XMC5200 microcontroller

32-bit Arm® Cortex®-M4F

General description

XMC5200 is part of the XMC5000 microcontroller family designed for industrial systems. It features an Arm® Cortex®-M4F CPU for primary processing and an Arm® Cortex®-M0+ CPU for peripheral and security processing. These devices include embedded peripherals that support Controller Area Network with Flexible Data Rate (CAN FD).

XMC5000 series devices are manufactured using an advanced 40-nm process. XMC5200 integrates low-power flash memory and multiple high-performance analog and digital peripherals, enabling the creation of secure computing platforms.

Features

- **Dual CPU subsystem**
  - 160-MHz (max) 32-bit Arm® Cortex®-M4F CPU with
    - Single-cycle multiply
    - Single-precision floating-point unit (FPU)
    - Memory protection unit (MPU)
  - 100-MHz (max) 32-bit Arm® Cortex®-M0+ CPU with
    - Single-cycle multiply
    - Memory protection unit
    - Inter-processor communication in hardware
  - **Three DMA controllers**
    - Peripheral DMA controller #0 (P-DMA0) with 89 channels
    - Peripheral DMA controller #1 (P-DMA1) with 33 channels
    - Memory DMA controller #0 (M-DMA0) with 4 channels
- **Integrated memories**
  - 1088 KB of code-flash with an additional 96 KB of work-flash
    - Read-while-write (RWW) allows updating code flash/work flash during execution
    - Single- and dual-bank modes for firmware updates over the air (FOTA)
    - Flash programming through SWD/JTAG interface
  - 128 KB of SRAM with selectable retention granularity
- **Crypto engine**
  - Enhanced Secure Hardware Extension (eSHE)
  - Secure boot and authentication:
    - Digital signature verification
    - Fast secure boot
  - AES encryption: 128-bit blocks with 128-/192-/256-bit keys
  - True random number generator (TRNG) and pseudo-random number generator (PRNG)
  - Galois/counter mode (GCM)
- **Functional safety for application**
  - Memory protection unit (MPU)
  - Shared memory protection unit (SMPU)
  - Peripheral protection unit (PPU)
  - Watchdog timer (WDT)
  - Multi-counter watchdog timer (MCWDT)

Features

- Low-voltage detector (LVD)
- Brown-out detector (BOD)
- Overvoltage detection (OVD)
- Clock supervisor (CSV)
- Hardware error correction (single error correction and double error detection [SECDED ECC]) on all safety-critical memories (SRAM and Flash)

- **Low-power operation**
  - Operates from 2.7 V to 5.5 V
  - Multiple power modes for fine-grained power management: Active, Sleep, Low-power Sleep, DeepSleep, and Hibernate mode
  - Configurable brown-out detection (BOD) with:
    - Two threshold levels (2.7 V and 3.0 V) for BOD on V_{DDD} and VDDA
    - One threshold level (1.1 V) for BOD on VCCD
- **Wakeup support**
  - Up to two pins to wakeup from Hibernate mode
  - Up to 122 GPIO pins to wakeup from Sleep modes
  - Event generator, SCB, watchdog timer, RTC alarms to wake from DeepSleep modes
- **Clock sources**
  - Internal main oscillator (IMO)
  - Internal low-speed oscillator (ILO)
  - External crystal oscillator (ECO)
  - Watch crystal oscillator (WCO)
  - Phase-locked loop (PLL)
  - Frequency-locked loop (FLL)
- **Communication interfaces**
  - Up to six CAN FD channels
    - Increased data rate (up to 8 Mbps) compared to classic CAN, limited by physical layer topology and transceivers
    - ISO 11898-1:2015 compliant
    - ISO 16845:2015 certificate available
  - Up to eight runtime-reconfigurable serial communication block (SCB) channels, each configurable as I^2C, SPI, or UART
- **Timers**
  - Up to 75 16-bit and four 32-bit timer/counter pulse-width modulator (TCPWM) blocks:
    - Up to 12 16-bit counters for motor control
    - Up to 63 16-bit counters and four 32-bit counters for regular operations
    - Timer modes: Supports timer, capture, quadrature decoding, pulse-width modulation (PWM), PWM with dead time (PWM_DT), pseudo-random PWM (PWM_PR), and shift-register (SR) modes
  - Up to 11 Event Generation (EVTGEN) timers supporting cyclic wakeup from DeepSleep
    - Events trigger specific device operations such as interrupt handling or ADC conversions
- **Real time clock (RTC)**
  - Year/Month/Date, Day-of-week, Hour:Minute:Second fields
  - 12-hour and 24-hour formats
  - Automatic leap-year correction

Features

- **I/O**
  - Up to 122 programmable I/Os
  - Two types of I/Os:
    - GPIO Standard (GPIO_STD)
    - GPIO Enhanced (GPIO_ENH)
- **Regulators**
  - Generates 1.1 V nominal core supply from a 2.7 V to 5.5 V input supply
  - Two regulator types:
    - DeepSleep
    - Core internal
- **Programmable analog**
  - Three SAR analog-to-digital converters (ADC) converters with up to 57 external channels (54 I/Os + 3 I/Os for motor control)
    - ADC0 supports 24 logical channels, with 21 + 1 physical connections
    - ADC1 supports 32 logical channels, with 25 + 1 physical connections
    - ADC2 supports 8 logical channels, with 8 + 1 physical connections
    - Any external channel can be connected to any logical channel in the respective SAR
  - Internal analog inputs:
    - Bandgap reference for absolute voltage levels
    - Calibrated diode for junction temperature calculations
    - Two AMUXBUS inputs and two direct connections to monitor supply levels
  - Addressing of external multiplexers
  - Autonomous scanning of configured channels via ADC sequencers
  - Synchronized sampling for motor-sensing applications
- **Smart I/O**
  - Up to five Smart I/O blocks for Boolean operations on signals going to and from I/Os
  - Up to 29 I/Os (GPIO_STD) supported
- **Debug interface**
  - JTAG controller and interface compliant with IEEE-1149.1-2001
  - Arm® Serial Wire Debug (SWD) port
  - Arm® Embedded Trace Macrocell (ETM) supports:
    - Data trace using SWD
    - Instruction and data trace using JTAG
- **Packages**
  - 64-LQFP, 10 × 10 × 1.7 mm (max), 0.5-mm lead pitch
  - 100-LQFP, 14 × 14 × 1.7 mm (max), 0.5-mm lead pitch
  - 144-LQFP, 20 × 20 × 1.7 mm (max), 0.5-mm lead pitch

Table of contents

Table of contents

Table of contents

1 Features list

Table 1 XMC5200 feature list for all packages

Features	Package		
	64-LQFP	**100-LQFP**	**144-LQFP**
CPU			
Core	32-bit Arm® Cortex®-M4F CPU and 32-bit Arm® Cortex® M0+ CPU		
Operating voltage	2.7 V to 5.5 V		
Core voltage	1.05 V to 1.15 V		
Operating frequency	Arm® Cortex®-M4F 160 MHz (max) and Arm® Cortex®-M0+ 100 MHz (max), related by integer frequency ratio (that is, 1:1, 1:2, 1:3, and so on)		
MPU, PPU	Supported		
FPU	Single precision (32-bit)		
DSP-MUL/DIV/MAC	Supported by Arm® Cortex®-M4F CPU		
Memory			
Code flash	1088 KB (960 KB + 128 KB)		
Work flash	96 KB (72 KB + 24 KB)		
SRAM (configurable for retention)	128 KB		
ROM	32 KB		
Communication interfaces			
CAN0 (CAN FD: Up to 8 Mbps)	3 ch		
CAN1 (CAN FD: Up to 8 Mbps)	2 ch	3 ch	
CAN RAM	24 KB per instance (3 ch), 48 KB in total		
Serial communication block (SCB/UART)	7 ch	8 ch	
Serial communication block (SCB/I^2C)	6 ch	8 ch	
Serial communication block (SCB/SPI)	3 ch	8 ch	
Timers			
RTC	1 ch		
TCPWM (16-bit) (Motor control)	12 ch		
TCPWM (16-bit)	63 ch		
TCPWM (32-bit)	4 ch		
External interrupts	49	78	122
Analog			
12-bit, 1 Msps SAR ADC	3 Units (SAR0/24, SAR1/32, SAR2/8 logical channels)		

(table continues...)

1 Features list

Table 1 (continued) XMC5200 feature list for all packages

Features	Package		
	64-LQFP	**100-LQFP**	**144-LQFP**
	27 external channels (SAR0 11 ch, SAR1 9 ch, SAR2 7 ch)	39 external channels (SAR0 14 ch, SAR1 17 ch, SAR2 8 ch)	54 external channels (SAR0 21 ch, SAR1 25 ch, SAR2 8 ch)
	18 ch (6 per ADC) Internal sampling		
Motor control input	3 ch (synchronous sampling of one channel on each of the 3 ADCs)		
Security			
Flash security (program/work read protection)	Supported		
Flash chip erase enable	Configurable		
eSHE	By separate firmware[1)]		
System			
DMA controller	P-DMA0 with 89 channels (16 general purpose), P-DMA1 with 33 channels (8 general purpose), and M-DMA0 with 4 channels		
Internal main oscillator	8 MHz		
Internal low-speed oscillator	32.768 kHz (nominal)		
PLL	Input frequency: 3.988 to 33.34 MHz, PLL output frequency: up to 160 MHz		
FLL	Input frequency: 0.25 to 80 MHz, FLL output frequency: up to 100 MHz		
Watchdog timer and multi-counter watchdog timer	Supported (WDT + 2× MCWDT) MCWDT#0 tied to CM0+, MCWDT#1 to CM4		
Clock supervisor	Supported		
Cyclic wakeup from DeepSleep	Supported		
GPIO_STD	45	74	118
GPIO_ENH	4		
Smart I/O (Blocks)	3 blocks, 9 I/Os	4 blocks, 20 I/Os	5 blocks, 29 I/Os
Low-voltage detect	Two, 26 selectable levels		
Maximum ambient temperature	125°C		
Debug interface	SWD/JTAG		
Debug trace	Arm® Cortex®-M4F ETB size of 8 KB, Arm® Cortex® M0+ MTB size of 4 KB		

1 Enhanced secure hardware extension (eSHE) support is enabled by third-party firmware.

1.1 Communication peripheral instance list

The following table lists the supported instances of communication peripherals for each package, based on the minimum number of pins required for functionality.

Table 2 Peripheral instance list

Module	64-LQFP	100-LQFP	144-LQFP	Minimum pin functions
CAN0	0/1/2	0/1/2	0/1/2	TX, RX
CAN1	0/2	0/1/2	0/1/2	TX, RX
SCB/UART	0/1/2/3/4/5/7	0/1/2/3/4/5/6/7	0/1/2/3/4/5/6/7	TX, RX
SCB/I2C	0/2/3/4/5/7	0/1/2/3/4/5/6/7	0/1/2/3/4/5/6/7	SCL, SDA
SCB/SPI	0/3/4	0/1/2/3/4/5/6/7	0/1/2/3/4/5/6/7	MISO, MOSI, SCK, SELECT0

2 Blocks and functionality

2.1 Block diagram

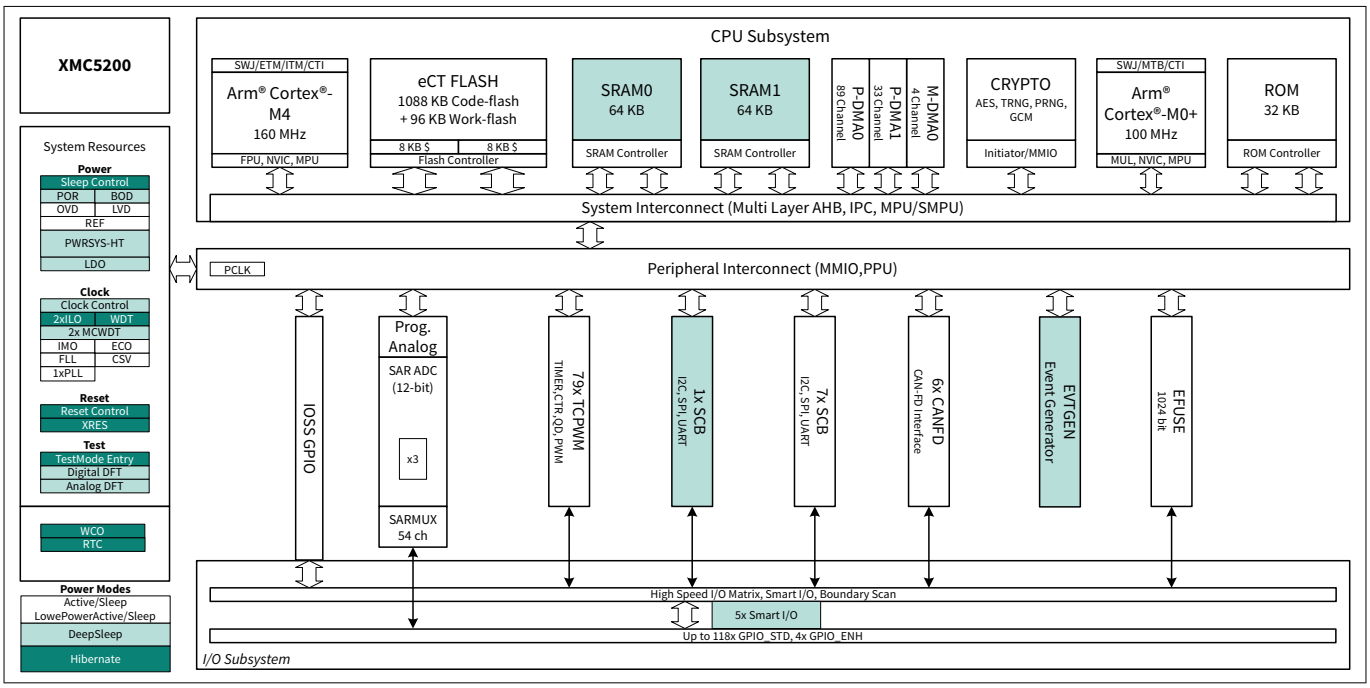


Figure 1 Block diagram

Figure 1 illustrates the XMC5200 architecture, providing a simplified view of the interconnections between subsystems and blocks. The architecture is divided into four major subsystems: CPU, system resources, peripherals, and I/O [2) 3)].

The color coding in the architecture diagram indicates the lowest power mode in which a particular block remains functional.

XMC5200 provides extensive support for programming, testing, debugging, and tracing of both hardware and firmware. The debug-on-chip functionality allows in-system debugging directly on the production device without requiring special interfaces, debugging pods, simulators, or emulators.

The JTAG interface is fully compatible with industry-standard third-party probes, such as I-jet and J-Link. Debug circuits are enabled by default to facilitate development and testing.

XMC5200 offers a high level of security by providing robust flash protection and options to disable specific features, such as debugging. Additionally, each device interface can be permanently disabled to mitigate security threats, including phishing attacks from maliciously reprogrammed devices or attempts to compromise security by interrupting critical flash programming sequences.

When maximum device security is enabled, all programming, debug, and test interfaces are disabled to ensure complete protection.

2 GPIO_STD supports a 2.7 V to 5.5 V V_{DDIO} range

3 GPIO_ENH supports the same 2.7 V to 5.5 V V_{DDIO} range but offers higher current capabilities at lower voltages

3 Functional description

3.1 CPU subsystem

3.1.1 CPU

The CPU subsystem consists of two 32-bit processors:

- An Arm® Cortex®-M0+ CPU with a memory protection unit (MPU)
- An Arm® Cortex®-M4F CPU with an MPU and a single-precision floating-point unit (FPU)

This subsystem also includes P-/M-DMA controllers, a cryptographic accelerator, 1088 KB of code flash, 96 KB of work flash, 128 KB of SRAM, and 32 KB of ROM.

The Cortex®-M0+ CPU provides a secure and uninterruptible boot process, ensuring that system integrity is validated and privileges are enforced upon boot completion. Access to shared resources, such as flash, SRAM, and peripherals, is managed through bus arbitration. Exclusive access to these resources is supported by an inter-processor communication (IPC) mechanism that uses hardware semaphores.

3.1.2 DMA controllers

XMC5200 has three DMA controllers: P-DMA0 with 16 general-purpose and 73 dedicated channels, P-DMA1 with 8 general-purpose and 25 dedicated channels, and M-DMA0 with four channels.

The peripheral DMA (P-DMA) controllers handle peripheral-to-memory and memory-to-peripheral data transfers. They provide low latency and support a large number of channels. Each P-DMA controller uses a single data-transfer engine shared among all its associated channels.

- **General-purpose channels**: Feature a rich interconnect matrix, including P-DMA cross-triggering, which supports complex and demanding data-transfer scenarios
- **Dedicated channels**: Have a single triggering input, such as an ADC channel, to manage common transfer requirements efficiently

The memory DMA (M-DMA) controller handles memory-to-memory data transfers, offering high memory bandwidth for a small number of channels. Unlike P-DMA, M-DMA employs a dedicated data-transfer engine for each channel, enabling independent access to peripherals through the AHB multi-layer bus.

3.1.3 Flash

XMC5200 has 1088 KB (960 KB with a 32-KB sector size, and 128 KB with an 8-KB sector size) of code flash with an additional work flash of up to 96 KB (72 KB with 2-KB sector size, and 24 KB with 128-KB sector size). Work flash is optimized for significantly higher reprogramming cycles compared to code flash. Code flash supports read-while-write (RWW) functionality, enabling flash updates while the CPU remains active.

Both code flash and work flash support dual-bank operation, which facilitates over-the-air (OTA) programming.

