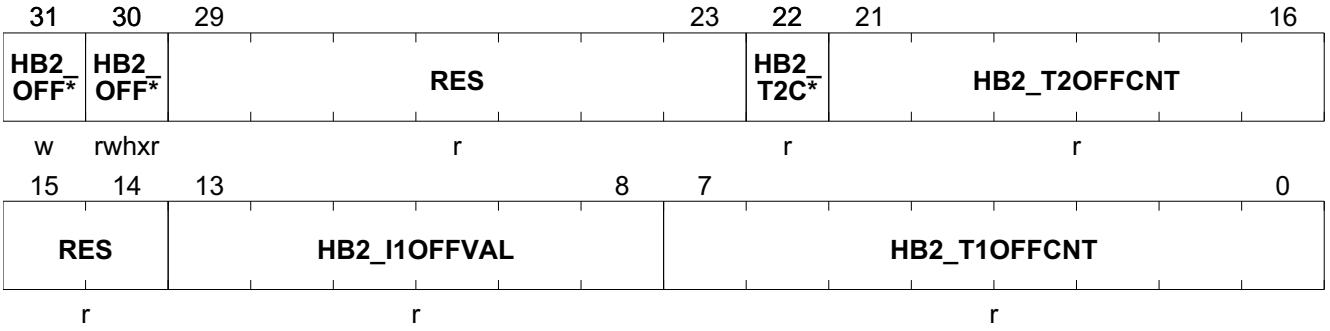
Field	Bits	Type	Description
HB2_I1ONVAL	13:8	r	**Half Bridge 2-slew rate sequencer on-phase 1 current setting** *Note: the current used I1ON value for the adaptive mode is cleared by the bit HB2_T12ONCNTVF.* 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}
HB2_T12ONCNT	7:0	r	**Half Bridge 2-Turn on slew rate-time value measured from beginning of phase 1 to end of phase 2.** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration

Table 568 RESET of BDRV_HB2ASEQONVAL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Slewrate Adaptive Sequencers Off Values for Half Bridge 2

BDRV_HB2ASEQOFFVAL	Offset	Reset Value
Half Bridge 2 Adaptive Sequencer Off Values	$D4_H$	see Table 569



Field	Bits	Type	Description
HB2_OFFVALVF_CLR	31	w	**Half Bridge 2-Turn off slew rate values Valid Flag - Clear.** 0_B **NOT CLEAR**, no clear of HB2_OFFVALVF flag 1_B **CLEAR**, clear of HB2_OFFVALVF flag
HB2_OFFVALVF	30	rwhxr	**Half Bridge 2-Turn off slew rate values - Valid Flag.** 0_B **NOT VALID**, no new valid LS2/HS2_OFF values available 1_B **VALID**, LS2/HS2_OFF fields contain valid data
RES	29:23	r	**Reserved** Always read as 0
HB2_T2CMP_STS	22	r	**Half Bridge 2-Fast comparator status.** *Note: If HB2_SEQMAP = '0' the high-side (supply rail related) comparator is used to check if the low-side MOSFET is off. If HB2_SEQMAP = '1' the low-side (ground rail related) comparator is used to check if the high-side MOSFET is off.* 0_B **OFF**, selected MOSFET is off 1_B **ON**, selected MOSFET is on
HB2_T2OFFCNT	21:16	r	**Half Bridge 2-Turn off slew rate-time value measured from beginning of phase 2 to end of phase 2.** 0_H **50ns**, 50ns phase duration $3E_H$ **3.15us**, 3.15us phase duration $3F_H$ **ERROR**, T2 value not valid - Measurement Error
RES	15:14	r	**Reserved** Always read as 0

Field	Bits	Type	Description
HB2_I1OFFVAL	13:8	r	**Half Bridge 2-slew rate sequencer off-phase 1 current setting** *Note: the current used I1OFF value for the adaptive mode is cleared by the bit HB2_T1OFFCNTVF.* 0_H **min. current**, I_{CHGmin} $3F_H$ **max. current**, I_{CHGmax}
HB2_T1OFFCNT	7:0	r	**Half Bridge 2-Turn off slew rate-time value measured from beginning of phase 1 to end of phase 1.** 0_H **50ns**, 50ns phase duration FF_H **12.8us**, 12.8us phase duration

Table 569 RESET of BDRV_HB2ASEQOFFVAL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

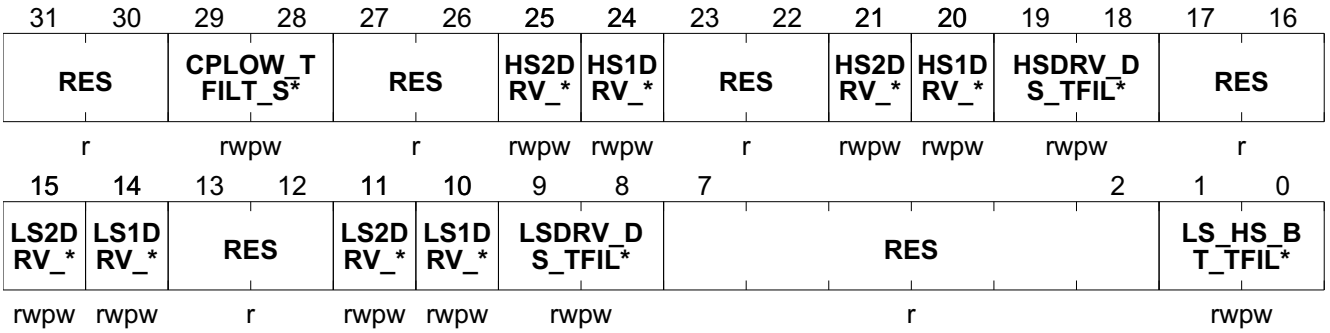
27.4.9 Driver Trimming Register

Note: time values are given for BRDRV_TFILT_CLK = 20 MHz

Trimming Driver

This register is password protected. Writing to it is only possible if password is set.