3.1.4 SRAM

XMC5200 has 128 KB of SRAM with two independent controllers. The first controller, SRAM0, provides DeepSleep retention in 32-KB increments, while SRAM1 offers a selectable retention mode, allowing it to be either fully retained or not retained during DeepSleep.

3.1.5 ROM

XMC5200 includes 32 KB of ROM, which contains boot and configuration routines. This ROM supports secure boot and authentication of user flash, ensuring a secure system.

3.1.6 Cryptography accelerator for security

The cryptographic accelerator supports the implementation of:

- AES block cipher
- Pseudo-random number generation (PRNG)
- True random number generation (TRNG)
- Galois/counter mode (GCM)

3.2 System resources

3.2.1 Power system

The power system ensures that supply voltage levels meet the requirements of each power mode and provides a full-system reset if these levels become invalid. An internal power-on reset (POR) guarantees a full-chip reset during the initial power ramp.

Brown-out detection

The system includes three BOD circuits to monitor external supply voltages (VDDD, VDDA, VCCD):

- The BOD circuits for VDDD and VCCD are enabled by default and cannot be disabled
- The BOD circuit for VDDA is initially disabled but can be enabled by the user

For external supplies (VDDD and VDDA), the BOD circuits are software-configurable with two voltage settings:

1. **2.7 V minimum**: Ensures robust internal signaling
2. **3.0 V minimum**: Ensures robust I/O operation, which is guaranteed at 2.7 V

The BOD on VCCD is provided as a safety measure but is not designed as a robust detector.

Overvoltage detection (OVD) and overcurrent detection (OCD)

The system includes:

- Three overvoltage detection (OVD) circuits to monitor external supplies (VDDD, VDDA, VCCD)
- Overcurrent detection (OCD) circuits for monitoring internal and external regulators

The OVD thresholds for VDDD and VDDA are software-configurable with two maximum voltage settings: 5.0 V and 5.5 V. Additionally, two voltage-detection circuits monitor the external supply voltage (VDDD) for falling and rising levels. These circuits are configurable for one of 26 selectable levels.

Reset and fault generation

All BOD, OVD, and OCD circuits on VDDD and VCCD generate a reset to protect the CPUs and fault logic. The BOD and OVD circuits for VDDA can be configured to generate either a reset or a fault, depending on system requirements.

3.2.2 Regulators

XMC5200 includes two regulators that provide power to the low-voltage core transistors: DeepSleep regulator and core internal regulator. These regulators accept a 2.7 V to 5.5 V VDDD supply and generate a low-noise 1.1 V supply for various parts of the device. The regulators are automatically enabled and disabled by hardware and firmware when transitioning between power modes. The core internal and core external regulators operate in active mode and supply power to the CPU subsystem and its associated peripherals.

DeepSleep

The DeepSleep regulator maintains power to a small number of blocks during DeepSleep mode. These blocks include:

- The ILO and WDT timers
- The BOD detector
- SCB0
- SRAM memories
- Smart I/O
- Other configuration memories

The DeepSleep regulator is enabled during DeepSleep mode, while the core internal regulator is disabled. It is disabled when XRES_L is asserted (LOW) and also when the core internal regulator is disabled.

Core internal

The core internal regulator supports load currents of up to 150 mA and operates during device startup (boot process) as well as in Active and Sleep modes.

3.2.3 Clock system

The XMC5200 clock system provides clocks to all subsystems that require them and supports glitch-free switching between different clock sources. It also ensures that no metastable conditions occur during operation.

The clock system consists of the following components:

- An 8-MHz internal main oscillator (IMO)
- Two internal low-speed oscillators (ILOs)
- Three watchdog timers
- A phase-locked loop (PLL)
- A frequency-locked loop (FLL)
- Five clock supervisors (CSVs)
- A 3.988- to 33.34-MHz external crystal oscillator (ECO)
- A 32.768-kHz watch crystal oscillator (WCO)

The clock system supports two main clock domains: CLK_HF and CLK_LF.

- **CLK_HF**:
  - Active domain clocks used during normal operation
  - Can source from any high-frequency clock, including IMO, EXT_CLK, ECO, FLL, or PLL
- **CLK_LF**:
  - DeepSleep domain clock that provides the clock source for modules such as the multi-counter watchdog timer (MCWDT) or the real-time clock (RTC)
  - The reference clock for the CLK_LF domain is selectable from ILO0, ILO1, or WCO, or it can be disabled

Table 3 CLK_HF destinations

Name	Description
CLK_HF0	CPUSS clocks, PERI, and AHB infrastructure
CLK_HF1	Event generator, also available in HSIOM as an output

3.2.3.1 IMO clock source

The internal main oscillator (IMO) serves as the frequency reference in XMC5200 when no external clock source is available or enabled. The IMO operates at a frequency of approximately 8 MHz.

3.2.3.2 ILO clock source

An ILO is a low-power oscillator, typically operating at 32.768 kHz, that generates clocks for a watchdog timer during DeepSleep mode. There are two ILOs to ensure clock supervisor (CSV) functionality in DeepSleep mode. ILO-driven counters can be calibrated to the IMO, WCO, or ECO to enhance their accuracy. ILO1 is also used for clock supervision.

3.2.3.3 PLL and FLL

A PLL or FLL can be used to generate high-speed clocks from the IMO, ECO, or EXT_CLK. The FLL locks significantly faster than the PLL (5 µs compared to 35 µs) but introduces a small frequency error of ±2%.[4)]

3.2.3.4 Clock supervisor

Each CSV allows one clock (reference) to supervise the behavior of another clock (monitored). Each CSV includes counters for both the monitored and reference clocks. Parameters for each counter determine the frequency of the reference clock and the upper and lower frequency limits of the monitored clock. If the frequency range comparator detects a stopped clock or a clock outside the specified frequency range, it signals an abnormal state and generates either a reset or an interrupt.

3.2.3.5 EXTCLK

One of the two GPIO_STD I/Os can provide an external clock input of up to 80 MHz. This clock can serve as the source for the PLL or FLL, or it can be used directly by the CLK_HF domain.

3.2.3.6 ECO

The ECO provides high-frequency clocking using an external crystal connected to the ECO_IN and ECO_OUT pins. It supports fundamental-mode quartz crystals (non-overtone) within a frequency range of 3.988 to 33.34 MHz. When used with the PLL, it generates CPU and peripheral clocks up to the device's maximum frequency. The accuracy of the ECO depends on the selected crystal. If the ECO is disabled, the associated pins are available for any of the supported I/O functions.

3.2.3.7 WCO

The WCO is a low-power watch crystal oscillator designed for real-time clock applications. It requires an external 32.768-kHz crystal connected to the WCO_IN and WCO_OUT pins. The WCO can also be configured as a clock reference for CLK_LF, which serves as the clock source for the MCWDT and RTC.

3.2.4 Reset

XMC5200 can be reset from various sources, including software. Reset events are asynchronous and ensure reversion to a known state. The reset cause such as POR, BOD, OVD, overcurrent, XRES_L, WDT, MCWDT,

[4] Using an FLL-based reference for operating reference-timed peripherals, such as a UART, is not recommended due to the allowed frequency error.

software reset, fault, CSV, Hibernate wakeup, or debug is recorded in a register. This register is sticky through reset, allowing software to identify the cause of the reset. An XRES_L pin is also available for external reset.

3.2.5 Watchdog timers

XMC5200 includes one watchdog timer (WDT) and two multi-counter watchdog timers (MCWDT).

- **Watchdog timer**:The WDT is a free-running counter clocked exclusively by ILO0, enabling it to function as a wakeup source from Hibernate. It operates across all power modes and must be serviced within a configured window. If it is not serviced before the timeout, a watchdog reset is triggered. This reset is recorded in the Reset Cause register
- **Multi-counter watchdog timer**: An MCWDT is available for each CPU core. These timers offer more capabilities than the WDT and are only functional in Active, Sleep, and DeepSleep modes. Each MCWDT includes multiple counters that can operate independently or in cascading mode to trigger interrupts and/or resets. They are clocked by either ILO0 or the WCO

3.2.6 Power modes

XMC5200 supports six different power modes:

- **Active**: All peripherals are available
- **Low-power active (LPACTIVE)**: A low-power variant of Active mode where all peripherals and CPUs remain available, but with reduced capability
- **Sleep**: All peripherals except the CPUs are available
- **Low-power sleep (LPSLEEP)**: A low-power variant of Sleep mode where all peripherals except the CPUs remain available, but with reduced capability
- **DeepSleep**: Only peripherals operating with CLK_LF are available
- **Hibernate**: The device and I/O states are frozen, and the device resets upon wakeup

3.3 Peripherals

3.3.1 Peripheral clock dividers

Integer and fractional clock dividers are available for peripheral and timing purposes.

Table 4 Clock dividers

Divider	Count	Description
div_8	32	Integer divider, 8 bits
div_16	16	Integer divider, 16 bits
div_24_5	8	Fractional divider, 24.5 bits (24 integer bits, 5 fractional bits)

3.3.2 Peripheral protection unit

The peripheral protection unit (PPU) controls and monitors unauthorized access from all masters, including the CPU, P-/M-DMA, Crypto module, and any enabled debug interface, to the peripherals. It regulates data transfers on the bus infrastructure by allowing or restricting access. Access rules are enforced based on specific transfer properties, such as the address range and access attributes (read/write, user/privilege, and secure/non-secure).

3.3.3 12-bit SAR ADC

XMC5200 contains three 1-Msps SAR ADCs. These ADCs are clocked at up to 26.67 MHz and provide a 12-bit result in 26 clock cycles.

All three SAR ADCs use a dedicated pair of reference inputs: VREFH and VREFL.[5)]

XMC5200 devices support up to 85 logical ADC channels, and external inputs from up to 57 I/Os. Each ADC also includes six internal connections for diagnostic and monitoring purposes. The number of ADC channels (per ADC and package type) are listed in Table 1.

Each ADC has an autonomous sequencer that cycles through configured channels (sequencer scan) with zero-switching overhead. This means that at a clock rate of 26.67 MHz, the aggregate sampling bandwidth is 1 Msps, regardless of whether it is for a single channel or distributed over multiple channels. The sequencer operates through a state machine or firmware and handles:

- Trigger request prioritization
- Enabling the appropriate analog channel
- Controlling ADC sampling
- Initiating ADC data conversion
- Managing results
- Performing repetitive or grouped conversions without CPU intervention

Each SAR ADC includes an analog multiplexer to connect signals for measurement. The multiplexer supports 54 GPIO_STD inputs, one special GPIO_STD input for motor-sense, and six additional inputs for measuring internal signals such as a band-gap reference, a temperature sensor, and power supplies. The device supports synchronous sampling of one motor-sense channel per ADC, for a total of three motor-sense channels across the ADCs.

XMC5200 has a single temperature sensor that is shared among all three ADCs. Only one ADC can sample the temperature sensor at a time. Software post-processing is required to convert the temperature sensor readings into Kelvin or Celsius values.

Different sample times can be programmed for each channel to accommodate signals with varying source impedances and frequencies. Each ADC also supports range comparison, enabling fast detection of out-of-range values without waiting for a sequencer scan to complete or CPU firmware to evaluate the results.

The ADCs are not available in DeepSleep or Hibernate modes because they require a high-speed clock. The ADC input reference voltage range (VREFH) is 2.7 V to VDDA, while VREFL is VSSA.

3.3.4 Timer/counter/PWM block (TCPWM)

The TCPWM block consists of 16-bit (75 channels) and 32-bit counters (four channels) with a user-programmable period. Twelve of the 16-bit counters include additional features to support motor-control operations.

Each TCPWM counter contains the following:

- **Capture register**: To record the counter value at the time of an event
- **Period register**: To either stop or auto-reload the counter when its value matches the programmed period
- **Compare registers**: To generate signals that are used as PWM duty-cycle outputs

Functional modes

Each counter within the TCPWM block supports several functional modes, including:

- Timer
- Capture

5 VREFL prevents IR drops in the VSSIO and VSSA paths from affecting measurements. When VREFL is properly connected, it reduces or eliminates the impact of IR drops in these paths on the measurements.

- Quadrature
- PWM
- PWM with dead-time insertion (PWM_DT, 8-bit)
- Pseudo-random PWM (PWM_PR)
- Shift-register

Motor-control applications

In motor-control applications, the TCPWM counters support enhanced quadrature mode with advanced features such as:

- Asymmetric PWM generation
- Dead-time insertion (16-bit)
- Assigning different dead times to PWM output signals

The TCPWM block also provides true and complement outputs with a programmable offset, enabling their use as deadband complementary PWM outputs.

Kill input

The TCPWM block includes a kill input (specific to PWM mode) to force outputs into a predetermined state. This is particularly useful in motor-drive systems where an overcurrent condition is detected, and the PWMs driving FETs need to be shut off immediately, without requiring software intervention.

3.3.5 Serial communication blocks (SCB)

XMC5200 contains up to eight serial communication blocks, each configurable to support I^2C, UART, or SPI.

- **I^2C interface**: An SCB can be configured as a full I^2C master (capable of multi-master arbitration) or slave. Each SCB configured for I^2C can operate at speeds of up to 1 Mbps. [6]). Flexible buffering options reduce CPU interrupt overhead and latency, while FIFO buffering for receive and transmit data reduces the need for clock stretching by increasing the time available for the CPU to read data. The I^2C interface is compatible with Standard, Fast-mode, and Fast-mode Plus devices, as specified in the NXP I^2C-bus specification and user manual (UM10204). The I^2C-bus I/O is implemented using GPIO in open-drain mode.[7] [8]
- **UART interface**: When configured as a UART, each SCB provides a full-featured UART with a maximum signaling rate determined by the configured peripheral clock frequency and oversampling rate. The UART supports infrared interface (IrDA) and SmartCard (ISO 7816) protocols, both of which are minor variations of the UART protocol. It also supports a 9-bit multiprocessor mode, allowing peripherals on shared Rx and Tx lines to be addressed.

  Common UART features include:

  - Parity
  - Configurable number of stop bits
  - Break detection
  - Frame error detection

6 Fast-mode Plus I/Os drive level does not support the full bus capacitance at Fast-mode Plus speeds.

7 This implementation is not 100% compliant with the I^2C-bus specification:

- I/Os are not high-voltage compliant
- I/Os do not support the 20-mA sink requirement of Fast-mode Plus
- Leakage specifications are violated when no power is applied

8 Only Port 0, with the slow feature enabled, meets the minimum fall time requirement.

FIFO buffering for transmit and receive data allows the system to tolerate greater CPU service latencies.

- **SPI interface**: The SPI configuration supports the following protocols:
  - Full Motorola SPI
  - TI Synchronous Serial Protocol (SSP), which adds a start pulse for synchronizing SPI-based codecs
  - National Microwire, a half-duplex SPI variant

The SPI interface can use FIFO for efficient data handling and operates at up to 12.5 MHz for the SPI clock. Additionally, SCBs support the Easy SPI (EZSPI) protocol.[9)]

SCB0 includes the following additional features:

- Operates as a slave in DeepSleep mode
- I^2C slave EZ (EZI2C) mode with a 256-B data buffer, enabling multi-byte communication without CPU intervention. [10)]
- I^2C slave externally-clocked operations
- Command/response mode with a 512-B data buffer, enabling multi-byte communication without CPU intervention

3.3.6 CAN FD

XMC5200 supports two CAN FD controller blocks, capable of supporting up to three CAN FD channels. All CAN FD controllers comply with the ISO 11898-1:2015 standard, and an ISO 16845:2015 certificate is available. The controllers also fully implement the time-triggered CAN (TTCAN) protocol, as specified in ISO 11898-4, including TTCAN protocol levels 1 and 2, entirely in hardware.

Message handling: All message-handling functions are managed by the Rx and Tx handlers.

- **Rx Handler**:
  - Manages message acceptance filtering
  - Transfers received messages from the CAN core to the message RAM
  - Provides receive-message status
- **Tx Handler**:
  - Transfers transmit messages from the message RAM to the CAN core
  - Provides transmit-message status

3.3.7 One-time-programmable (OTP) eFuse

XMC5200 include a 1024-bit OTP eFuse memory that stores and provides access to a unique, unalterable identifier or serial number for each device. The eFuses also manage the device life cycle, covering stages such as manufacturing, programming, normal operation, and end-of-life, as well as the security state.

Of the 1024 bits, 192 bits are available for user-defined purposes.

3.3.8 Event generator

The event generator supports the generation of interrupts and triggers in Active mode and interrupts in DeepSleep mode. Event generators are used to trigger specific device functions, such as executing an interrupt handler or starting a SAR ADC conversion, and to implement a cyclic wakeup mechanism in DeepSleep mode.

They provide CPU-free triggers for device functions, reducing CPU involvement in triggering operations. This minimizes overall power consumption and processing overhead.

9 EZSPI is based on the Motorola SPI and operates in any SPI mode (0, 1, 2, or 3). It facilitates master-slave communication while reducing CPU intervention.

10 EZI2C is a unique protocol built on top of the I^2C standard, using indexed memory transfers to reduce CPU intervention.

3.3.9 Trigger multiplexer

XMC5200 supports connecting various peripherals using trigger signals. Triggers are used to notify a peripheral of an event or a change in state, which can initiate or affect actions in other peripherals. The trigger multiplexer routes triggers from a source peripheral to a destination peripheral. Triggers provide active-logic functionality and are typically supported in Active mode.

3.4 I/Os

XMC5200 includes up to 122 programmable I/Os.

The I/Os are organized into logical entities called ports, with each port being a maximum of 8 bits wide. During power-on and reset, all I/Os are forced into a High-Z state. In Hibernate mode, the I/Os are frozen.

Each I/O can generate an interrupt (if enabled), and every port has an associated interrupt request (IRQ) and interrupt service routine (ISR).

The I/O port power source mapping is listed in Table 5. The associated supply determines the V_{OH}, V_{OL}, V_{IH}, and V_{IL} levels when configured for CMOS and industrial thresholds.

Table 5 I/O port power source

Supply	Ports
VDDD	P0, P1, P2, P3, P4, P5, P16, P17, P18, P19, P20, P21, P22, P23
VDDIO_1	P6, P7, P8, P9[11]
VDDIO_2	P10, P11, P12, P13, P14, P15

3.4.1 Port nomenclature

The notation `Px.y` refers to a specific bit `y` within an I/O port `x`. For example, `P4.2` represents "port 4, bit 2."

Each I/O supports the following programmable drive modes:

- High impedance
- Resistive pull-up
- Resistive pull-down
- Open drain with strong pull-down
- Open drain with strong pull-up
- Strong pull-up or pull-down
- Weak pull-up or pull-down

XMC5200 includes two types of programmable I/Os: GPIO Standard and GPIO Enhanced.

3.4.2 GPIO standard (GPIO_STD)

Supports standard industrial signaling across the 2.7-V to 5.5-V VDDIO range. GPIO standard I/Os offer multiple configurable drive levels, drive modes, and selectable input levels.

3.4.3 GPIO enhanced (GPIO_ENH)

Supports extended industrial signaling functionality across the 2.7-V to 5.5-V VDDIO range, offering higher currents at lower voltages with features such as full I^2C timing support and slew-rate control.

11 The I/Os in VDDIO_1 domain refer to the VDDD domain in 64-LQFP package.

Both GPIO_STD and GPIO_ENH include the following features:

- Configurable input threshold (CMOS, TTL, or industrial)
- Hold mode to latch the previous state (used for retaining the I/O state in DeepSleep mode)
- Analog input mode (disables input and output buffers)

3.4.4 Smart I/O

Smart I/O enables Boolean operations on signals either sent to the I/O from the chip's subsystems or received from external sources. XMC5200 has up to five Smart I/O blocks. Smart I/O can operate either synchronously or asynchronously and functions in all device power modes except Hibernate mode.

4 XMC5200 address map

The XMC5200 microcontroller supports the following memory configurations as shown in Figure 2:

- 1088 KB (960 KB + 128 KB) of code flash, used in the single- or dual-bank mode based on the associated bit in the flash control register
  - **Single-bank mode**: 1088 KB
  - **Dual-bank mode**: 544 KB per bank
- 96 KB (72 KB + 24 KB) of work flash, used in the single- or dual-bank mode based on the associated bit in the flash control register
  - **Single-bank mode**: 96 KB
  - **Dual-bank mode**: 48 KB per bank
- 32 KB of secure ROM
- 128 KB of SRAM (the first 2 KB is reserved for internal usage)

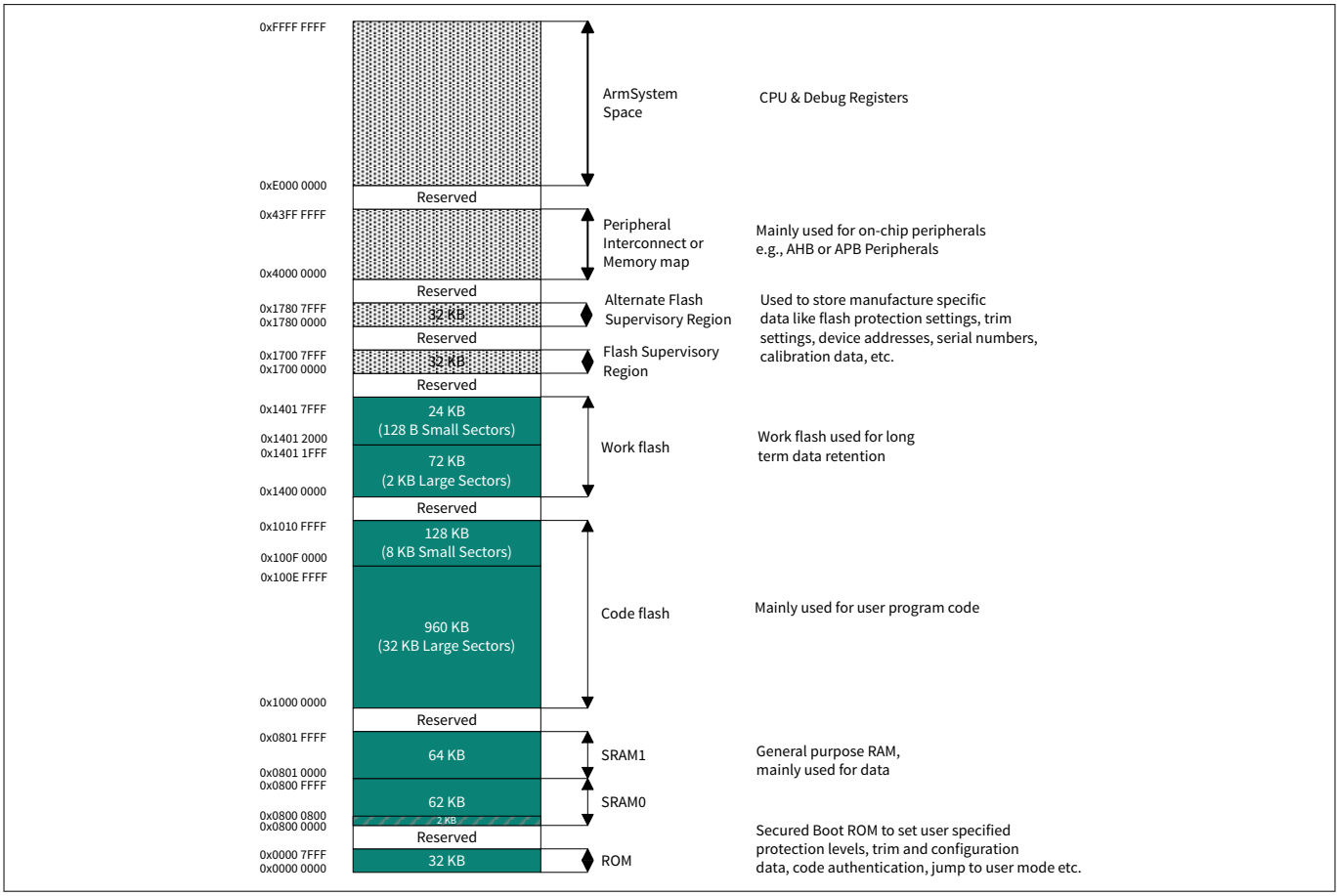


Figure 2 XMC5200 address map

Note:
- *The size representation for memory blocks is not to scale*
- *The first 2 KB of SRAM is reserved and not available for user applications*
- *Ensure that the power for the first 32-KB block of SRAM0 remains enabled or retained in all Active, Low-Power Active (LP Active), Sleep, Low-Power Sleep (LP Sleep), and DeepSleep modes*

5 Flash base address map

The sector mapping of the code-flash and work-flash regions, along with their respective base addresses, is shown from Table 6 to Table 11.

Table 6 Code flash address mapping in single bank mode

Code flash size (KB)	Large sectors (LS)	Small sectors (SS)	Large sector base address	Small sector base address
1088	32 KB × 30	8 KB x 16	0x1000 0000	0x100F 0000

Table 7 Work flash address mapping in single bank mode

Work flash size (KB)	Large sectors (LS)	Small sectors (SS)	Large sector base address	Small sector base address
96	2 KB × 36	128 B × 192	0x1400 0000	0x1401 2000

Table 8 Code flash address mapping in dual bank mode (mapping A)

Code flash size (KB)	First half LS	First half SS	Second half LS	Second half SS	First half LS base address	First half SS base address	Second half LS base address	Second half SS base address
1088	32 KB × 15	8KB × 8	32 KB × 15	8 KB × 8	0x1000 0000	0x1007 8000	0x1200 0000	0x1207 8000

Table 9 Code flash address mapping in dual bank mode (mapping B)

Code flash size (KB)	First half LS	First half SS	Second Half LS	Second half SS	First half LS base address	First half SS base address	Second half LS base address	Second half SS base address
1088	32 KB × 15	8 KB × 8	32 KB × 15	8 KB × 8	0x1200 0000	0x1207 8000	0x1000 0000	0x1007 8000

Table 10 Work flash address mapping in dual bank mode (mapping A)

Work flash size (KB)	First half LS	First half SS	Second half LS	Second half SS	First half LS base address	First half SS base address	Second half LS base address	Second half SS base address
96	2 KB × 18	128 B × 96	2 KB × 18	128 B × 96	0x1400 0000	0x1400 9000	0x1500 0000	0x1500 9000

Table 11 Work flash address mapping in dual bank mode (mapping B)

Work flash size (KB)	First half LS	First half SS	Second half LS	Second half SS	First half LS base address	First half SS base address	Second half LS base address	Second half SS base address
96	2 KB × 18	128 B × 96	2 KB × 18	128 B × 96	0x1500 0000	0x1500 9000	0x1400 0000	0x1400 9000

6 Peripheral I/O map

Table 12 XMC5200 peripheral I/O map

Section	Description	Base address	Instances	Instance size	Group	Slave
PERI	Peripheral interconnect	0x4000 0000	-	-	0	0
	Peripheral group (0, 1, 2, 3, 5, 6, 9)	0x4000 4000	7	0x20		
	Peripheral trigger group	0x4000 8000	11	0x400		
	Peripheral 1:1 trigger group	0x4000 C000	11	0x400		
PERI_MS	Peripheral interconnect, master interface	0x4001 0000	-	-	0	1
	PERI Programmable PPU	0x4001 0000	6[12)]	0x40		
	PERI Fixed PPU	0x4001 0800	458	0x40		
Crypto	Cryptography component	0x4010 0000	-	-	1	0
CPUSS	CPU subsystem (CPUSS)	0x4020 0000	-	-	2	0
FAULT	Fault structure subsystem	0x4021 0000	-	-	2	1
	Fault structures	0x4021 0000	4	0x100		
IPC	Inter process communication	0x4022 0000	-	-	2	2
	IPC structures	0x4022 0000	8	0x20		
	IPC interrupt structures	0x4022 1000	8	0x20		
PROT	Protection	0x4023 0000	-	-	2	3
	Shared memory protection unit structures	0x4023 2000	16	0x40		
	Memory protection unit structures	0x4023 4000	16	0x400		

(table continues...)

12 The six programmable PPUs are configured by the Boot ROM and are available to the user based on access rights. Refer to the device-specific TRM for detailed information about the configuration of these programmable PPUs.

Table 12 (continued) XMC5200 peripheral I/O map

Section	Description	Base address	Instances	Instance size	Group	Slave
FLASHC	Flash controller	0x4024 0000	-	-	2	4
SRSS	System Resources Sub-System Core Registers	0x4026 0000	-	-	2	5
	Clock Supervision High Frequency	0x4026 1400	3	0x10		
	Clock Supervision Reference Frequency	0x4026 1710	1	-		
	Clock Supervision Low Frequency	0x4026 1720	1	-		
	Clock Supervision Internal Low Frequency	0x4026 1730	1	-		
	Multi Counter WDT	0x4026 8000	2	0x100		
	Free Running WDT	0x4026 C000	1	-		
BACKUP	SRSS Backup Domain/RTC	0x4027 0000	-	-	2	6
	Backup Register	0x4027 1000	4	0x04		
P-DMA	P-DMA0 Controller	0x4028 0000	-	-	2	7
	P-DMA0 channel structures	0x4028 8000	89	0x40		
	P-DMA1 Controller	0x4029 0000	-	-	2	8
	P-DMA1 channel structures	0x4029 8000	33	0x40		
M-DMA	M-DMA0 Controller	0x402A 0000	-	-	2	9
	M-DMA0 channels	0x402A 1000	4	0x100		
eFUSE	eFUSE Customer Data (192 bits)	0x402C 0868	6	0x04	2	10
HSIOM	High-Speed I/O Matrix (HSIOM)	0x4030 0000	24	0x10	3	0
GPIO	GPIO port control/configuration	0x4031 0000	24	0x80	3	1

(table continues...)

Table 12 (continued) XMC5200 peripheral I/O map

Section	Description	Base address	Instances	Instance size	Group	Slave
SMARTIO	Programmable I/O configuration	0x4032 0000	-	-	3	2
	SMART I/O port configuration	0x4032 0C00	5	0x100		
TCPWM	Timer/Counter/PWM 0 (TCPWM0)	0x4038 0000	-	-	3	3
	TCPWM0 Group #0 (16-bit)	0x4038 0000	63	0x80		
	TCPWM0 Group #1 (16-bit, Motor control)	0x4038 8000	12	0x80		
	TCPWM0 Group #2 (32-bit)	0x4039 0000	4	0x80		
EVTGEN	Event generator 0 (EVTGEN0)	0x403F 0000	-	-	3	4
	Event generator 0 comparator structures	0x403F 0800	11	0x20		
TTCANFD	CAN0 controller	0x4052 0000	3	0x200	5	1
	Message RAM CAN0	0x4053 0000	-	0x6000		
	CAN1 controller	0x4054 0000	3	0x200	5	2
	Message RAM CAN1	0x4055 0000	-	0x6000		
SCB	Serial Communications Block (SPI/UART/ I2C)	0x4060 0000	8	0x10000	6	0-7
PASS0 SAR	Programmable Analog Subsystem (PASS0)	0x4090 0000	-	-	9	0
	SAR0 channel controller	0x4090 0000	-	-		
	SAR1 channel controller	0x4090 1000	-	-		
	SAR2 channel controller	0x4090 2000	-	-		
	SAR0 channel structures	0x4090 0800	24	0x40		
	SAR1 channel structures	0x4090 1800	32	0x40		

(table continues...)

Table 12 (continued) XMC5200 peripheral I/O map

Section	Description	Base address	Instances	Instance size	Group	Slave
	SAR2 channel structures	0x4090 2800	8	0x40		

7 XMC5200 clock diagram

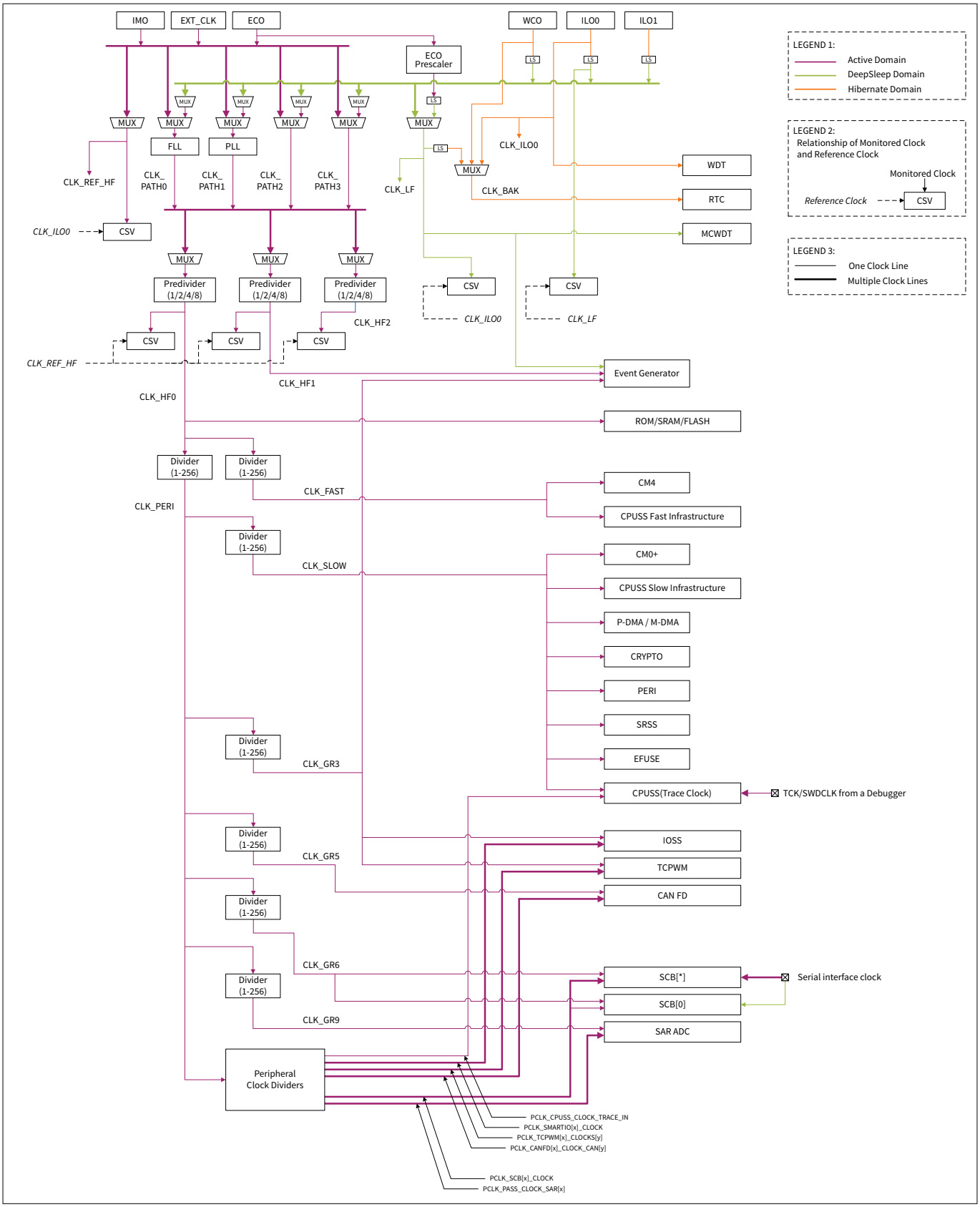


Figure 3 XMC5200 clock diagram

8 XMC5200 CPU startup sequence

The following steps describe the startup sequence:

1. System reset (at 0x0000 0000)
2. CM0+ executes ROM boot (at 0x0000 0004):
   - **a.** Applies trims
   - **b.** Configures Debug Access Port (DAP) access restrictions and system protection based on eFuse and supervisory flash settings
   - **c.** Authenticates flash boot (only in the secure lifecycle stage) and transfers control to it
3. CM0+ executes flash boot (from Supervisory flash at 0x1700 2000):
   - **a.** Debug pins are configured according to the SWD/JTAG specification. [13)]
   - **b.** Sets the CM0+ vector offset register (CM0_VTOR, part of the Arm system space) to the start of flash (at 0x1000 0000)
   - **c.** CM0+ branches to its reset handler and starts execution
4. CM0+ starts execution:
   - **a.** Moves the CM0+ vector table to SRAM and updates the CM0+ vector table base
   - **b.** Sets the CM4_VECTOR_TABLE_BASE (at 0x0000 0200) to the location of the CM4 vector table specified in flash (defined in the CM4 linker configuration file)
   - **c.** Releases the CM4 core from reset
   - **d.** CM0+ continues executing the user application
5. CM4 starts execution directly from code-flash or SRAM
   - **a.** CM4 branches to its reset handler and begins executing the CM4 user application

13 WD/JTAG pins are reconfigured from their default GPIO mode to support debugging after the boot process. See Table 14 for pin assignments.