BDRV_TRIM_DRVx	Offset	Reset Value
Trimming of Driver	18_H	see Table 570



Field	Bits	Type	Description
RES	31:30	r	**Reserved** Always read as 0
CPLOW_TFILT_SEL	29:28	rwpw	**Filter Time for Charge Pump Voltage Low Diagnosis** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 00_B **4_us**, 4 µs filter time 01_B **8_us**, 8 µs filter time 10_B **16_us**, 16 µs filter time 11_B **32_us**, 32 µs filter time
RES	27:26	r	**Reserved** Always read as 0
HS2DRV_OCSDN_DIS	25	rwpw	**High Side 2 Predriver in overcurrent situation disable** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 0_B **Enable**, Predriver shutdown in overcurrent situation enable 1_B **Disable**, Predriver shutdown in overcurrent situation disable

Field	Bits	Type	Description
HS1DRV_OCSDN_DIS	24	rwpw	**High Side 1 Predriver in overcurrent situation disable** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 0_B **Enable**, Predriver shutdown in overcurrent situation enable 1_B **Disable**, Predriver shutdown in overcurrent situation disable
RES	23:22	r	**Reserved** Always read as 0
HS2DRV_FDISCHG_DIS	21	rwpw	**High Side 2 Predriver fast discharge disable** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 0_B **Enable**, Predriver shutdown fast discharge enable 1_B **Disable**, Predriver shutdown fast discharge disable
HS1DRV_FDISCHG_DIS	20	rwpw	**High Side 1 Predriver fast discharge disable** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 0_B **Enable**, Predriver shutdown fast discharge enable 1_B **Disable**, Predriver shutdown fast discharge disable
HSDRV_DS_TFILT_SEL	19:18	rwpw	**Filter Time for Drain-Source Monitoring of High Side Drivers** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 00_B **1_us**, 1 µs filter time 01_B **2_us**, 2 µs filter time 10_B **4_us**, 4 µs filter time 11_B **8_us**, 8 µs filter time
RES	17:16	r	**Reserved** Always read as 0

Field	Bits	Type	Description
LS2DRV_OCSDN_DIS	15	rwpw	**Low Side 2 Predriver in overcurrent situation disable** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 0_B **Enable**, Predriver shutdown in overcurrent situation enable 1_B **Disable**, Predriver shutdown in overcurrent situation disable
LS1DRV_OCSDN_DIS	14	rwpw	**Low Side 1 Predriver in overcurrent situation disable** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 0_B **Enable**, Predriver shutdown in overcurrent situation enable 1_B **Disable**, Predriver shutdown in overcurrent situation disable
RES	13:12	r	**Reserved** Always read as 0
LS2DRV_FDISCHG_DIS	11	rwpw	**Low Side 2 Predriver fast discharge disable** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 0_B **Enable**, Predriver shutdown fast discharge enable 1_B **Disable**, Predriver shutdown fast discharge disable
LS1DRV_FDISCHG_DIS	10	rwpw	**Low Side 1 Predriver fast discharge disable** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 0_B **Enable**, Predriver shutdown fast discharge enable 1_B **Disable**, Predriver shutdown fast discharge disable

Field	Bits	Type	Description
LSDRV_DS_TFILT_SEL	9:8	rwpw	**Filter Time for Drain-Source Monitoring of Low Side Drivers** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 00_B **1_us**, 1 µs filter time 01_B **2_us**, 2 µs filter time 10_B **4_us**, 4 µs filter time 11_B **8_us**, 8 µs filter time
RES	7:2	r	**Reserved** Always read as 0
LS_HS_BT_TFILT_SEL	1:0	rwpw	**Blanking Time for Drain-Source Monitoring of Low / High Side Drivers** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 00_B **1_us**, 1 µs filter time 01_B **2_us**, 2 µs filter time 10_B **4_us**, 4 µs filter time 11_B **8_us**, 8 µs filter time

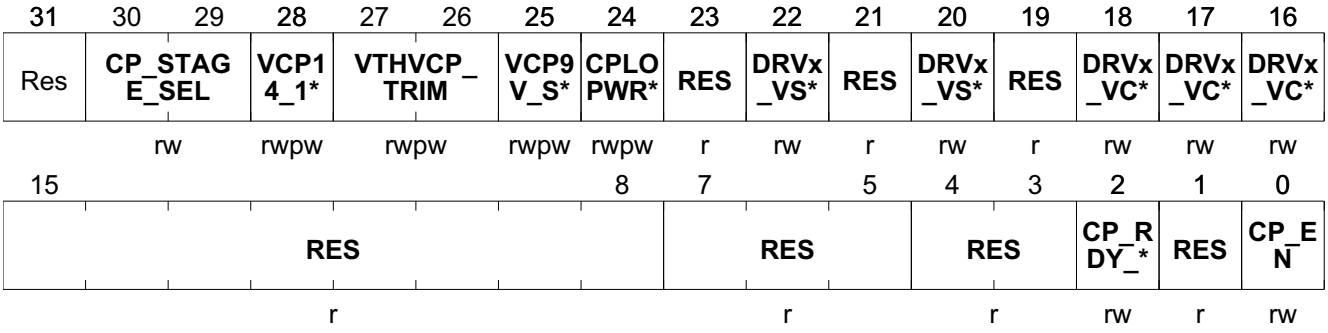
Table 570 RESET of BDRV_TRIM_DRVx

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00000000_H	RESET_TYPE_4		
TRIM_1	00000000_H	RESET		

27.4.10 Charge Pump Control and Status Register

Charge Pump Control and Status Register

BDRV_CP_CTRL	Offset	Reset Value
Charge Pump Control and Status Register	**20_H**	**see Table 571**



Field	Bits	Type	Description
CP_STAGE_SEL	30:29	rw	**Charge Pump Stage Selection** 00_B **2-stage**, 2-stage mode enabled 01_B **1-stage1**, single stage mode enable (1st stage) 10_B **1-stage2**, single stage mode enable (2nd stage) 11_B **auto**, automatic switch to single stage mode above 18V VSD (2nd stage) and switching back to 2-stage mode below 17V VSD
VCP14_15V_SEL	28	rwpw	**Charge Pump 15V/14V Output Voltage Sel** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 0_B **14V**, output voltage set to 14V 1_B **15V**, output voltage set to 15V
VTHVCP_TRIM	27:26	rwpw	**Charge Pump Output Voltage Trimming** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 00_B **0.0_V**, default 01_B **0.5_V**, plus 0.5V nom at 15V 10_B **1.0_V**, plus 1.0V nom at 15V 11_B **1.5_V**, plus 1.5V nom at 15V
VCP9V_SET	25	rwpw	**Charge Pump 9 V Output Voltage Set** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* *3. 9V mode is only enabled if CPLOPWRM_EN is also set!* 0_B **15_14V Set**, output voltage set according to VCP14_15V_SEL 1_B **9V Set**, output voltage set to 9V