9 Pin assignment

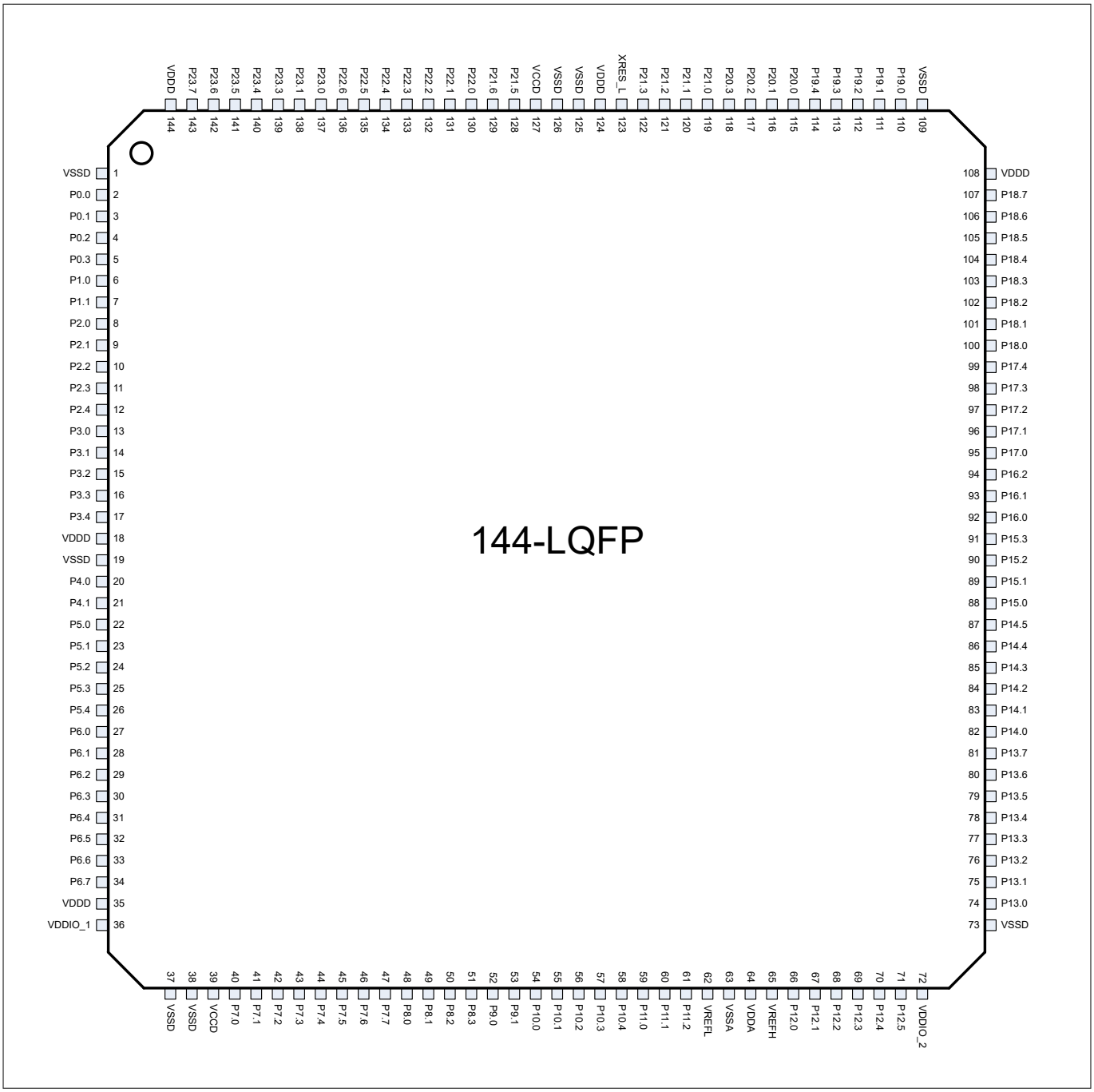


Figure 4 144-LQFP pin assignment

9 Pin assignment

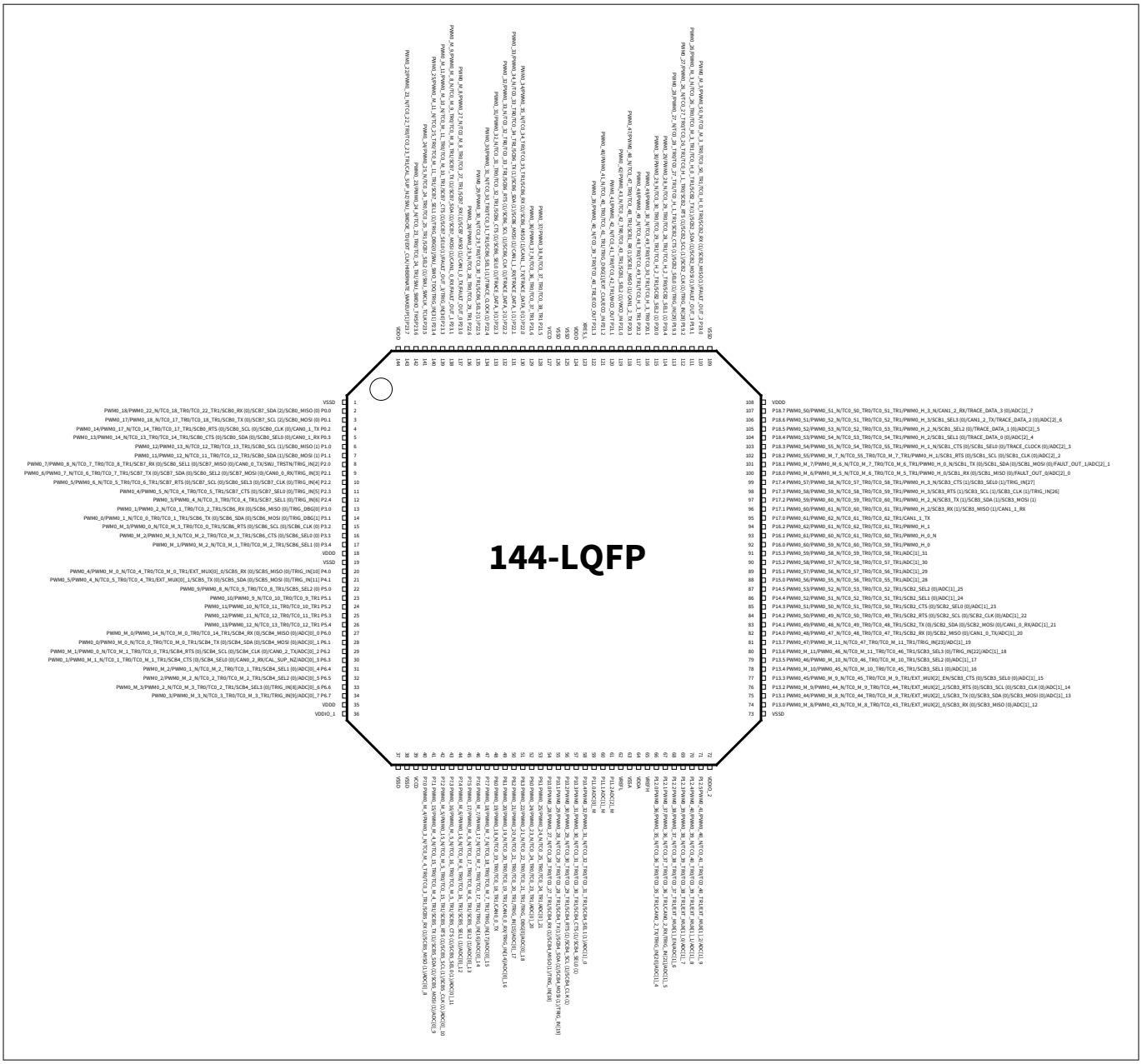


Figure 5 144-LQFP pin assignment with alternate functions

9 Pin assignment

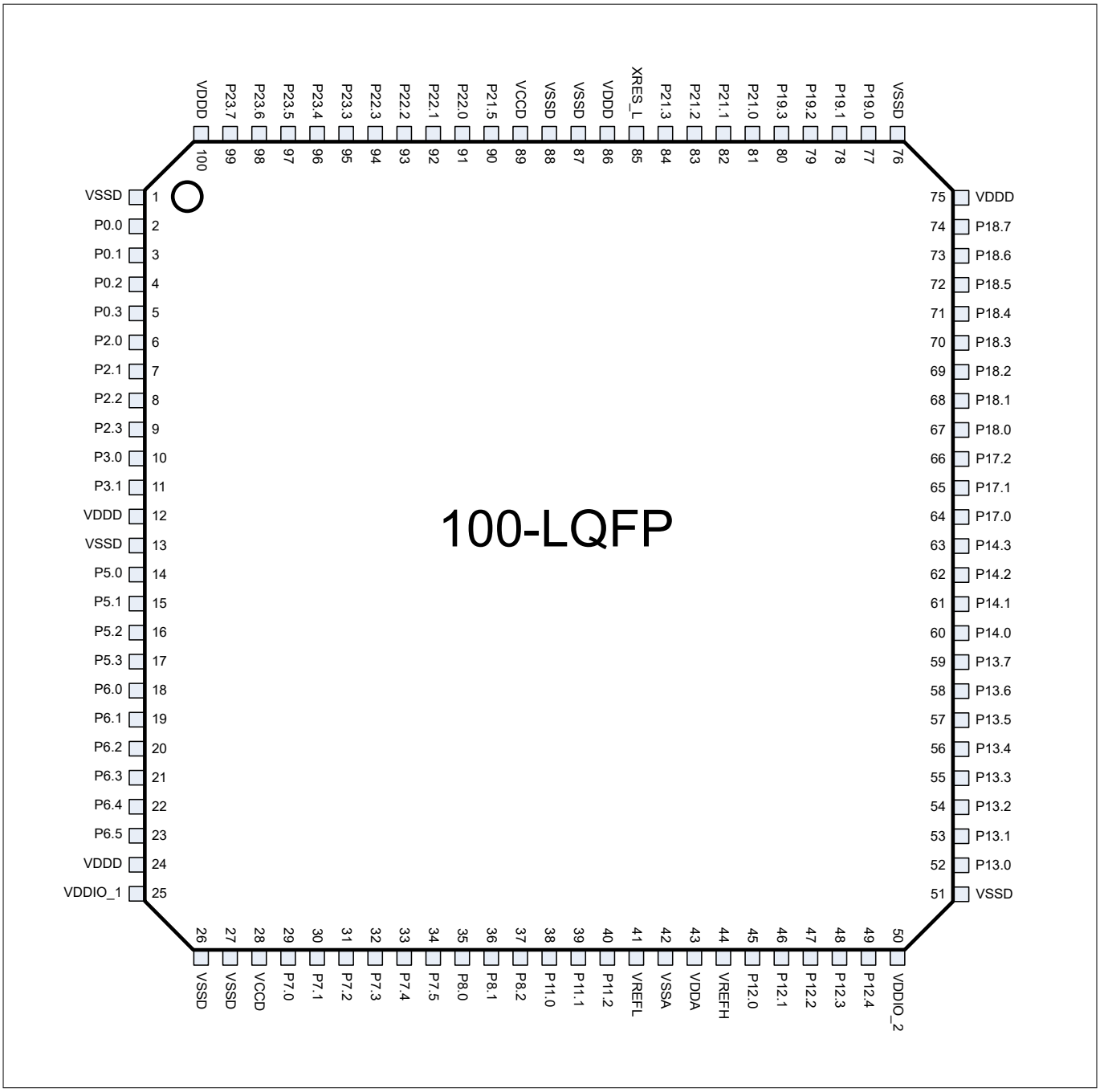


Figure 6 100-LQFP pin assignment

9 Pin assignment

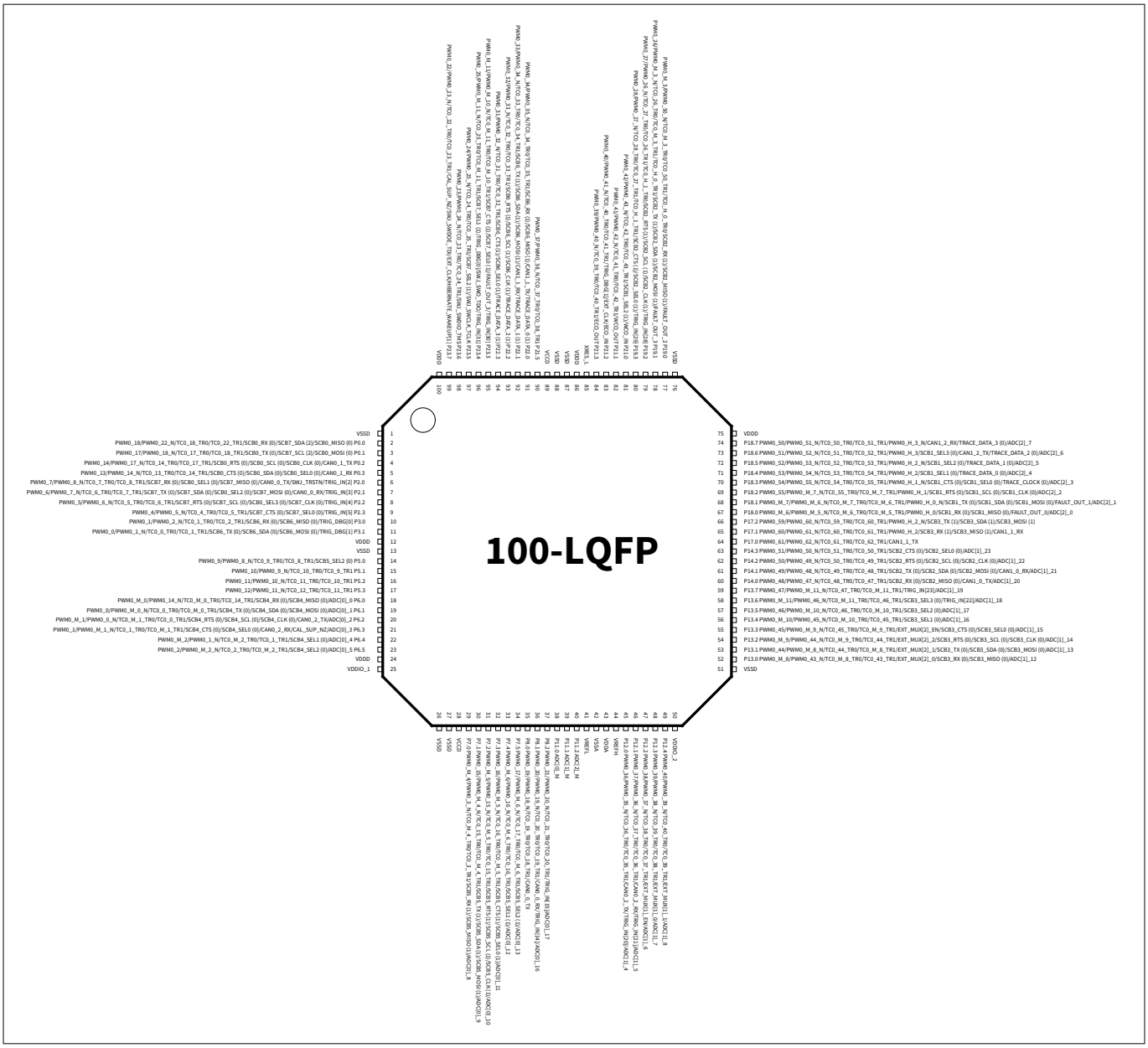


Figure 7 100-LQFP pin assignment with alternate functions

9 Pin assignment

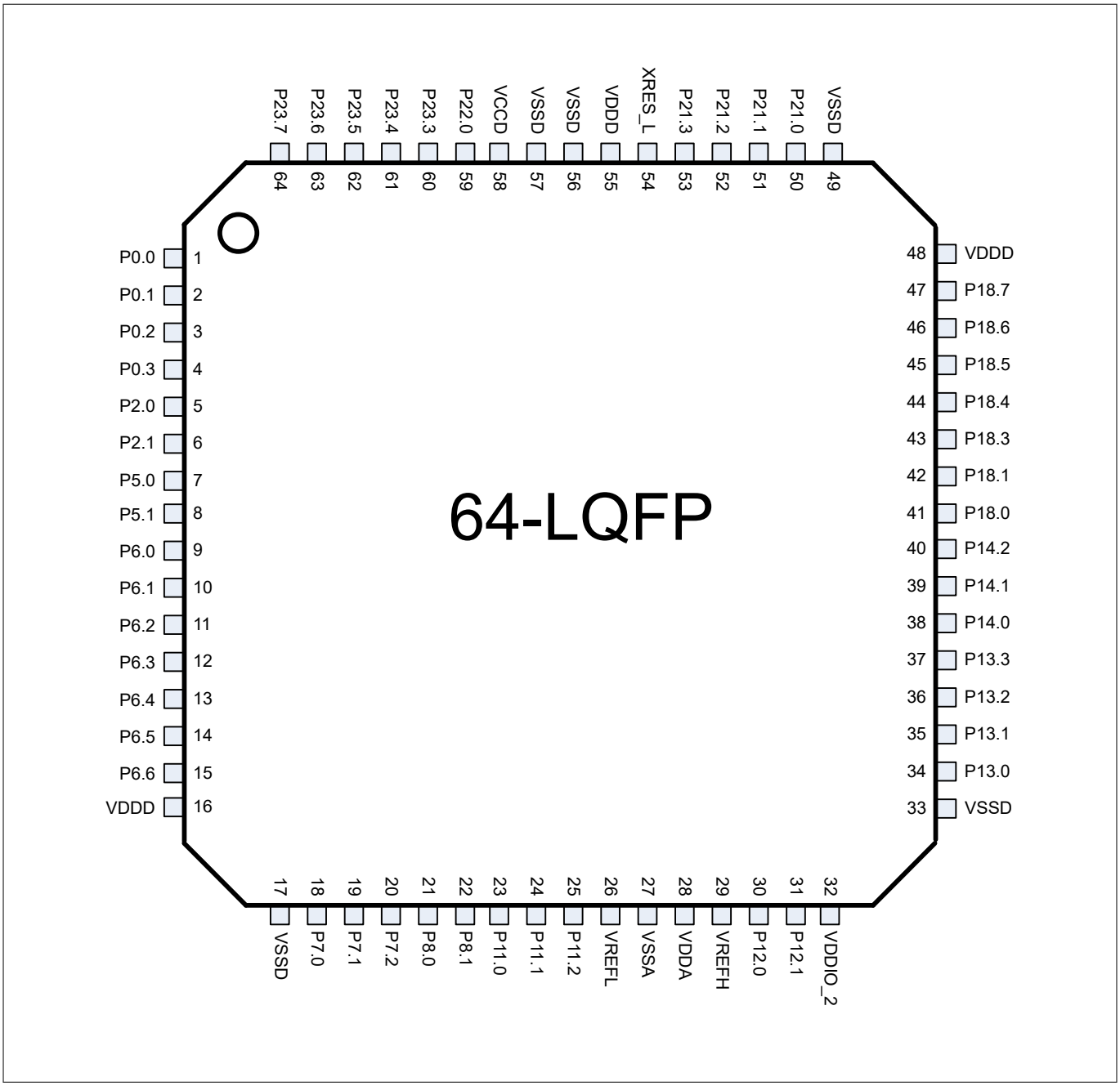


Figure 8 64-LQFP pin assignment

9 Pin assignment

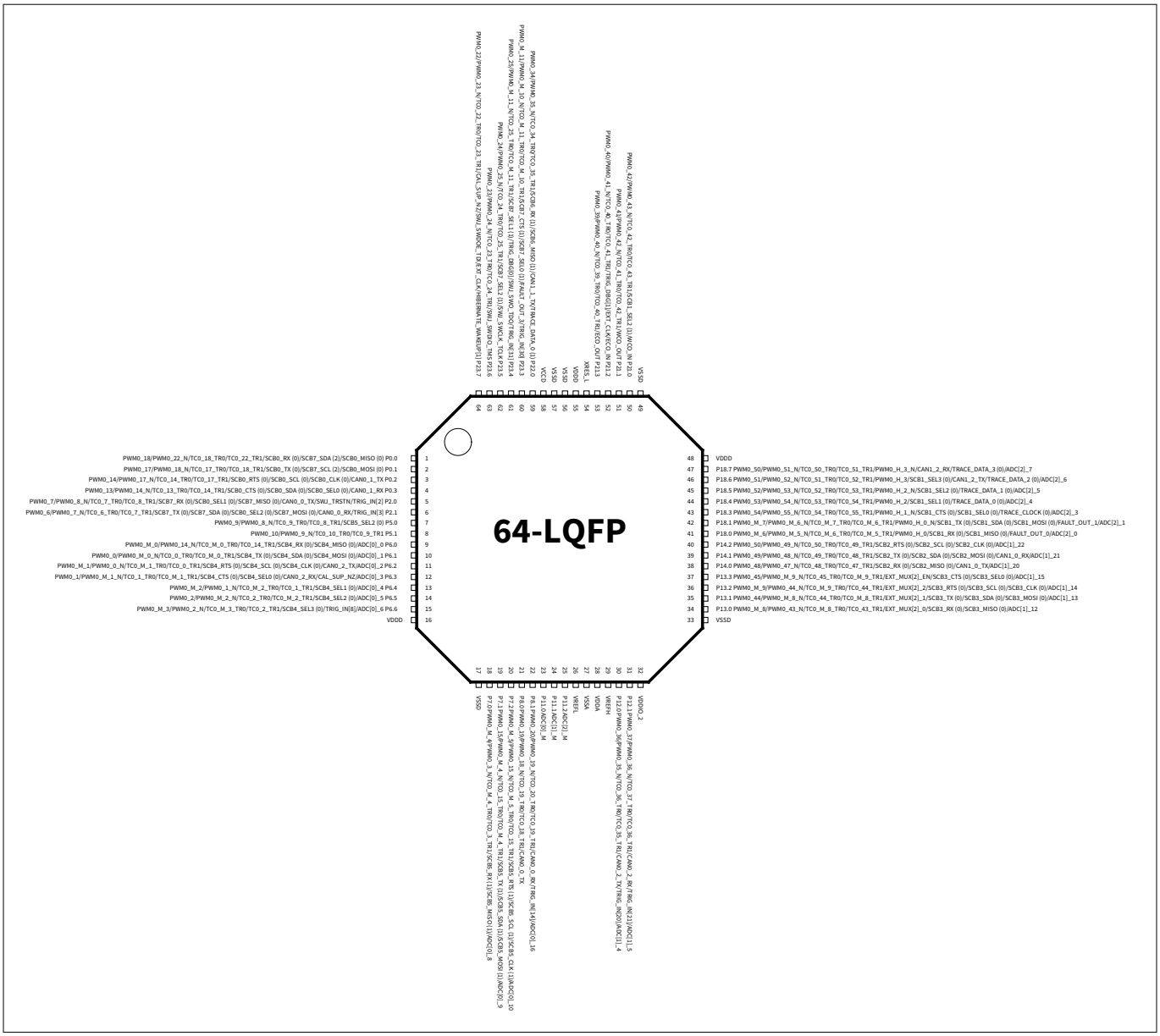


Figure 9 64-LQFP pin assignment with alternate functions

10 High-speed I/O matrix connections

Table 13 HSIOM connections reference

Name	Number	Description
HSIOM_SEL_GPIO	0	GPIO controls 'out'
HSIOM_SEL_GPIO_DSI	1	Reserved
HSIOM_SEL_DSI_DSI	2	
HSIOM_SEL_DSI_GPIO	3	
HSIOM_SEL_AMUXA	4	
HSIOM_SEL_AMUXB	5	
HSIOM_SEL_AMUXA_DSI	6	
HSIOM_SEL_AMUXB_DSI	7	
HSIOM_SEL_ACT_0	8	Active functionality 0
HSIOM_SEL_ACT_1	9	Active functionality 1
HSIOM_SEL_ACT_2	10	Active functionality 2
HSIOM_SEL_ACT_3	11	Active functionality 3
HSIOM_SEL_DS_0	12	DeepSleep functionality 0
HSIOM_SEL_DS_1	13	DeepSleep functionality 1
HSIOM_SEL_DS_2	14	DeepSleep functionality 2
HSIOM_SEL_DS_3	15	DeepSleep functionality 3
HSIOM_SEL_ACT_4	16	Active functionality 4
HSIOM_SEL_ACT_5	17	Active functionality 5
HSIOM_SEL_ACT_6	18	Active functionality 6
HSIOM_SEL_ACT_7	19	Active functionality 7
HSIOM_SEL_ACT_8	20	Active functionality 8
HSIOM_SEL_ACT_9	21	Active functionality 9
HSIOM_SEL_ACT_10	22	Active functionality 10
HSIOM_SEL_ACT_11	23	Active functionality 11
HSIOM_SEL_ACT_12	24	Active functionality 12
HSIOM_SEL_ACT_13	25	Active functionality 13
HSIOM_SEL_ACT_14	26	Active functionality 14
HSIOM_SEL_ACT_15	27	Active functionality 15
HSIOM_SEL_DS_4	28	DeepSleep functionality 4
HSIOM_SEL_DS_5	29	DeepSleep functionality 5
HSIOM_SEL_DS_6	30	DeepSleep functionality 6
HSIOM_SEL_DS_7	31	DeepSleep functionality 7

11 Package pin list and alternate functions

Most pins have alternate functionality, as specified in Table 14.

Port 11 has the following additional features:

- Capability to pass full-level analog signals to the SAR without clipping to VDDD when VDDD < VDDA
- Ability to simultaneously capture all three ADC signals with the highest priority (ADC[0:2]_M)
- Lower noise for the most sensitive sensors

***Note**: See Table 17 for detailed information on pin multiplexer abbreviations.*

***Note**: For any function marked with an identifier (n), the AC timing is guaranteed only within the respective group "n".*

Table 14 Pin selector and alternate pin functions in DeepSleep (DS) mode, Analog, Smart I/O

Name		Package pins			DeepSleep mapping[14)]			Analog	Smart I/O
	HCon#0[1 5)]	144-LQFP	100-LQFP	64-LQFP	HCon#14	HCon#29	HCon#30		
	GPIO				DS #0[16)]	DS #1	DS #2		
P0.0	GPIO_ENH	2	2	1	-	-	SCB0_MISO (0)	-	-
P0.1	GPIO_ENH	3	3	2	-	-	SCB0_MOSI (0)	-	-
P0.2	GPIO_ENH	4	4	3	SCB0_SCL (0)	-	SCB0_CLK (0)	-	-
P0.3	GPIO_ENH	5	5	4	SCB0_SDA (0)	-	SCB0_SEL0 (0)	-	-
P1.0	GPIO_STD	6	NA	NA	SCB0_SCL (1)	-	SCB0_MISO (1)	-	-
P1.1	GPIO_STD	7	NA	NA	SCB0_SDA (1)	-	SCB0_MOSI (1)	-	-

(table continues...)

14 All port pin functions available in DeepSleep mode are also functional in Active mode.
15 High Speed I/O matrix connection (HCon) reference as per Table 13

11 Package pin list and alternate functions

Table 14 (continued) Pin selector and alternate pin functions in DeepSleep (DS) mode, Analog, Smart I/O

Name	Package pins			DeepSleep mapping[14]				Analog	Smart I/O
	HCon#0[1][5]	144-LQFP	100-LQFP	64-LQFP	HCon#14	HCon#29	HCon#30		
	GPIO				DS #0[16]	DS #1	DS #2		
P2.0	GPIO_STD	8	6	5	-	SWJ_TRSTN	SCB0_SEL1 (0)	-	-
P2.1	GPIO_STD	9	7	6	-	-	SCB0_SEL2 (0)	-	-
P2.2	GPIO_STD	10	8	NA	-	-	SCB0_SEL3 (0)	-	-
P2.3	GPIO_STD	11	9	NA	-	-	-	-	-
P2.4	GPIO_STD	12	NA	NA	-	-	-	-	-
P3.0	GPIO_STD	13	10	NA	-	-	-	-	-
P3.1	GPIO_STD	14	11	NA	-	-	-	-	-
P3.2	GPIO_STD	15	NA	NA	-	-	-	-	-
P3.3	GPIO_STD	16	NA	NA	-	-	-	-	-
P3.4	GPIO_STD	17	NA	NA	-	-	-	-	-
P4.0	GPIO_STD	20	NA	NA	-	-	-	-	-
P4.1	GPIO_STD	21	NA	NA	-	-	-	-	-

(table continues...)

16 DeepSleep ordering (DS#0, DS#1, DS#2) does not impact the selection of alternate functions. The HSIOM module manages the assignment of individual alternate functions.

Table 14 (continued) Pin selector and alternate pin functions in DeepSleep (DS) mode, Analog, Smart I/O

Name		Package pins			DeepSleep mapping[14]			Analog	Smart I/O
	HCon#0[1] [5]	144-LQFP	100-LQFP	64-LQFP	HCon#14	HCon#29	HCon#30		
	GPIO				DS #0[16]	DS #1	DS #2		
P5.0	GPIO_STD	22	14	7	-	-	-	-	-
P5.1	GPIO_STD	23	15	8	-	-	-	-	-
P5.2	GPIO_STD	24	16	NA	-	-	-	-	-
P5.3	GPIO_STD	25	17	NA	-	-	-	-	-
P5.4	GPIO_STD	26	NA	NA	-	-	-	-	-
P6.0	GPIO_STD	27	18	9	-	-	-	ADC[0]_0	-
P6.1	GPIO_STD	28	19	10	-	-	-	ADC[0]_1	-
P6.2	GPIO_STD	29	20	11	-	-	-	ADC[0]_2	-
P6.3	GPIO_STD	30	21	12	-	-	-	ADC[0]_3	-
P6.4	GPIO_STD	31	22	13	-	-	-	ADC[0]_4	-
P6.5	GPIO_STD	32	23	14	-	-	-	ADC[0]_5	-
P6.6	GPIO_STD	33	NA	15	-	-	-	ADC[0]_6	-

(table continues...)

14 All port pin functions available in DeepSleep mode are also functional in Active mode.
15 High Speed I/O matrix connection (HCon) reference as per Table 13

11 Package pin list and alternate functions

Table 14 (continued) Pin selector and alternate pin functions in DeepSleep (DS) mode, Analog, Smart I/O

Name	Package pins			DeepSleep mapping[14]			Analog	Smart I/O
	HCon#0[1] [5]	144-LQFP	100-LQFP	64-LQFP	HCon#14	HCon#29	HCon#30		
	GPIO				DS #0[16]	DS #1	DS #2		
P6.7	GPIO_STD	34	NA	NA	-	-	-	ADC[0]_7	-
P7.0	GPIO_STD	40	29	18	-	-	-	ADC[0]_8	-
P7.1	GPIO_STD	41	30	19	-	-	-	ADC[0]_9	-
P7.2	GPIO_STD	42	31	20	-	-	-	ADC[0]_10	-
P7.3	GPIO_STD	43	32	NA	-	-	-	ADC[0]_11	-
P7.4	GPIO_STD	44	33	NA	-	-	-	ADC[0]_12	-
P7.5	GPIO_STD	45	34	NA	-	-	-	ADC[0]_13	-
P7.6	GPIO_STD	46	NA	NA	-	-	-	ADC[0]_14	-
P7.7	GPIO_STD	47	NA	NA	-	-	-	ADC[0]_15	-
P8.0	GPIO_STD	48	35	21	-	-	-	-	-
P8.1	GPIO_STD	49	36	22	-	-	-	ADC[0]_16	-
P8.2	GPIO_STD	50	37	NA	-	-	-	ADC[0]_17	-

(table continues...)

14 All port pin functions available in DeepSleep mode are also functional in Active mode.
15 High Speed I/O matrix connection (HCon) reference as per Table 13

11 Package pin list and alternate functions

Table 14 (continued) Pin selector and alternate pin functions in DeepSleep (DS) mode, Analog, Smart I/O

Name	Package pins			DeepSleep mapping[14]				Analog	Smart I/O
	HCon#0[1] [5]	144-LQFP	100-LQFP	64-LQFP	HCon#14	HCon#29	HCon#30		
	GPIO				DS #0[16]	DS #1	DS #2		
P8.3	GPIO_STD	51	NA	NA	-	-	-	ADC[0]_18	-
P9.0	GPIO_STD	52	NA	NA	-	-	-	ADC[0]_20	-
P9.1	GPIO_STD	53	NA	NA	-	-	-	ADC[0]_21	-
P10.0	GPIO_STD	54	NA	NA	-	-	-	-	-
P10.1	GPIO_STD	55	NA	NA	-	-	-	-	-
P10.2	GPIO_STD	56	NA	NA	-	-	-	-	-
P10.3	GPIO_STD	57	NA	NA	-	-	-	-	-
P10.4	GPIO_STD	58	NA	NA	-	-	-	ADC[1]_0	-
P11.0	GPIO_STD	59	38	23	-	-	-	ADC[0]_M	-
P11.1	GPIO_STD	60	39	24	-	-	-	ADC[1]_M	-
P11.2	GPIO_STD	61	40	25	-	-	-	ADC[2]_M	-
P12.0	GPIO_STD	66	45	30	-	-	-	ADC[1]_4	SMARTIO12_0

(table continues...)

16 DeepSleep ordering (DS#0, DS#1, DS#2) does not impact the selection of alternate functions. The HSIOM module manages the assignment of individual alternate functions.