Field	Bits	Type	Description
CPLOPWRM_EN	24	rwpw	**Charge Pump Low Power Mode Enable** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 0_B **Low Power Mode Disable**, low power mode inactive 1_B **Low Power Mode Enable**, low power mode active
RES	23	r	**Reserved** Always read as 0
DRVx_VSDUP_DIS	22	rw	**Driver shutdown in case of VSD overvoltage** 0_B **Enable**, DRVx shutdown in case of VSD overvoltage 1_B **Disable**, DRVx not shut down
RES	21	r	**Reserved** Always read as 0
DRVx_VSDLO_DIS	20	rw	**Driver shutdown in case of VSD undervoltage** 0_B **Enable**, DRVx shutdown in case of VSD undervoltage 1_B **Disable**, DRVx not shut down
RES	19	r	**Reserved** Always read as 0
DRVx_VCPUP_DIS	18	rw	**Driver shutdown in case of VCP overvoltage** 0_B **Enable**, DRVx shutdown in case of VCP overvoltage 1_B **Disable**, DRVx not shut down
DRVx_VCPLO_SDEN	17	rw	**Driver Charge Pump Low Voltage Shut-Down** 0_B **Shut-Down Disable**, DRVx Shut-Down for Charge Pump undervoltage disable. 1_B **Shut-Down Enable**, DRVx Shut-Down for Charge Pump undervoltage enable.
DRVx_VCPLO_DIS	16	rw	**Driver shutdown in case of VCP undervoltage** 0_B **Enable**, DRVx shutdown in case of VCP undervoltage 1_B **Disable**, DRVx not shut down
RES	15:8	r	**Reserved** Always read as 0
RES	7:5	r	**Reserved** Always read as 0
RES	4:3	r	**Reserved** Always read as 0
CP_RDY_EN	2	rw	**Bridge Driver on Charge Pump Ready Enable** 0_B **OFF**, Bridge Driver can be immediately enabled 1_B **ON**, Bridge Driver can only be enabled when Charge Pump is ready
RES	1	r	**Reserved** Always read as 0

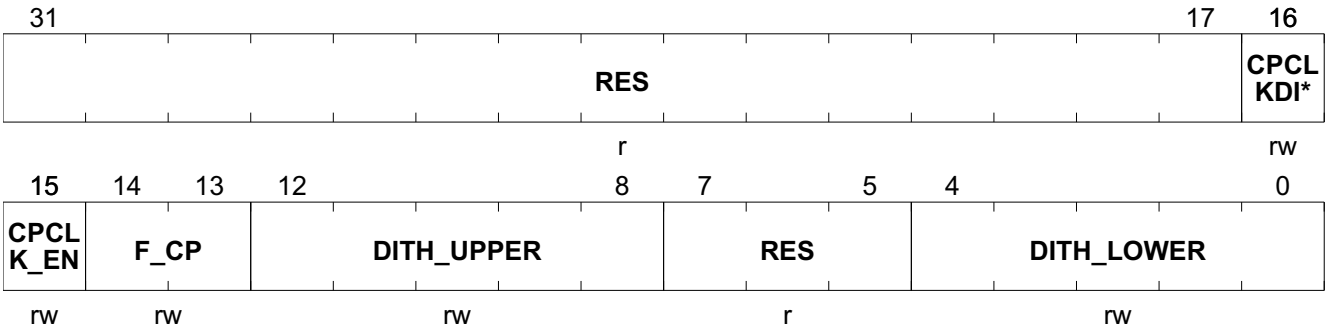
Field	Bits	Type	Description
CP_EN	0	rw	**Charge Pump Enable** 0_B **DISABLE**, Charge Pump, circuit power off 1_B **ENABLE**, Charge Pump, circuit power on

Table 571 RESET of BDRV_CP_CTRL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	10020000_H	RESET_TYPE_4		
TRIM_1	10020000_H	TRIM		

Charge Pump Clock Control Register

BDRV_CP_CLK_CTRL	Offset	Reset Value
Charge Pump Clock Control Register	24_H	see Table 572



Field	Bits	Type	Description
RES	31:17	r	**Reserved** Always read as 0
CPCLKDIS_SET	16	rw	**Charge Pump Clock Set If Disabled** 0_B **LOW**, Charge Pump Clock is 0 if disabled 1_B **HIGH**, Charge Pump Clock is 1 if disabled
CPCLK_EN	15	rw	**Charge Pump Clock Enable** 0_B **DISABLE**, Charge Pump Clock is switched off and has value of according to CPCLKDIS_SET 1_B **ENABLE**, Charge Pump Clock is running
F_CP	14:13	rw	**MSB of CP_CLK divider** CP_CLK frequency is defined by the concatenation of f_cp and dith_upper for the upper boundary during dithering, and dith_lower for the lower boundary
DITH_UPPER	12:8	rw	**CP_CLK upper frequency boundary during dithering** legal values are equal or less than DITH_LOWER, see definition of f_cp
RES	7:5	r	**Reserved** Always read as 0

Field	Bits	Type	Description
DITH_LOWER	4:0	rw	**CP_CLK lower frequency boundary during dithering** legal values are equal or greater than DITH_UPPER,see definition of f_cp

Table 572 RESET of BDRV_CP_CLK_CTRL

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	$0000CA16_H$	RESET_TYPE_3		

Charge Pump Status Register

BDRV_CP_IRQS	Offset	Reset Value
Charge Pump Status Register	40_H	see Table 573

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
VSD_UPT*	RES	VSD_LOT*	RES	VCP_UPT*	RES	VCP_LOT*	RES	RES			VCP_OTS*	RES			VCP_OTW*
rwhxr	r	rwhxr	r	rwhxr	r	rwhxr	r	r			rwhxr	r			rwhxr

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VSD_UPT*	RES	VSD_LOT*	RES	VCP_UPT*	RES	VCP_LOT*	RES	RES			VCP_OTS*	RES			VCP_OTW*
rwhxre	r	rwhxre	r	rwhxre	r	rwhxre	r	r			rwhxre	r			rwhxre

Field	Bits	Type	Description
VSD_UPTH_STS	31	rwhxr	**Driver Supply MU High Status** 0_B **Driver Supply Voltage ok**, no overvoltage detected 1_B **Driver Supply Voltage too high**, overvoltage on VSD Pin detected
RES	30	r	**Reserved** Always read as 0
VSD_LOTH_STS	29	rwhxr	**Driver Supply MU Low Status** 0_B **Driver Supply Voltage ok**, no undervoltage detected. 1_B **Driver Supply Voltage too low**, undervoltage on VSD Pin detected.
RES	28	r	**Reserved** Always read as 0
VCP_UPTH_STS	27	rwhxr	**Charge Pump MU High Status** 0_B **Charge Pump Output Voltage ok**, no overvoltage detected 1_B **Charge Pump Output Voltage too high**, overvoltage on charge pump output detected
RES	26	r	**Reserved** Always read as 0