14 All port pin functions available in DeepSleep mode are also functional in Active mode.

11 Package pin list and alternate functions

Table 14 (continued) Pin selector and alternate pin functions in DeepSleep (DS) mode, Analog, Smart I/O

Name	Package pins			DeepSleep mapping[14]			Analog	Smart I/O
	HCon#0[1][5]	144-LQFP	100-LQFP	64-LQFP	HCon#14	HCon#29	HCon#30		
	GPIO				DS #0[16]	DS #1	DS #2		
P12.1	GPIO_STD	67	46	31	-	-	-	ADC[1]_5	SMARTIO12_1
P12.2	GPIO_STD	68	47	NA	-	-	-	ADC[1]_6	SMARTIO12_2
P12.3	GPIO_STD	69	48	NA	-	-	-	ADC[1]_7	SMARTIO12_3
P12.4	GPIO_STD	70	49	NA	-	-	-	ADC[1]_8	SMARTIO12_4
P12.5	GPIO_STD	71	NA	NA	-	-	-	ADC[1]_9	SMARTIO12_5
P13.0	GPIO_STD	74	52	34	-	-	-	ADC[1]_12	SMARTIO13_0
P13.1	GPIO_STD	75	53	35	-	-	-	ADC[1]_13	SMARTIO13_1
P13.2	GPIO_STD	76	54	36	-	-	-	ADC[1]_14	SMARTIO13_2
P13.3	GPIO_STD	77	55	37	-	-	-	ADC[1]_15	SMARTIO13_3
P13.4	GPIO_STD	78	56	NA	-	-	-	ADC[1]_16	SMARTIO13_4
P13.5	GPIO_STD	79	57	NA	-	-	-	ADC[1]_17	SMARTIO13_5
P13.6	GPIO_STD	80	58	NA	-	-	-	ADC[1]_18	SMARTIO13_6

(table continues...)

14 All port pin functions available in DeepSleep mode are also functional in Active mode.
15 High Speed I/O matrix connection (HCon) reference as per Table 13

11 Package pin list and alternate functions

Table 14 (continued) Pin selector and alternate pin functions in DeepSleep (DS) mode, Analog, Smart I/O

Name	Package pins				DeepSleep mapping[14]			Analog	Smart I/O
	HCon#0[1][5]	144-LQFP	100-LQFP	64-LQFP	HCon#14	HCon#29	HCon#30		
	GPIO				DS #0[16]	DS #1	DS #2		
P13.7	GPIO_STD	81	59	NA	-	-	-	ADC[1]_19	SMARTIO13_7
P14.0	GPIO_STD	82	60	38	-	-	-	ADC[1]_20	SMARTIO14_0
P14.1	GPIO_STD	83	61	39	-	-	-	ADC[1]_21	SMARTIO14_1
P14.2	GPIO_STD	84	62	40	-	-	-	ADC[1]_22	SMARTIO14_2
P14.3	GPIO_STD	85	63	NA	-	-	-	ADC[1]_23	SMARTIO14_3
P14.4	GPIO_STD	86	NA	NA	-	-	-	ADC[1]_24	SMARTIO14_4
P14.5	GPIO_STD	87	NA	NA	-	-	-	ADC[1]_25	SMARTIO14_5
P15.0	GPIO_STD	88	NA	NA	-	-	-	ADC[1]_28	SMARTIO15_0
P15.1	GPIO_STD	89	NA	NA	-	-	-	ADC[1]_29	SMARTIO15_1
P15.2	GPIO_STD	90	NA	NA	-	-	-	ADC[1]_30	SMARTIO15_2
P15.3	GPIO_STD	91	NA	NA	-	-	-	ADC[1]_31	SMARTIO15_3
P16.0	GPIO_STD	92	NA	NA	-	-	-	-	-

(table continues...)

14 All port pin functions available in DeepSleep mode are also functional in Active mode.
15 High Speed I/O matrix connection (HCon) reference as per Table 13

11 Package pin list and alternate functions

Table 14 (continued) Pin selector and alternate pin functions in DeepSleep (DS) mode, Analog, Smart I/O

Name	Package pins			DeepSleep mapping[14]			Analog	Smart I/O
	HCon#0[1][5]	144-LQFP	100-LQFP	64-LQFP	HCon#14	HCon#29	HCon#30		
	GPIO				DS #0[16]	DS #1	DS #2		
P16.1	GPIO_STD	93	NA	NA	-	-	-	-	-
P16.2	GPIO_STD	94	NA	NA	-	-	-	-	-
P17.0	GPIO_STD	95	64	NA	-	-	-	-	SMARTIO17_0
P17.1	GPIO_STD	96	65	NA	-	-	-	-	SMARTIO17_1
P17.2	GPIO_STD	97	66	NA	-	-	-	-	SMARTIO17_2
P17.3	GPIO_STD	98	NA	NA	-	-	-	-	SMARTIO17_3
P17.4	GPIO_STD	99	NA	NA	-	-	-	-	SMARTIO17_4
P18.0	GPIO_STD	100	67	41	-	-	-	ADC[2]_0	-
P18.1	GPIO_STD	101	68	42	-	-	-	ADC[2]_1	-
P18.2	GPIO_STD	102	69	NA	-	-	-	ADC[2]_2	-
P18.3	GPIO_STD	103	70	43	-	-	-	ADC[2]_3	-
P18.4	GPIO_STD	104	71	44	-	-	-	ADC[2]_4	-

(table continues...)

14 All port pin functions available in DeepSleep mode are also functional in Active mode.
15 High Speed I/O matrix connection (HCon) reference as per Table 13

Table 14 (continued) Pin selector and alternate pin functions in DeepSleep (DS) mode, Analog, Smart I/O

Name		Package pins			DeepSleep mapping[14)]			Analog	Smart I/O
	HCon#0[1 5)]	144-LQFP	100-LQFP	64-LQFP	HCon#14	HCon#29	HCon#30		
	GPIO				DS #0[16)]	DS #1	DS #2		
P18.5	GPIO_STD	105	72	45	-	-	-	ADC[2]_5	-
P18.6	GPIO_STD	106	73	46	-	-	-	ADC[2]_6	-
P18.7	GPIO_STD	107	74	47	-	-	-	ADC[2]_7	-
P19.0	GPIO_STD	110	77	NA	-	-	-	-	-
P19.1	GPIO_STD	111	78	NA	-	-	-	-	-
P19.2	GPIO_STD	112	79	NA	-	-	-	-	-
P19.3	GPIO_STD	113	80	NA	-	-	-	-	-
P19.4	GPIO_STD	114	NA	NA	-	-	-	-	-
P20.0	GPIO_STD	115	NA	NA	-	-	-	-	-
P20.1	GPIO_STD	116	NA	NA	-	-	-	-	-
P20.2	GPIO_STD	117	NA	NA	-	-	-	-	-
P20.3	GPIO_STD	118	NA	NA	-	-	-	-	-

(table continues...)

14 All port pin functions available in DeepSleep mode are also functional in Active mode.
15 High Speed I/O matrix connection (HCon) reference as per Table 13

11 Package pin list and alternate functions

Table 14 (continued) Pin selector and alternate pin functions in DeepSleep (DS) mode, Analog, Smart I/O

Name	Package pins			DeepSleep mapping[14]			Analog	Smart I/O
	HCon#0[1][5]	144-LQFP	100-LQFP	64-LQFP	HCon#14	HCon#29	HCon#30		
	GPIO				DS #0[16]	DS #1	DS #2		
P21.0	GPIO_STD	119	81	50	-	-	-	WCO_IN[17]	-
P21.1	GPIO_STD	120	82	51	-	-	-	WCO_OUT[18]	-
P21.2	GPIO_STD	121	83	52	-	-	-	ECO_IN[18]	-
P21.3	GPIO_STD	122	84	53	-	-	-	ECO_OUT[18]	-
P21.5	GPIO_STD	128	90	NA	-	-	-	-	-
P21.6	GPIO_STD	129	NA	NA	-	-	-	-	-
P22.0	GPIO_STD	130	91	59	-	-	-	-	-
P22.1	GPIO_STD	131	92	NA	-	-	-	-	-
P22.2	GPIO_STD	132	93	NA	-	-	-	-	-
P22.3	GPIO_STD	133	94	NA	-	-	-	-	-
P22.4	GPIO_STD	134	NA	NA	-	-	-	-	-
P22.5	GPIO_STD	135	NA	NA	-	-	-	-	-

(table continues...)

14 All port pin functions available in DeepSleep mode are also functional in Active mode.
15 High Speed I/O matrix connection (HCon) reference as per Table 13

11 Package pin list and alternate functions

Table 14 (continued) Pin selector and alternate pin functions in DeepSleep (DS) mode, Analog, Smart I/O

Name	Package pins			DeepSleep mapping[14]			Analog	Smart I/O
	HCon#0[1][5]	144-LQFP	100-LQFP	64-LQFP	HCon#14	HCon#29	HCon#30		
	GPIO				DS #0[16]	DS #1	DS #2		
P22.6	GPIO_STD	136	NA	NA	-	-	-	-	-
P23.0	GPIO_STD	137	NA	NA	-	-	-	-	-
P23.1	GPIO_STD	138	NA	NA	-	-	-	-	-
P23.3	GPIO_STD	139	95	60	-	-	-	-	-
P23.4	GPIO_STD	140	96	61	-	SWJ_SWO_TDO	-	-	-
P23.5	GPIO_STD	141	97	62	-	SWJ_SWCLK_TCLK	-	-	-
P23.6	GPIO_STD	142	98	63	-	SWJ_SWDIO_TMS	-	-	-
P23.7	GPIO_STD	143	99	64	-	SWJ_SWDOE_TDI	-	HIBERNATE_WAKEUP[1]	-
XRES_L	-	123	85	54	-	-	-	-	-

14 All port pin functions available in DeepSleep mode are also functional in Active mode.
15 High Speed I/O matrix connection (HCon) reference as per Table 13

12 Power pin assignments

Table 15 Power pin assignments

Name	Package			Remarks
	64-LQFP	100-LQFP	144-LQFP	
VDDD	55, 48, 16	100, 86, 75, 24, 12	144, 124, 108, 35, 18	Main digital supply
VSSD	57, 56, 49, 33, 17	88, 87, 76, 51, 27, 26, 13, 1	126, 125, 109, 73, 38, 37, 19, 1	Main digital ground
VDDIO_1	NA	25	36	I/O supply for group 1
VDDIO_2	32	50	72	I/O supply for group 2
VCCD[19]	58	89, 28	127, 39	Main regulated supply. Driven by LDO regulator
VREFH	29	44	65	High reference voltage for SAR
VREFL	26	41	62	Low reference voltage for SAR
VDDA	28	43	64	Main analog supply (for PASS SAR)
VSSA	27	42	63	Main analog ground

14 All port pin functions available in DeepSleep mode are also functional in Active mode.
15 High Speed I/O matrix connection (HCon) reference as per Table 13
16 DeepSleep ordering (DS#0, DS#1, DS#2) does not impact the selection of alternate functions. The HSIOM module manages the assignment of individual alternate functions.
17 GPIO pins supporting oscillator functions (WCO or ECO) must be configured as high-impedance when the oscillator is enabled.
18 GPIO pins supporting oscillator functions (WCO or ECO) must be configured as high-impedance when the oscillator is enabled.
19 The VCCD pins must be connected together to ensure a low-impedance connection. (see the requirement in Figure 12).

13 Alternate function pin assignments

***Note**: Refer to Table 17 for more information on pin multiplexer abbreviations used.*

***Note**: For any function marked with an identifier (n), the AC timing is only guaranteed within the respective group "n".*

Table 16 Alternate pin functions in active mode

Name	Active mapping										
	HCon#8[20]	HCon#9	HCon#10	HCon#11	HCon#16	HCon#17	HCon#18	HCon#19	HCon#21	HCon#26	HCon#27
	ACT#0[21]	ACT#1	ACT#2	ACT#3	ACT#4	ACT#5	ACT#6	ACT#7	ACT#9	ACT#14	ACT#15
P0.0	PWM0_18	PWM0_22_N	TC0_18_TR0	TC0_22_TR1		SCB0_RX (0)	SCB7_SDA (2)				
P0.1	PWM0_17	PWM0_18_N	TC0_17_TR0	TC0_18_TR1		SCB0_TX (0)	SCB7_SCL (2)				
P0.2	PWM0_14	PWM0_17_N	TC0_14_TR0	TC0_17_TR1		SCB0_RTS (0)			CAN0_1_TX		
P0.3	PWM0_13	PWM0_14_N	TC0_13_TR0	TC0_14_TR1		SCB0_CTS (0)			CAN0_1_RX		
P1.0	PWM0_12	PWM0_13_N	TC0_12_TR0	TC0_13_TR1							
P1.1	PWM0_11	PWM0_12_N	TC0_11_TR0	TC0_12_TR1							
P2.0	PWM0_7	PWM0_8_N	TC0_7_TR0	TC0_8_TR1		SCB7_RX (0)		SCB7_MISO (0)	CAN0_0_TX	TRIG_IN[2]	
P2.1	PWM0_6	PWM0_7_N	TC0_6_TR0	TC0_7_TR1		SCB7_TX (0)	SCB7_SDA (0)	SCB7_MOSI (0)	CAN0_0_RX	TRIG_IN[3]	
P2.2	PWM0_5	PWM0_6_N	TC0_5_TR0	TC0_6_TR1		SCB7_RTS (0)	SCB7_SCL (0)	SCB7_CLK (0)		TRIG_IN[4]	
P2.3	PWM0_4	PWM0_5_N	TC0_4_TR0	TC0_5_TR1		SCB7_CTS (0)		SCB7_SEL0 (0)		TRIG_IN[5]	
P2.4	PWM0_3	PWM0_4_N	TC0_3_TR0	TC0_4_TR1				SCB7_SEL1 (0)		TRIG_IN[6]	
P3.0	PWM0_1	PWM0_2_N	TC0_1_TR0	TC0_2_TR1		SCB6_RX (0)		SCB6_MISO (0)	CAN0_3_TX		TRIG_DBG[0]
P3.1	PWM0_0	PWM0_1_N	TC0_0_TR0	TC0_1_TR1		SCB6_TX (0)	SCB6_SDA (0)	SCB6_MOSI (0)	CAN0_3_RX		TRIG_DBG[1]

(table continues...)

20 High-Speed I/O matrix connection (HCon) reference as per Table 13.

21 Active Mode ordering (ACT#0, ACT#1, and so on) does not have any impact on choosing any alternate functions; HSIOM module will handle the individual alternate function assignment.

Table 16 (continued) Alternate pin functions in active mode

Name	Active mapping										
	HCon#8[20]	HCon#9	HCon#10	HCon#11	HCon#16	HCon#17	HCon#18	HCon#19	HCon#21	HCon#26	HCon#27
	ACT#0[21]	ACT#1	ACT#2	ACT#3	ACT#4	ACT#5	ACT#6	ACT#7	ACT#9	ACT#14	ACT#15
P3.2	PWM0_M_3	PWM0_0_N	TC0_M_3_TR0	TC0_0_TR1		SCB6_RTS (0)	SCB6_SCL (0)	SCB6_CLK (0)			
P3.3	PWM0_M_2	PWM0_M_3_N	TC0_M_2_TR0	TC0_M_3_TR1		SCB6_CTS (0)		SCB6_SEL0 (0)			
P3.4	PWM0_M_1	PWM0_M_2_N	TC0_M_1_TR0	TC0_M_2_TR1				SCB6_SEL1 (0)			
P4.0	PWM0_4	PWM0_M_0_N	TC0_4_TR0	TC0_M_0_TR1	EXT_MUX[0]_0	SCB5_RX (0)		SCB5_MISO (0)		TRIG_IN[10]	
P4.1	PWM0_5	PWM0_4_N	TC0_5_TR0	TC0_4_TR1	EXT_MUX[0]_1	SCB5_TX (0)	SCB5_SDA (0)	SCB5_MOSI (0)		TRIG_IN[11]	
P5.0	PWM0_9	PWM0_8_N	TC0_9_TR0	TC0_8_TR1				SCB5_SEL2 (0)			
P5.1	PWM0_10	PWM0_9_N	TC0_10_TR0	TC0_9_TR1							
P5.2	PWM0_11	PWM0_10_N	TC0_11_TR0	TC0_10_TR1							
P5.3	PWM0_12	PWM0_11_N	TC0_12_TR0	TC0_11_TR1							
P5.4	PWM0_13	PWM0_12_N	TC0_13_TR0	TC0_12_TR1							
P6.0	PWM0_M_0	PWM0_14_N	TC0_M_0_TR0	TC0_14_TR1		SCB4_RX (0)		SCB4_MISO (0)			
P6.1	PWM0_0	PWM0_M_0_N	TC0_0_TR0	TC0_M_0_TR1		SCB4_TX (0)	SCB4_SDA (0)	SCB4_MOSI (0)			
P6.2	PWM0_M_1	PWM0_0_N	TC0_M_1_TR0	TC0_0_TR1		SCB4_RTS (0)	SCB4_SCL (0)	SCB4_CLK (0)	CAN0_2_TX		
P6.3	PWM0_1	PWM0_M_1_N	TC0_1_TR0	TC0_M_1_TR1		SCB4_CTS (0)		SCB4_SEL0 (0)	CAN0_2_RX		CAL_SUP_NZ
P6.4	PWM0_M_2	PWM0_1_N	TC0_M_2_TR0	TC0_1_TR1				SCB4_SEL1 (0)			
P6.5	PWM0_2	PWM0_M_2_N	TC0_2_TR0	TC0_M_2_TR1				SCB4_SEL2 (0)			

(table continues...)

20 High-Speed I/O matrix connection (HCon) reference as per Table 13.

21 Active Mode ordering (ACT#0, ACT#1, and so on) does not have any impact on choosing any alternate functions; HSIOM module will handle the individual alternate function assignment.

Table 16 (continued) Alternate pin functions in active mode

Name	Active mapping										
	HCon#8[20]	HCon#9	HCon#10	HCon#11	HCon#16	HCon#17	HCon#18	HCon#19	HCon#21	HCon#26	HCon#27
	ACT#0[21]	ACT#1	ACT#2	ACT#3	ACT#4	ACT#5	ACT#6	ACT#7	ACT#9	ACT#14	ACT#15
P6.6	PWM0_M_3	PWM0_2_N	TC0_M_3_TR0	TC0_2_TR1				SCB4_SEL3 (0)		TRIG_IN[8]	
P6.7	PWM0_3	PWM0_M_3_N	TC0_3_TR0	TC0_M_3_TR1						TRIG_IN[9]	
P7.0	PWM0_M_4	PWM0_3_N	TC0_M_4_TR0	TC0_3_TR1		SCB5_RX (1)		SCB5_MISO (1)			
P7.1	PWM0_15	PWM0_M_4_N	TC0_15_TR0	TC0_M_4_TR1		SCB5_TX (1)	SCB5_SDA (1)	SCB5_MOSI (1)			
P7.2	PWM0_M_5	PWM0_15_N	TC0_M_5_TR0	TC0_15_TR1		SCB5_RTS (1)	SCB5_SCL (1)	SCB5_CLK (1)			
P7.3	PWM0_16	PWM0_M_5_N	TC0_16_TR0	TC0_M_5_TR1		SCB5_CTS (1)		SCB5_SEL0 (1)			
P7.4	PWM0_M_6	PWM0_16_N	TC0_M_6_TR0	TC0_16_TR1				SCB5_SEL1 (1)			
P7.5	PWM0_17	PWM0_M_6_N	TC0_17_TR0	TC0_M_6_TR1				SCB5_SEL2 (1)			
P7.6	PWM0_M_7	PWM0_17_N	TC0_M_7_TR0	TC0_17_TR1						TRIG_IN[16]	
P7.7	PWM0_18	PWM0_M_7_N	TC0_18_TR0	TC0_M_7_TR1						TRIG_IN[17]	
P8.0	PWM0_19	PWM0_18_N	TC0_19_TR0	TC0_18_TR1					CAN0_0_TX		
P8.1	PWM0_20	PWM0_19_N	TC0_20_TR0	TC0_19_TR1					CAN0_0_RX	TRIG_IN[14]	
P8.2	PWM0_21	PWM0_20_N	TC0_21_TR0	TC0_20_TR1						TRIG_IN[15]	
P8.3	PWM0_22	PWM0_21_N	TC0_22_TR0	TC0_21_TR1							TRIG_DBG[0]
P9.0	PWM0_24	PWM0_23_N	TC0_24_TR0	TC0_23_TR1							
P9.1	PWM0_25	PWM0_24_N	TC0_25_TR0	TC0_24_TR1							

(table continues...)

20 High-Speed I/O matrix connection (HCon) reference as per Table 13.

21 Active Mode ordering (ACT#0, ACT#1, and so on) does not have any impact on choosing any alternate functions; HSIOM module will handle the individual alternate function assignment.

Table 16 (continued) Alternate pin functions in active mode

Name	Active mapping										
	HCon#8[20]	HCon#9	HCon#10	HCon#11	HCon#16	HCon#17	HCon#18	HCon#19	HCon#21	HCon#26	HCon#27
	ACT#0[21]	ACT#1	ACT#2	ACT#3	ACT#4	ACT#5	ACT#6	ACT#7	ACT#9	ACT#14	ACT#15
P10.0	PWM0_28	PWM0_27_N	TC0_28_TR0	TC0_27_TR1		SCB4_RX (1)		SCB4_MISO (1)		TRIG_IN[18]	
P10.1	PWM0_29	PWM0_28_N	TC0_29_TR0	TC0_28_TR1		SCB4_TX (1)	SCB4_SDA (1)	SCB4_MOSI (1)		TRIG_IN[19]	
P10.2	PWM0_30	PWM0_29_N	TC0_30_TR0	TC0_29_TR1		SCB4_RTS (1)	SCB4_SCL (1)	SCB4_CLK (1)			
P10.3	PWM0_31	PWM0_30_N	TC0_31_TR0	TC0_30_TR1		SCB4_CTS (1)		SCB4_SEL0 (1)			
P10.4	PWM0_32	PWM0_31_N	TC0_32_TR0	TC0_31_TR1				SCB4_SEL1 (1)			
P11.0											
P11.1											
P11.2											
P12.0	PWM0_36	PWM0_35_N	TC0_36_TR0	TC0_35_TR1					CAN0_2_TX	TRIG_IN[20]	
P12.1	PWM0_37	PWM0_36_N	TC0_37_TR0	TC0_36_TR1					CAN0_2_RX	TRIG_IN[21]	
P12.2	PWM0_38	PWM0_37_N	TC0_38_TR0	TC0_37_TR1	EXT_MUX[1]_EN						
P12.3	PWM0_39	PWM0_38_N	TC0_39_TR0	TC0_38_TR1	EXT_MUX[1]_0						
P12.4	PWM0_40	PWM0_39_N	TC0_40_TR0	TC0_39_TR1	EXT_MUX[1]_1						
P12.5	PWM0_41	PWM0_40_N	TC0_41_TR0	TC0_40_TR1	EXT_MUX[1]_2						
P13.0	PWM0_M_8	PWM0_43_N	TC0_M_8_TR0	TC0_43_TR1	EXT_MUX[2]_0	SCB3_RX (0)		SCB3_MISO (0)			
P13.1	PWM0_44	PWM0_M_8_N	TC0_44_TR0	TC0_M_8_TR1	EXT_MUX[2]_1	SCB3_TX (0)	SCB3_SDA (0)	SCB3_MOSI (0)			

(table continues...)

20 High-Speed I/O matrix connection (HCon) reference as per Table 13.

21 Active Mode ordering (ACT#0, ACT#1, and so on) does not have any impact on choosing any alternate functions; HSIOM module will handle the individual alternate function assignment.

Table 16 (continued) Alternate pin functions in active mode

Name	Active mapping										
	HCon#8[20]	HCon#9	HCon#10	HCon#11	HCon#16	HCon#17	HCon#18	HCon#19	HCon#21	HCon#26	HCon#27
	ACT#0[21]	ACT#1	ACT#2	ACT#3	ACT#4	ACT#5	ACT#6	ACT#7	ACT#9	ACT#14	ACT#15
P13.2	PWM0_M_9	PWM0_44_N	TC0_M_9_TR0	TC0_44_TR1	EXT_MUX[2]_2	SCB3_RTS (0)	SCB3_SCL (0)	SCB3_CLK (0)			
P13.3	PWM0_45	PWM0_M_9_N	TC0_45_TR0	TC0_M_9_TR1	EXT_MUX[2]_EN	SCB3_CTS (0)		SCB3_SEL0 (0)			
P13.4	PWM0_M_10	PWM0_45_N	TC0_M_10_TR0	TC0_45_TR1				SCB3_SEL1 (0)			
P13.5	PWM0_46	PWM0_M_10_N	TC0_46_TR0	TC0_M_10_TR1				SCB3_SEL2 (0)			
P13.6	PWM0_M_11	PWM0_46_N	TC0_M_11_TR0	TC0_46_TR1				SCB3_SEL3 (0)		TRIG_IN[22]	
P13.7	PWM0_47	PWM0_M_11_N	TC0_47_TR0	TC0_M_11_TR1						TRIG_IN[23]	
P14.0	PWM0_48	PWM0_47_N	TC0_48_TR0	TC0_47_TR1		SCB2_RX (0)		SCB2_MISO (0)	CAN1_0_TX		
P14.1	PWM0_49	PWM0_48_N	TC0_49_TR0	TC0_48_TR1		SCB2_TX (0)	SCB2_SDA (0)	SCB2_MOSI (0)	CAN1_0_RX		
P14.2	PWM0_50	PWM0_49_N	TC0_50_TR0	TC0_49_TR1		SCB2_RTS (0)	SCB2_SCL (0)	SCB2_CLK (0)			
P14.3	PWM0_51	PWM0_50_N	TC0_51_TR0	TC0_50_TR1		SCB2_CTS (0)		SCB2_SEL0 (0)			
P14.4	PWM0_52	PWM0_51_N	TC0_52_TR0	TC0_51_TR1				SCB2_SEL1 (0)			
P14.5	PWM0_53	PWM0_52_N	TC0_53_TR0	TC0_52_TR1				SCB2_SEL2 (0)			
P15.0	PWM0_56	PWM0_55_N	TC0_56_TR0	TC0_55_TR1					CAN1_3_TX		
P15.1	PWM0_57	PWM0_56_N	TC0_57_TR0	TC0_56_TR1					CAN1_3_RX		
P15.2	PWM0_58	PWM0_57_N	TC0_58_TR0	TC0_57_TR1							
P15.3	PWM0_59	PWM0_58_N	TC0_59_TR0	TC0_58_TR1							

(table continues...)

20 High-Speed I/O matrix connection (HCon) reference as per Table 13.

21 Active Mode ordering (ACT#0, ACT#1, and so on) does not have any impact on choosing any alternate functions; HSIOM module will handle the individual alternate function assignment.

Table 16 (continued) Alternate pin functions in active mode

Name	Active mapping										
	HCon#8[20]	HCon#9	HCon#10	HCon#11	HCon#16	HCon#17	HCon#18	HCon#19	HCon#21	HCon#26	HCon#27
	ACT#0[21]	ACT#1	ACT#2	ACT#3	ACT#4	ACT#5	ACT#6	ACT#7	ACT#9	ACT#14	ACT#15
P16.0	PWM0_60	PWM0_59_N	TC0_60_TR0	TC0_59_TR1	PWM0_H_0						
P16.1	PWM0_61	PWM0_60_N	TC0_61_TR0	TC0_60_TR1	PWM0_H_0_N						
P16.2	PWM0_62	PWM0_61_N	TC0_62_TR0	TC0_61_TR1	PWM0_H_1						
P17.0	PWM0_61	PWM0_62_N	TC0_61_TR0	TC0_62_TR1					CAN1_1_TX		
P17.1	PWM0_60	PWM0_61_N	TC0_60_TR0	TC0_61_TR1	PWM0_H_2	SCB3_RX (1)		SCB3_MISO (1)	CAN1_1_RX		
P17.2	PWM0_59	PWM0_60_N	TC0_59_TR0	TC0_60_TR1	PWM0_H_2_N	SCB3_TX (1)	SCB3_SDA (1)	SCB3_MOSI (1)			
P17.3	PWM0_58	PWM0_59_N	TC0_58_TR0	TC0_59_TR1	PWM0_H_3	SCB3_RTS (1)	SCB3_SCL (1)	SCB3_CLK (1)		TRIG_IN[26]	
P17.4	PWM0_57	PWM0_58_N	TC0_57_TR0	TC0_58_TR1	PWM0_H_3_N	SCB3_CTS (1)		SCB3_SEL0 (1)		TRIG_IN[27]	
P18.0	PWM0_M_6	PWM0_M_5_N	TC0_M_6_TR0	TC0_M_5_TR1	PWM0_H_0	SCB1_RX (0)		SCB1_MISO (0)			FAULT_OUT_0
P18.1	PWM0_M_7	PWM0_M_6_N	TC0_M_7_TR0	TC0_M_6_TR1	PWM0_H_0_N	SCB1_TX (0)	SCB1_SDA (0)	SCB1_MOSI (0)			FAULT_OUT_1
P18.2	PWM0_55	PWM0_M_7_N	TC0_55_TR0	TC0_M_7_TR1	PWM0_H_1	SCB1_RTS (0)	SCB1_SCL (0)	SCB1_CLK (0)			
P18.3	PWM0_54	PWM0_55_N	TC0_54_TR0	TC0_55_TR1	PWM0_H_1_N	SCB1_CTS (0)		SCB1_SEL0 (0)			TRACE_CLOCK (0)
P18.4	PWM0_53	PWM0_54_N	TC0_53_TR0	TC0_54_TR1	PWM0_H_2			SCB1_SEL1 (0)			TRACE_DATA_0 (0)
P18.5	PWM0_52	PWM0_53_N	TC0_52_TR0	TC0_53_TR1	PWM0_H_2_N			SCB1_SEL2 (0)			TRACE_DATA_1 (0)
P18.6	PWM0_51	PWM0_52_N	TC0_51_TR0	TC0_52_TR1	PWM0_H_3			SCB1_SEL3 (0)	CAN1_2_TX		TRACE_DATA_2 (0)
P18.7	PWM0_50	PWM0_51_N	TC0_50_TR0	TC0_51_TR1	PWM0_H_3_N				CAN1_2_RX		TRACE_DATA_3 (0)

(table continues...)

20 High-Speed I/O matrix connection (HCon) reference as per Table 13.

21 Active Mode ordering (ACT#0, ACT#1, and so on) does not have any impact on choosing any alternate functions; HSIOM module will handle the individual alternate function assignment.

Table 16 (continued) Alternate pin functions in active mode

Name	Active mapping										
	HCon#8[20]	HCon#9	HCon#10	HCon#11	HCon#16	HCon#17	HCon#18	HCon#19	HCon#21	HCon#26	HCon#27
	ACT#0[21]	ACT#1	ACT#2	ACT#3	ACT#4	ACT#5	ACT#6	ACT#7	ACT#9	ACT#14	ACT#15
P19.0	PWM0_M_3	PWM0_50_N	TC0_M_3_TR0	TC0_50_TR1	TC0_H_0_TR0	SCB2_RX (1)		SCB2_MISO (1)	CAN1_3_TX		FAULT_OUT_2
P19.1	PWM0_26	PWM0_M_3_N	TC0_26_TR0	TC0_M_3_TR1	TC0_H_0_TR1	SCB2_TX (1)	SCB2_SDA (1)	SCB2_MOSI (1)	CAN1_3_RX		FAULT_OUT_3
P19.2	PWM0_27	PWM0_26_N	TC0_27_TR0	TC0_26_TR1	TC0_H_1_TR0	SCB2_RTS (1)	SCB2_SCL (1)	SCB2_CLK (1)		TRIG_IN[28]	
P19.3	PWM0_28	PWM0_27_N	TC0_28_TR0	TC0_27_TR1	TC0_H_1_TR1	SCB2_CTS (1)		SCB2_SEL0 (1)		TRIG_IN[29]	
P19.4	PWM0_29	PWM0_28_N	TC0_29_TR0	TC0_28_TR1	TC0_H_2_TR0			SCB2_SEL1 (1)			
P20.0	PWM0_30	PWM0_29_N	TC0_30_TR0	TC0_29_TR1	TC0_H_2_TR1			SCB2_SEL2 (1)			
P20.1	PWM0_49	PWM0_30_N	TC0_49_TR0	TC0_30_TR1	TC0_H_3_TR0						
P20.2	PWM0_48	PWM0_49_N	TC0_48_TR0	TC0_49_TR1	TC0_H_3_TR1						
P20.3	PWM0_47	PWM0_48_N	TC0_47_TR0	TC0_48_TR1		SCB1_RX (1)		SCB1_MISO (1)	CAN1_2_TX		
P21.0	PWM0_42	PWM0_43_N	TC0_42_TR0	TC0_43_TR1				SCB1_SEL2 (1)			
P21.1	PWM0_41	PWM0_42_N	TC0_41_TR0	TC0_42_TR1							
P21.2	PWM0_40	PWM0_41_N	TC0_40_TR0	TC0_41_TR1						EXT_CLK	TRIG_DBG[1]
P21.3	PWM0_39	PWM0_40_N	TC0_39_TR0	TC0_40_TR1							
P21.5	PWM0_37	PWM0_38_N	TC0_37_TR0	TC0_38_TR1							
P21.6	PWM0_36	PWM0_37_N	TC0_36_TR0	TC0_37_TR1							
P22.0	PWM0_34	PWM0_35_N	TC0_34_TR0	TC0_35_TR1		SCB6_RX (1)		SCB6_MISO (1)	CAN1_1_TX		TRACE_DATA_0 (1)

(table continues...)

20 High-Speed I/O matrix connection (HCon) reference as per Table 13.

21 Active Mode ordering (ACT#0, ACT#1, and so on) does not have any impact on choosing any alternate functions; HSIOM module will handle the individual alternate function assignment.