Field	Bits	Type	Description
VCP_LOTH1_STS	25	rwhxr	**Charge Pump MU Low Status** 0_B **Charge Pump Output Voltage ok**, no undervoltage detected. 1_B **Charge Pump Output Voltage too low**, undervoltage on charge pump output detected.
RES	24	r	**Reserved** Always read as 0
RES	23:21	r	**Reserved** Always read as 0
VCP_OTSD_STS	20	rwhxr	**Charge Pump Overtemperature Shutdown Status** 0_B **Charge Pump Overtemperature Shutdown Threshold not reached**, no charge pump overtemperature shutdown detected. 1_B **Charge Pump Overtemperature Shutdown**, overtemperature shutdown on charge pump occured.
RES	19:17	r	**Reserved** Always read as 0
VCP_OTW_STS	16	rwhxr	**Charge Pump Overtemperature Warning Status** 0_B **Charge Pump Temperature ok**, no charge pump overtemperature warning detected. 1_B **Charge Pump Overtemperature Warning**, overtemperature threshold on charge pump reached.
VSD_UPTH_IS	15	rwhxre	**Driver Supply MU High Interrupt Status** 0_B **Driver Supply Voltage ok**, no overvoltage detected 1_B **Driver Supply Voltage too high**, overvoltage on VSD Pin detected
RES	14	r	**Reserved** Always read as 0
VSD_LOTH_IS	13	rwhxre	**Driver Supply MU Low Interrupt Status** 0_B **Driver Supply Voltage ok**, no undervoltage detected. 1_B **Driver Supply Voltage too low**, undervoltage on VSD Pin detected.
RES	12	r	**Reserved** Always read as 0
VCP_UPTH_IS	11	rwhxre	**Charge Pump MU High Interrupt Status** 0_B **Charge Pump Output Voltage ok**, no overvoltage detected 1_B **Charge Pump Output Voltage too high**, overvoltage on charge pump output detected
RES	10	r	**Reserved** Always read as 0

Field	Bits	Type	Description
VCP_LOTH1_IS	9	rwhxre	**Charge Pump MU Low Interrupt Status** 0_B **Charge Pump Output Voltage ok**, no undervoltage detected. 1_B **Charge Pump Output Voltage too low**, undervoltage on charge pump output detected.
RES	8	r	**Reserved** Always read as 0
RES	7:5	r	**Reserved** Always read as 0
VCP_OTSD_IS	4	rwhxre	**Charge Pump Overtemperature Shutdown Interrupt Status** 0_B **Charge Pump Overtemperature Shutdown Threshold not reached**, no charge pump overtemperature shutdown detected. 1_B **Charge Pump Overtemperature Shutdown**, overtemperature shutdown on charge pump occured.
RES	3:1	r	**Reserved** Always read as 0
VCP_OTW_IS	0	rwhxre	**Charge Pump Overtemperature Warning Interrupt Status** 0_B **Charge Pump Temperature ok**, no charge pump overtemperature warning detected. 1_B **Charge Pump Overtemperature Warning**, overtemperature threshold on charge pump reached.

Table 573 RESET of BDRV_CP_IRQS

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Charge Pump Interrupt Status Clear Register

BDRV_CP_IRQCLR	Offset	Reset Value
Charge Pump Interrupt Status Clear Register	44_H	see Table 574

31	30	29	28	27	26	25	24	23		21	20	19		17	16
VSD_UPT*	RES	VSD_LOT*	RES	VCP_UPT*	RES	VCP_LOT*	RES	RES			VCP_OTS*	RES			VCP_OTW*
w	r	w	r	w	r	w	r	r			w	r			w
15	14	13	12	11	10	9	8	7		5	4	3		1	0
VSD_UPT*	RES	VSD_LOT*	RES	VCP_UPT*	RES	VCP_LOT*	RES	RES			VCP_OTS*	RES			VCP_OTW*
w	r	w	r	w	r	w	r	r			w	r			w

Field	Bits	Type	Description
VSD_UPTH_SC	31	w	**Driver Supply MU High Status Clear** 0_B **no Clear**, 1_B **Clear**,
RES	30	r	**Reserved** Always read as 0
VSD_LOTH_SC	29	w	**Driver Supply MU Low Status Clear** 0_B **no Clear**, 1_B **Clear**,
RES	28	r	**Reserved** Always read as 0
VCP_UPTH_SC	27	w	**Charge Pump MU High Status Clear** 0_B **no Clear**, 1_B **Clear**,
RES	26	r	**Reserved** Always read as 0
VCP_LOTH1_SC	25	w	**Charge Pump MU Low Status Clear** 0_B **no Clear**, 1_B **Clear**,
RES	24	r	**Reserved** Always read as 0
RES	23:21	r	**Reserved** Always read as 0
VCP_OTSD_SC	20	w	**Charge Pump Over-temperature Shutdown Status Clear** 0_B **no Clear**, 1_B **Clear**,
RES	19:17	r	**Reserved** Always read as 0
VCP_OTW_SC	16	w	**Charge Pump Over-temperature Warning Status Clear** 0_B **no Clear**, 1_B **Clear**,
VSD_UPTH_ISC	15	w	**Driver Supply MU High Interrupt Status Clear** 0_B **no Clear**, 1_B **Clear**,

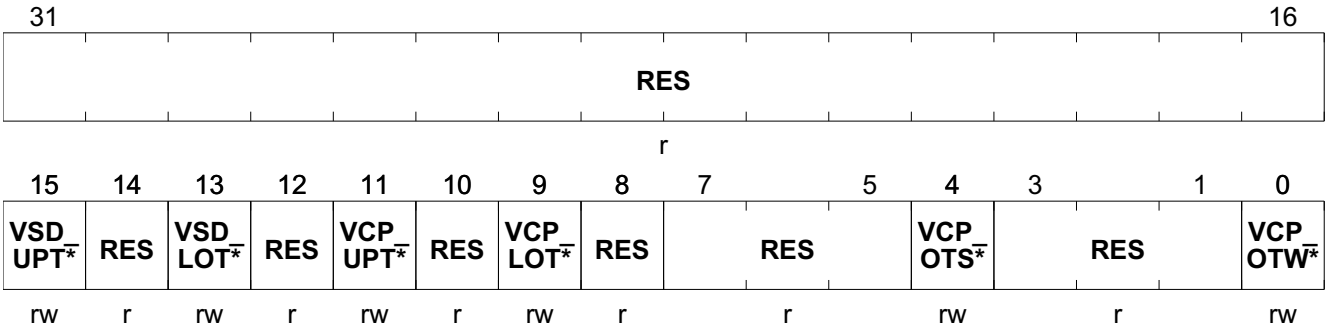
Field	Bits	Type	Description
RES	14	r	**Reserved** Always read as 0
VSD_LOTH_ISC	13	w	**Driver Supply MU Low Interrupt Status Clear** 0_B **no Clear**, 1_B **Clear**,
RES	12	r	**Reserved** Always read as 0
VCP_UPTH_ISC	11	w	**Charge Pump MU High Interrupt Status Clear** 0_B **no Clear**, 1_B **Clear**,
RES	10	r	**Reserved** Always read as 0
VCP_LOTH1_ISC	9	w	**Charge Pump MU Low Interrupt Status Clear** 0_B **no Clear**, 1_B **Clear**,
RES	8	r	**Reserved** Always read as 0
RES	7:5	r	**Reserved** Always read as 0
VCP_OTSD_ISC	4	w	**Charge Pump Over-temperature Shutdown Interrupt Status Clear** 0_B **no Clear**, 1_B **Clear**,
RES	3:1	r	**Reserved** Always read as 0
VCP_OTW_ISC	0	w	**Charge Pump Over-temperature Warning Interrupt Status Clear** 0_B **no Clear**, 1_B **Clear**,

Table 574 RESET of BDRV_CP_IRQCLR

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

Charge Pump Interrupt Enable Register

BDRV_CP_IRQEN	Offset	Reset Value
Charge Pump Interrupt Enable Register	48_H	see Table 575



Field	Bits	Type	Description
RES	31:16	r	**Reserved** Always read as 0
VSD_UPTH_IEN	15	rw	**Driver Supply MU High Interrupt Enable** 0_B **disable**, 1_B **enable**,
RES	14	r	**Reserved** Always read as 0
VSD_LOTH_IEN	13	rw	**Driver Supply MU Low Interrupt Enable** 0_B **disable**, 1_B **enable**,
RES	12	r	**Reserved** Always read as 0
VCP_UPTH_IEN	11	rw	**Charge Pump MU High Interrupt Enable** 0_B **disable**, 1_B **enable**,
RES	10	r	**Reserved** Always read as 0
VCP_LOTH1_IEN	9	rw	**Charge Pump MU Low Interrupt Enable** 0_B **disable**, 1_B **enable**,
RES	8	r	**Reserved** Always read as 0
RES	7:5	r	**Reserved** Always read as 0
VCP_OTSD_IEN	4	rw	**Charge Pump Over-temperature Shutdown Interrupt Enable** 0_B **disable**, 1_B **enable**,
RES	3:1	r	**Reserved** Always read as 0
VCP_OTW_IEN	0	rw	**Charge Pump Over-temperature Warning Interrupt Enable** 0_B **disable**, 1_B **enable**,

Table 575 RESET of BDRV_CP_IRQEN

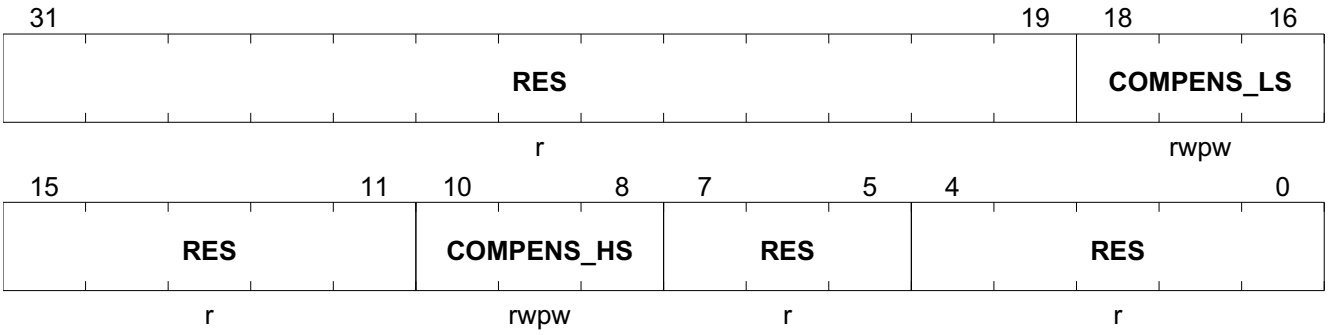
Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_3	00000000_H	RESET_TYPE_3		

27.4.11 Dynamic Compensation Trimming Register

Dynamic Compensation Trimming Register of Driver

This register is password protected. Writing to it is only possible if password is set.

BDRV_DCTRIM_DRVx	Offset	Reset Value
Dynamic Compensation Trimming of Driver	$E0_H$	see Table 576



Field	Bits	Type	Description
RES	31:19	r	**Reserved** Always read as 0
COMPENS_LS	18:16	rwpw	**Gain Settings for Low Side Charge Current Compensation** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 000_B **DISABLE**, Dynamic Compensation is disabled. 001_B **gain_1**, gain setting 1 (min) 010_B **gain_2**, gain setting 2 011_B **gain_3**, gain setting 3 100_B **gain_4**, gain setting 4 (max) 101_B **gain_4**, gain setting 4 (max) 110_B **gain_4**, gain setting 4 (max) 111_B **gain_4**, gain setting 4 (max)
RES	15:11	r	**Reserved** Always read as 0

Field	Bits	Type	Description
COMPENS_HS	10:8	rwpw	**Current Settings for High Side Charge Current Compensation** *Note: this SFR can only be written if the corresponding SCU_DM password register is written!* 000_B **DISABLE**, Dynamic Compensation is disabled. 001_B **gain_1**, gain setting 1 (min) 010_B **gain_2**, gain setting 2 011_B **gain_3**, gain setting 3 100_B **gain_4**, gain setting 4 (max) 101_B **gain_4**, gain setting 4 (max) 110_B **gain_4**, gain setting 4 (max) 111_B **gain_4**, gain setting 4 (max)
RES	7:5	r	**Reserved** Always read as 0
RES	4:0	r	**Reserved** Always read as 0

Table 576 RESET of BDRV_DCTRIM_DRVx

Register Reset Type	Reset Values	Reset Short Name	Reset Mode	Note
RESET_TYPE_4	00020200_H	RESET_TYPE_4		
TRIM_1	00020200_H	RESET		

28 Current Sense Amplifier

28.1 Features

Main Features

- Programmable gain settings: G = 10, 20, 40, 60
- Differential input voltage: ± 1.5V / G
- Wide common mode input range ± 2 V
- Low settling time < 1.4 µs

28.2 Introduction

The current sense amplifier in **Figure 232** can be used to measure near ground differential voltages via the 10-bit ADC. Its gain is digitally programmable through internal control registers.

Linear calibration has to be applied to achieve high gain accuracy, e.g. end-of-line calibration including the shunt resistor.

Figure 232 shows how the current sense amplifier can be used as a low-side current sense amplifier where the motor current is converted to a voltage by means of a shunt resistor R_{SH}. A differential amplifier input is used in order to eliminate measurement errors due to voltage drop across the stray resistance R_{Stray} and differences between the external and internal ground. If the voltage at one or both inputs is out of the operating range it has to be taken into account that the input circuit is overloaded and needs a certain specified **recovery time**.

In general, the external low pass filter should provide suppression of EMI.

28.2.1 Block Diagram

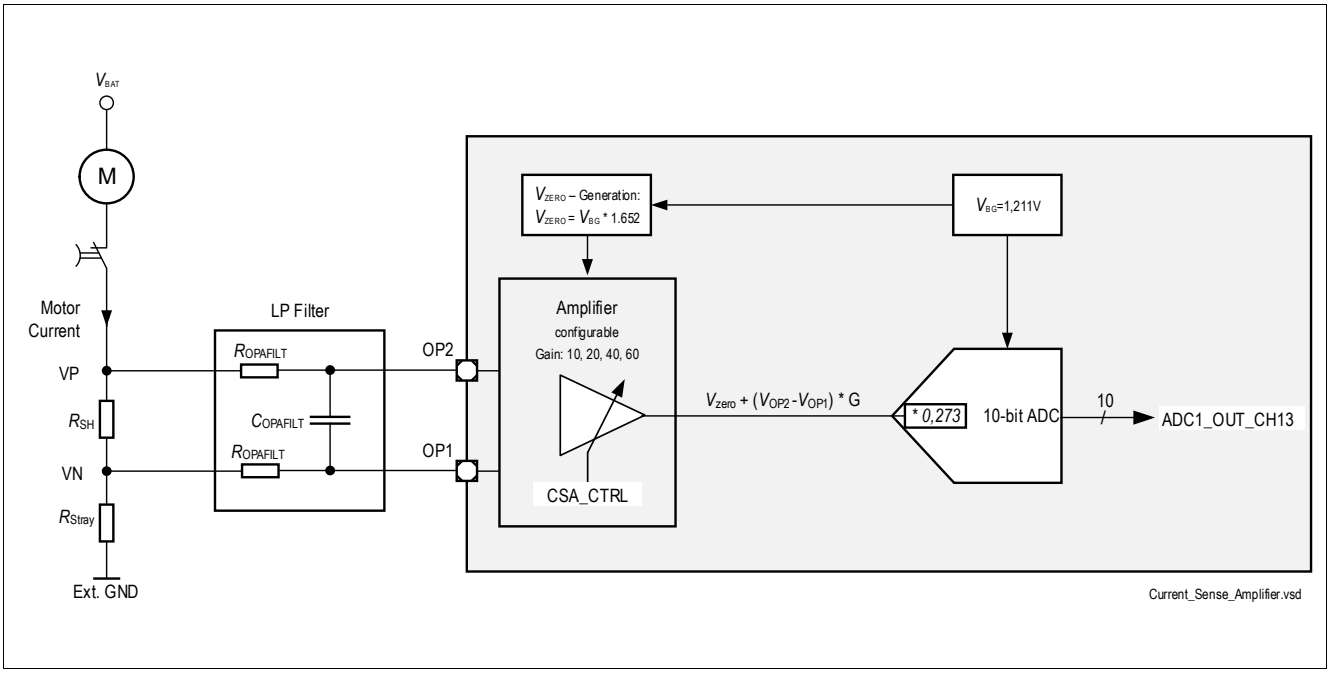


Figure 232 Simplified Application Diagram

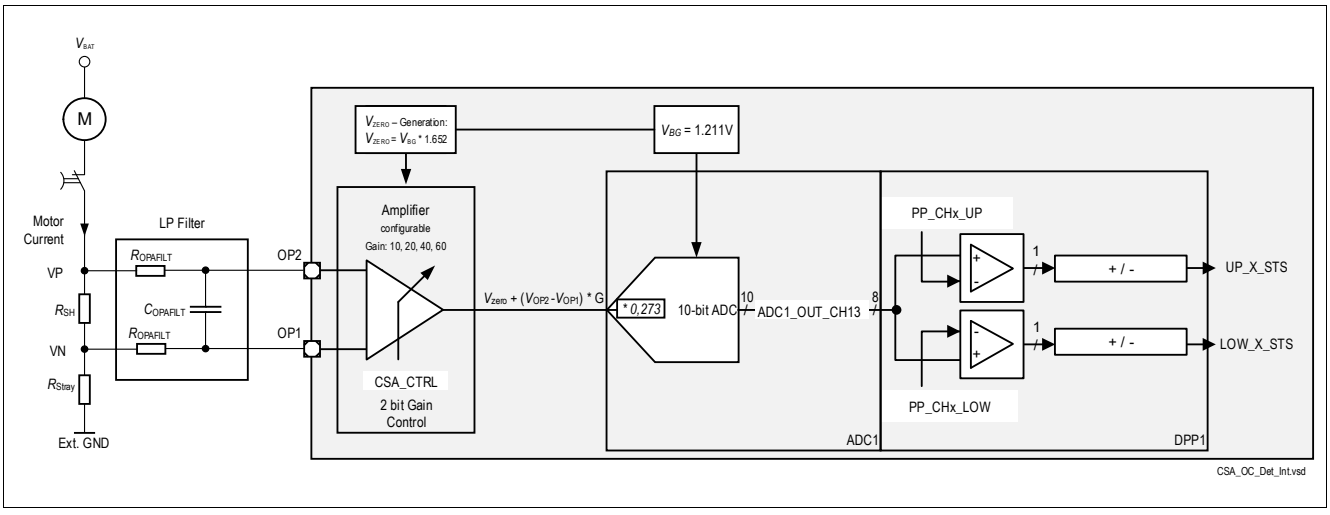


Figure 233 Simplified Application Diagram for Softshort Detection

28.3 Functional Description

28.3.1 ADC Code Calculation

The differential input voltage V_{OP2} - V_{OP1} of the embedded Current Sense Amplifier (CSA) is converted to an ADC code by the following equation:

(28.1)

$$ADC1out = floor\left(\frac{V_{zero} + (V_{OP2} - V_{OP1}) * G}{V_{LSB}} + 1\right)$$

wherein the parameter V_{OP2} and V_{OP1} are the voltages at the inputs of the amplifier and G is the configured gain.

The CSA output voltage V_{CSAout} = (V_{OP2} - V_{OP1}) * G is centered around an offset voltage V_{zero} which has the following dependency on the reference voltage V_{BG} of ADC1:

(28.2)

$$V_{zero} = 1.652 * V_{BG}$$

The LSB voltage is defined as follows:

(28.3)

$$V_{LSB} = \frac{V_{BG}}{1024 * 0.273}$$

28.4 Register Definition

The next chapter lists the configuration possibilities of the Current Sense Amplifier (CSA) which can be used for external current sensing.

Table 577 Amplifier Module Base Address List

Module	Base Address
CSA	48018000 $_H$

The base address of the module is the same as for the measurement unit (MU) as the current sense amplifier is a sub-block of the MU.

Table 578 Register Overview

Register Short Name	Register Long Name	Offset Address	Reset Value
Register Definition			
CSA_CTRL	Operational Amplifier Control and Status	00_H	0000 0000_H

The registers are addressed wordwise.

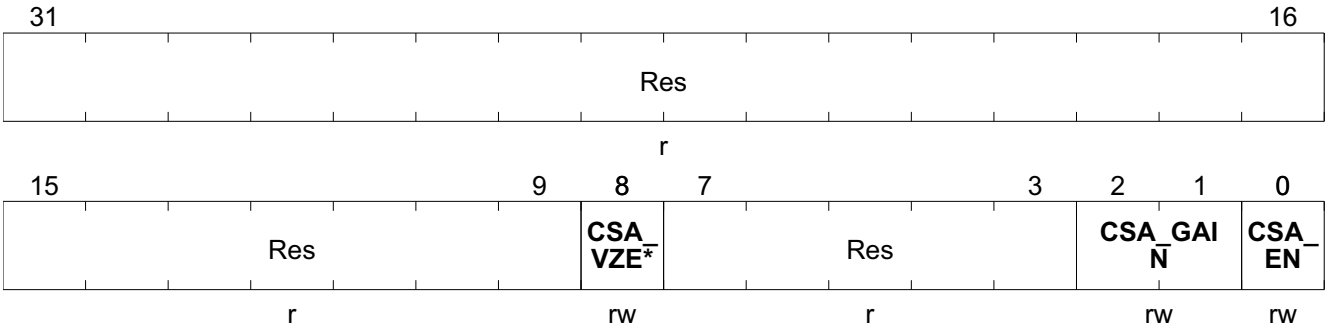
Operational Amplifier Control and Status

The following register consists of control and status bits. This Register is cleared by every reset.

The register is reset by RESET_TYPE_3.

The register is clocked by PCLK2 and reset by ap_all_reset_n_i.

CSA_CTRL	Offset	Reset Value
Operational Amplifier Control and Status	**00_H**	**0000 0000_H**



Field	Bits	Type	Description
Res	31:9	r	**Reserved** Always read as 0
CSA_VZERO	8	rw	**Current Sense Output Selection** 0_B **VOUT**, CSA output connected to ADC1 Ch13 1_B **VZERO**, voltage reference connected to ADC1 Ch13

Current Sense Amplifier

Field	Bits	Type	Description
Res	7:3	r	**Reserved** Always read as 0
CSA_GAIN	2:1	rw	**Operational Amplifier Gain Setting** 00_B **10**, Gain Factor 10 01_B **20**, Gain Factor 20 10_B **40**, Gain Factor 40 11_B **60**, Gain Factor 60
CSA_EN	0	rw	**CSA Enable** 0_B **DISABLE**, OPA switched off 1_B **ENABLE**, OPA switched on

29 Application Information

Note: The following information is given as a hint for the implementation of the device only and shall not be regarded as a description or warranty of a certain functionality, condition or quality of the device.

29.1 Window-Lift Application Diagram

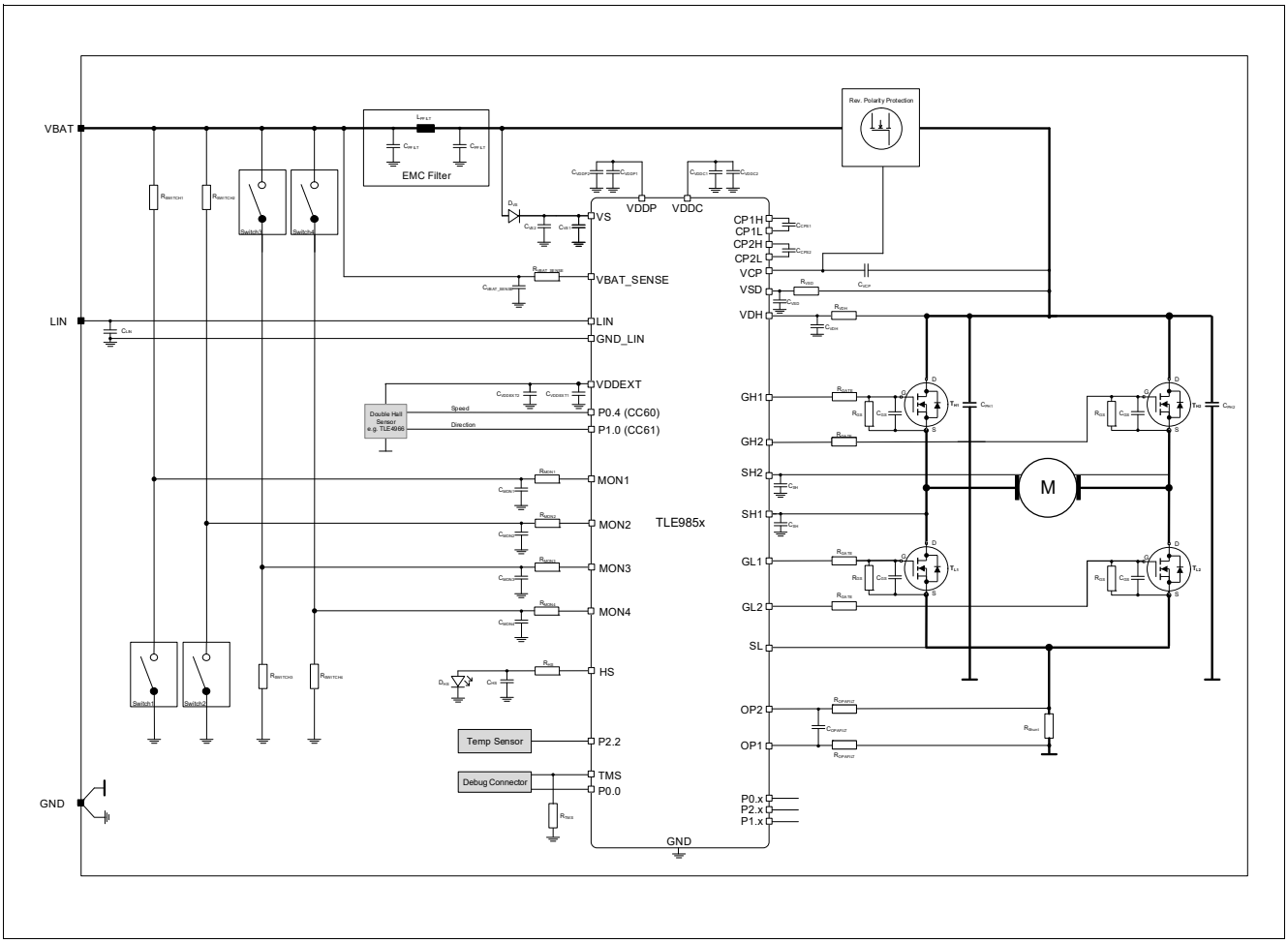


Figure 234 Simplified Application Diagram Example

Note: This is a very simplified example of an application circuit and bill of material. The function must be verified in the actual application.

Table 579 External Component (BOM)

Symbol	Function	Component
C_{VS1}	Capacitor 1 at VS pin	22 µF[1)]
C_{VS2}	Capacitor 2 at VS pin	100 nF[2)3)]
D_{VS}	Reverse-polarity protection diode	

Table 579 External Component (BOM) (cont'd)

Symbol	Function	Component
$C_{VDDEXT1}$	Capacitor 1 at VDDEXT pin	470 nF[3]
$C_{VDDEXT2}$	Capacitor 2 at VDDEXT pin	470 nF
C_{VDDC1}	Capacitor 1 at VDDC pin	100 nF[3]
C_{VDDC2}	Capacitor 2 at VDDC pin	330nF
C_{VDDP1}	Capacitor 1 at VDDP pin	470 nF[3]
C_{VDDP2}	Capacitor 2 at VDDP pin	470 nF
R_{MONx}	Resistor at MONx pin	1 kΩ
C_{MONx}	Capacitor at MONx pin	10 nF
R_{VBAT_SENSE}	Resistor at VBAT_SENSE pin	1 kΩ
C_{VBAT_SENSE}	Capacitor at VBAT_SENSE pin	10 nF
C_{LIN}	Capacitor at LIN pin	220 pF
R_{HS}	Resistor at HS pin	160 Ω[4]
C_{HS}	Capacitor at HS pin	6.8 nF or 33 nF (dependant on ESD GUN requirements)
D_{HS}	LED	
R_{VSD}	Limitation of reverse currents due to transients (-2V, 8ms)	2 Ω
C_{VSD}	Filter C for charge pump and driver	1 µF
C_{CPS1}	Charge pump flying capacitor 1	220 nF
C_{CPS2}	Charge pump flying capacitor 2	220 nF
C_{VCP}	Charge pump storage capacitor	470 nF
R_{VDH}	Resistor	1 kΩ
C_{VDH}	Capacitor	1 nF
C_{PH1}	DC-link buffer capacitor phase 1	220 µF
C_{PH2}	DC-link buffer capacitor phase 2	220 µF
R_{Shunt}	Shunt resistor	5 mΩ
$R_{OPAFILT}$	Resistor	12 Ω
$C_{OPAFILT}$	Capacitor	100 nF
C_{SH}	Capacitor	1 nF
R_{GATE}	Resistor	optional
R_{GS}	Resistor	100 kΩ
C_{GS}	Capacitor	4.7 nF (depends on MOSFET C_{GS})
L_{PFILT}	PI filter inductor	
C_{PFILT}	PI filter capacitor	10 µF
$R_{SWITCHx}$	Resistor	
R_{TMS}	Resistor	

1) to be dimensioned according to application requirements
2) to reduce the effect of fast voltage transients of Vs, these capacitors should be placed close to the device pin
3) ceramic capacitor

4) calculated for 24V (jump start)

29.2 Connection of unused pins

Table 580 shows recommendations how to connect pins, in case they are not needed by the application.

Table 580 Recommendation for connecting unused pins

type	pin number	recommendation 1 (if unused)	recommendation 2 (if unused)
LIN	48	open	
HS	2	VS	open
MON	17, 18, 19, 20	GND	open + configure internal PU/PD
GPIO	22, 24, 26, 27, 28, 29, 31, 32, 33, 34, 37, 38, 39, 40, 41	GND	external PU/PD or open + configure internal PU/PD
TMS	23	GND	
RESET	25	open	
P2/XTAL out	40	open	
P2/XTAL in	41	GND	
VDDEXT	45	open	
VBAT_SENSE	47	VS	

29.3 Connection of P0.2 for SWD debug mode

To enter the SWD debug mode, P0.2 needs to be 0 at the rising edge of the reset signal.

P0.2 has an internal pulldown, so it just needs to be ensured that there is no external 1 at P0.2 when the debug mode is entered.

29.4 Connection of TMS

For the debug mode, the TMS pin needs to be 1 at the rising edge of the reset signal. This is controlled by the debugger. The TMS pin has an internal PD.

To avoid the device entering the debug mode unintendedly in the final application, adding an external pull-down additionally is recommended.

29.5 ESD Tests

*Note: Tests for ESD robustness according to IEC61000-4-2 “gun test” (150pF, 330Ω) were performed. The results and test condition will be available in a test report. The target values for the test are listed in **Table 581** below.*

Table 581 ESD “Gun Test”

Performed Test	Result	Unit	Remarks
ESD at pin LIN, versus GND	>6	kV	1)positive pulse
ESD at pin LIN, versus GND	< -6	kV	1)negative pulse
ESD at pin VS, VBAT_SENSE, MONx, HS, versus GND	>6	kV	1)2)positive pulse
ESD at pin VS, VBAT_SENSE, MONx, HS, versus GND	< -6	kV	1)2)negative pulse

1) ESD susceptibility “ESD GUN”, tested by external test house (IBEE Zwickau, EMC Test report Nr. 07-01-19), according to “LIN Conformance Test Specification Package for LIN 2.1, October 10th, 2008” and “Hardware Requirements for LIN, CAN and FlexRay Interfaces in Automotive Application - AUDI, BMW, Daimler, Porsche, Volkswagen - Revision 1.3 / 2012”

2) With external circuit as shown in **Figure 234**.

30 Revision History

Revision	Date	Changes
1.0	2019-12-10	Initial version for AD-Step

Trademarks
All referenced product or service names and trademarks are the property of their respective owners.

Edition 2019-12-10
Published by
Infineon Technologies AG
81726 Munich, Germany

Do you have a question about any aspect of this document?
Email: erratum@infineon.com

Document reference
Z8F67526379

IMPORTANT NOTICE

The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics ("Beschaffenheitsgarantie").

With respect to any examples, hints or any typical values stated herein and/or any information regarding the application of the product, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

In addition, any information given in this document is subject to customer's compliance with its obligations stated in this document and any applicable legal requirements, norms and standards concerning customer's products and any use of the product of Infineon Technologies in customer's applications.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on technology, delivery terms and conditions and prices, please contact the nearest Infineon Technologies Office (**www.infineon.com**).

WARNINGS

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.