Table 16 (continued) Alternate pin functions in active mode

Name	Active mapping										
	HCon#8[20]	HCon#9	HCon#10	HCon#11	HCon#16	HCon#17	HCon#18	HCon#19	HCon#21	HCon#26	HCon#27
	ACT#0[21]	ACT#1	ACT#2	ACT#3	ACT#4	ACT#5	ACT#6	ACT#7	ACT#9	ACT#14	ACT#15
P22.1	PWM0_33	PWM0_34_N	TC0_33_TR0	TC0_34_TR1		SCB6_TX (1)	SCB6_SDA (1)	SCB6_MOSI (1)	CAN1_1_RX		TRACE_DATA_1 (1)
P22.2	PWM0_32	PWM0_33_N	TC0_32_TR0	TC0_33_TR1		SCB6_RTS (1)	SCB6_SCL (1)	SCB6_CLK (1)			TRACE_DATA_2 (1)
P22.3	PWM0_31	PWM0_32_N	TC0_31_TR0	TC0_32_TR1		SCB6_CTS (1)		SCB6_SEL0 (1)			TRACE_DATA_3 (1)
P22.4	PWM0_30	PWM0_31_N	TC0_30_TR0	TC0_31_TR1				SCB6_SEL1 (1)			TRACE_CLOCK (1)
P22.5	PWM0_29	PWM0_30_N	TC0_29_TR0	TC0_30_TR1				SCB6_SEL2 (1)			
P22.6	PWM0_28	PWM0_29_N	TC0_28_TR0	TC0_29_TR1							
P23.0	PWM0_M_8	PWM0_27_N	TC0_M_8_TR0	TC0_27_TR1		SCB7_RX (1)		SCB7_MISO (1)	CAN1_0_TX		FAULT_OUT_0
P23.1	PWM0_M_9	PWM0_M_8_N	TC0_M_9_TR0	TC0_M_8_TR1		SCB7_TX (1)	SCB7_SDA (1)	SCB7_MOSI (1)	CAN1_0_RX		FAULT_OUT_1
P23.3	PWM0_M_11	PWM0_M_10_N	TC0_M_11_TR0	TC0_M_10_TR1		SCB7_CTS (1)		SCB7_SEL0 (1)		TRIG_IN[30]	FAULT_OUT_3
P23.4	PWM0_25	PWM0_M_11_N	TC0_25_TR0	TC0_M_11_TR1				SCB7_SEL1 (1)		TRIG_IN[31]	TRIG_DBG[0]
P23.5	PWM0_24	PWM0_25_N	TC0_24_TR0	TC0_25_TR1				SCB7_SEL2 (1)			
P23.6	PWM0_23	PWM0_24_N	TC0_23_TR0	TC0_24_TR1							
P23.7	PWM0_22	PWM0_23_N	TC0_22_TR0	TC0_23_TR1						EXT_CLK	CAL_SUP_NZ

Table 17 Pin mux descriptions

Sl. No.	Pin	Module	Description
1	PWMx_y	TCPWM	TCPWM 16-bit PWM (no motor control), PWM_DT and PWM_PR line out, x-TCPWM block, y-counter number

(table continues...)

20 High-Speed I/O matrix connection (HCon) reference as per Table 13.

21 Active Mode ordering (ACT#0, ACT#1, and so on) does not have any impact on choosing any alternate functions; HSIOM module will handle the individual alternate function assignment.

Table 17 (continued) Pin mux descriptions

Sl. No.	Pin	Module	Description
2	PWMx_y_N	TCPWM	TCPWM 16-bit PWM (no motor control), PWM_DT and PWM_PR complementary line out (N), x-TCPWM block, y-counter number
3	PWMx_M_y	TCPWM	TCPWM 16-bit PWM with motor control line out, x-TCPWM block, y-counter number
4	PWMx_M_y_N	TCPWM	TCPWM 16-bit PWM with motor control complementary line out (N), x-TCPWM block, y-counter number
5	PWMx_H_y	TCPWM	TCPWM 32-bit PWM, PWM_DT and PWM_PR line out, x-TCPWM block, y-counter number
6	PWMx_H_y_N	TCPWM	TCPWM 32-bit PWM, PWM_DT and PWM_PR complementary line out (N), x-TCPWM block, y-counter number
7	TCx_y_TRz	TCPWM	TCPWM 16-bit dedicated counter input triggers, x-TCPWM block, y-counter number, z-trigger number
8	TCx_M_y_TRz	TCPWM	TCPWM 16-bit dedicated counter input triggers with motor control, x-TCPWM block, y-counter number, z-trigger number
9	TCx_H_y_TRz	TCPWM	TCPWM 32-bit dedicated counter input triggers, x-TCPWM block, y-counter number, z-trigger number
10	SCBx_RX	SCB	UART Receive, x-SCB block
11	SCBx_TX	SCB	UART Transmit, x-SCB block
12	SCBx_RTS	SCB	UART Request to Send (Handshake), x-SCB block
13	SCBx_CTS	SCB	UART Clear to Send (Handshake), x-SCB block
14	SCBx_SDA	SCB	I2C Data line, x-SCB block
15	SCBx_SCL	SCB	I2C Clock line, x-SCB block
16	SCBx_MISO	SCB	SPI Master Input Slave Output, x-SCB block
17	SCBx_MOSI	SCB	SPI Master Output Slave Input, x-SCB block
18	SCBx_CLK	SCB	SPI Serial Clock, x-SCB block
19	SCBx_SELy	SCB	SPI Slave Select, x-SCB block, y-select line
20	CANx_y_TX	CAN FD	CAN Transmit line, x-CAN block, y-channel number
21	CANx_y_RX	CAN FD	CAN Receive line, x-CAN block, y-channel number
22	CAL_SUP_NZ	CAN FD	ETAS Calibration support line

(table continues...)

Table 17 (continued) Pin mux descriptions

Sl. No.	Pin	Module	Description
23	FAULT_OUT_x	SRSS	Fault output line x-0 to 3
24	TRACE_DATA_x	SRSS	Trace data out line x-0 to 3
25	TRACE_CLOCK	SRSS	Trace clock line
26	SWJ_TRSTN	SRSS	JTAG Test reset line (Active low
27	SWJ_SWO_TDO	SRSS	JTAG Test data output/SWO (Serial Wire Output)
28	SWJ_SWCLK_TCLK	SRSS	JTAG Test clock/SWD clock (Serial Wire Clock)
29	SWJ_SWDIO_TMS	SRSS	JTAG Test mode select/SWD data (Serial Wire Data Input/Output)
30	SWJ_SWDOE_TDI	SRSS	JTAG Test data input
31	HIBERNATE_WAKEUP[x]	SRSS	Hibernate wakeup line x-0 to 1
32	ADC[x]_y	PASS SAR	SAR, channel, x-SAR number, y-channel number
33	ADC[x]_M	PASS SAR	SAR motor control input, x-SAR number
34	EXT_MUX[x]_y	PASS SAR	External SAR MUX inputs, x-MUX number, y-MUX input 0 to 2
35	EXT_MUX[x]_EN	PASS SAR	External SAR MUX enable line
36	EXT_CLK	SRSS	External clock input or output
37	TRIG_IN[x]	HSIOM	HSIOM_IO_INPUT[x] of trigger inputs, x-0 to 47
38	TRIG_DBG[x]	HSIOM	HSIOM_IO_OUTPUT[x] of trigger outputs, x-0 to 1
39	WCO_IN	SRSS	Watch crystal oscillator input
40	WCO_OUT	SRSS	Watch crystal oscillator output
41	ECO_IN	SRSS	External crystal oscillator input
42	ECO_OUT	SRSS	External crystal oscillator output

14 Interrupts and wake-up assignments

Table 18 Peripheral interrupt assignments and wake-up sources

Interrupt	Source	Power mode	Description
0	cpuss_interrupts_ipc_0_IRQn	DeepSleep	CPUSS inter process communication interrupt #0
1	cpuss_interrupts_ipc_1_IRQn	DeepSleep	CPUSS inter process communication interrupt #1
2	cpuss_interrupts_ipc_2_IRQn	DeepSleep	CPUSS inter process communication interrupt #2
3	cpuss_interrupts_ipc_3_IRQn	DeepSleep	CPUSS inter process communication interrupt #3
4	cpuss_interrupts_ipc_4_IRQn	DeepSleep	CPUSS inter process communication interrupt #4
5	cpuss_interrupts_ipc_5_IRQn	DeepSleep	CPUSS inter process communication interrupt #5
6	cpuss_interrupts_ipc_6_IRQn	DeepSleep	CPUSS inter process communication interrupt #6
7	cpuss_interrupts_ipc_7_IRQn	DeepSleep	CPUSS inter process communication interrupt #7
8	cpuss_interrupts_fault_0_IRQn	DeepSleep	CPUSS fault structure #0 interrupt
9	cpuss_interrupts_fault_1_IRQn	DeepSleep	CPUSS fault structure #1 interrupt
10	cpuss_interrupts_fault_2_IRQn	DeepSleep	CPUSS fault structure #2 interrupt
11	cpuss_interrupts_fault_3_IRQn	DeepSleep	CPUSS fault structure #3 interrupt
12	srss_interrupt_backup_IRQn	DeepSleep	BACKUP domain Interrupt
13	srss_interrupt_mcwdt_0_IRQn	DeepSleep	Multi-counter watchdog timer #0 interrupt
14	srss_interrupt_mcwdt_1_IRQn	DeepSleep	Multi-counter watchdog timer #1 interrupt
15	srss_interrupt_wdt_IRQn	DeepSleep	Hardware watchdog timer interrupt
16	srss_interrupt_IRQn	DeepSleep	Other combined interrupts for SRSS (LVD, CLK_CAL)
17	scb_0_interrupt_IRQn	DeepSleep	Serial communication block #0 (DeepSleep capable)
18	evtgen_0_interrupt_dpslp_IRQn	DeepSleep	Event gen DeepSleep domain interrupt
19	ioss_interrupt_vdd_IRQn	DeepSleep	I/O Supply (V_{DDIO}, V_{DDA}, V_{DDD}) state change Interrupt
20	ioss_interrupt_gpio_IRQn	DeepSleep	Consolidated interrupt for GPIO_STD and GPIO_ENH, All ports
21	ioss_interrupts_gpio_0_IRQn	DeepSleep	GPIO_ENH Port #0 interrupt
22	ioss_interrupts_gpio_1_IRQn	DeepSleep	GPIO_STD Port #1 interrupt
23	ioss_interrupts_gpio_2_IRQn	DeepSleep	GPIO_STD Port #2 interrupt
24	ioss_interrupts_gpio_3_IRQn	DeepSleep	GPIO_STD Port #3 interrupt
25	ioss_interrupts_gpio_4_IRQn	DeepSleep	GPIO_STD Port #4 interrupt
26	ioss_interrupts_gpio_5_IRQn	DeepSleep	GPIO_STD Port #5 interrupt
27	ioss_interrupts_gpio_6_IRQn	DeepSleep	GPIO_STD Port #6 interrupt
28	ioss_interrupts_gpio_7_IRQn	DeepSleep	GPIO_STD Port #7 interrupt

(table continues...)

Table 18 (continued) Peripheral interrupt assignments and wake-up sources

Interrupt	Source	Power mode	Description
29	ioss_interrupts_gpio_8_IRQn	DeepSleep	GPIO_STD Port #8 interrupt
30	ioss_interrupts_gpio_9_IRQn	DeepSleep	GPIO_STD Port #9 interrupt
31	ioss_interrupts_gpio_10_IRQn	DeepSleep	GPIO_STD Port #10 interrupt
32	ioss_interrupts_gpio_11_IRQn	DeepSleep	GPIO_STD Port #11 interrupt
33	ioss_interrupts_gpio_12_IRQn	DeepSleep	GPIO_STD Port #12 interrupt
34	ioss_interrupts_gpio_13_IRQn	DeepSleep	GPIO_STD Port #13 interrupt
35	ioss_interrupts_gpio_14_IRQn	DeepSleep	GPIO_STD Port #14 interrupt
36	ioss_interrupts_gpio_15_IRQn	DeepSleep	GPIO_STD Port #15 interrupt
37	ioss_interrupts_gpio_16_IRQn	DeepSleep	GPIO_STD Port #16 interrupt
38	ioss_interrupts_gpio_17_IRQn	DeepSleep	GPIO_STD Port #17 interrupt
39	ioss_interrupts_gpio_18_IRQn	DeepSleep	GPIO_STD Port #18 interrupt
40	ioss_interrupts_gpio_19_IRQn	DeepSleep	GPIO_STD Port #19 interrupt
41	ioss_interrupts_gpio_20_IRQn	DeepSleep	GPIO_STD Port #20 interrupt
42	ioss_interrupts_gpio_21_IRQn	DeepSleep	GPIO_STD Port #21 interrupt
43	ioss_interrupts_gpio_22_IRQn	DeepSleep	GPIO_STD Port #22 interrupt
44	ioss_interrupts_gpio_23_IRQn	DeepSleep	GPIO_STD Port #23 interrupt
45	cpuss_interrupt_crypto_IRQn	Active	Crypto accelerator interrupt
46	cpuss_interrupt_fm_IRQn	Active	Flash macro Interrupt
47	cpuss_interrupts_cm4_fp_IRQn	Active	CM4 floating point operation fault
48	cpuss_interrupts_cm0_cti_0_IRQn	Active	CM0+ CTI (Cross trigger interface) #0
49	cpuss_interrupts_cm0_cti_1_IRQn	Active	CM0+ CTI #1
50	cpuss_interrupts_cm4_cti_0_IRQn	Active	CM4 CTI #0
51	cpuss_interrupts_cm4_cti_1_IRQn	Active	CM4 CTI #1
52	evtgen_0_interrupt_IRQn	Active	Event Generator Active domain interrupt
53	canfd_0_interrupt0_IRQn	Active	CAN0, Consolidated interrupt #0 for all three channels
54	canfd_0_interrupt1_IRQn	Active	CAN0, Consolidated interrupt #1 for all three channels
55	canfd_1_interrupt0_IRQn	Active	CAN1, Consolidated interrupt #0 for all three channels

(table continues...)

Table 18 (continued) Peripheral interrupt assignments and wake-up sources

Interrupt	Source	Power mode	Description
56	canfd_1_interrupt1_IRQn	Active	CAN1, Consolidated interrupt #1 for all three channels
57	canfd_0_interrupts0_0_IRQn	Active	CAN0, Interrupt #0, Channel #0
58	canfd_0_interrupts0_1_IRQn	Active	CAN0, Interrupt #0, Channel #1
59	canfd_0_interrupts0_2_IRQn	Active	CAN0, Interrupt #0, Channel #2
60	canfd_0_interrupts1_0_IRQn	Active	CAN0, Interrupt #1, Channel #0
61	canfd_0_interrupts1_1_IRQn	Active	CAN0, Interrupt #1, Channel #1
62	canfd_0_interrupts1_2_IRQn	Active	CAN0, Interrupt #1, Channel #2
63	canfd_1_interrupts0_0_IRQn	Active	CAN1, Interrupt #0, Channel #0
64	canfd_1_interrupts0_1_IRQn	Active	CAN1, Interrupt #0, Channel #1
65	canfd_1_interrupts0_2_IRQn	Active	CAN1, Interrupt #0, Channel #2
66	canfd_1_interrupts1_0_IRQn	Active	CAN1, Interrupt #1, Channel #0
67	canfd_1_interrupts1_1_IRQn	Active	CAN1, Interrupt #1, Channel #1
68	canfd_1_interrupts1_2_IRQn	Active	CAN1, Interrupt #1, Channel #2
77	scb_1_interrupt_IRQn	Active	SCB1 Interrupt
78	scb_2_interrupt_IRQn	Active	SCB2 Interrupt
79	scb_3_interrupt_IRQn	Active	SCB3 Interrupt
80	scb_4_interrupt_IRQn	Active	SCB4 Interrupt
81	scb_5_interrupt_IRQn	Active	SCB5 Interrupt
82	scb_6_interrupt_IRQn	Active	SCB6 Interrupt
83	scb_7_interrupt_IRQn	Active	SCB7 Interrupt
84	pass_0_interrupts_sar_0_IRQn	Active	SAR0, Logical channel #0 interrupt
85	pass_0_interrupts_sar_1_IRQn	Active	SAR0, Logical channel #1 interrupt
86	pass_0_interrupts_sar_2_IRQn	Active	SAR0, Logical channel #2 interrupt
87	pass_0_interrupts_sar_3_IRQn	Active	SAR0, Logical channel #3 interrupt
88	pass_0_interrupts_sar_4_IRQn	Active	SAR0, Logical channel #4 interrupt
89	pass_0_interrupts_sar_5_IRQn	Active	SAR0, Logical channel #5 interrupt
90	pass_0_interrupts_sar_6_IRQn	Active	SAR0, Logical channel #6 interrupt
91	pass_0_interrupts_sar_7_IRQn	Active	SAR0, Logical channel #7 interrupt
92	pass_0_interrupts_sar_8_IRQn	Active	SAR0, Logical channel #8 interrupt
93	pass_0_interrupts_sar_9_IRQn	Active	SAR0, Logical channel #9 interrupt
94	pass_0_interrupts_sar_10_IRQn	Active	SAR0, Logical channel #10 interrupt
95	pass_0_interrupts_sar_11_IRQn	Active	SAR0, Logical channel #11 interrupt

(table continues...)

Table 18 (continued) Peripheral interrupt assignments and wake-up sources

Interrupt	Source	Power mode	Description
96	pass_0_interrupts_sar_12_IRQn	Active	SAR0, Logical channel #12 interrupt
97	pass_0_interrupts_sar_13_IRQn	Active	SAR0, Logical channel #13 interrupt
98	pass_0_interrupts_sar_14_IRQn	Active	SAR0, Logical channel #14 interrupt
99	pass_0_interrupts_sar_15_IRQn	Active	SAR0, Logical channel #15 interrupt
100	pass_0_interrupts_sar_16_IRQn	Active	SAR0, Logical channel #16 interrupt
101	pass_0_interrupts_sar_17_IRQn	Active	SAR0, Logical channel #17 interrupt
102	pass_0_interrupts_sar_18_IRQn	Active	SAR0, Logical channel #18 interrupt
103	pass_0_interrupts_sar_19_IRQn	Active	SAR0, Logical channel #19 interrupt
104	pass_0_interrupts_sar_20_IRQn	Active	SAR0, Logical channel #20 interrupt
105	pass_0_interrupts_sar_21_IRQn	Active	SAR0, Logical channel #21 interrupt
106	pass_0_interrupts_sar_22_IRQn	Active	SAR0, Logical channel #22 interrupt
107	pass_0_interrupts_sar_23_IRQn	Active	SAR0, Logical channel #23 interrupt
108	pass_0_interrupts_sar_32_IRQn	Active	SAR1, Logical channel #0 interrupt
109	pass_0_interrupts_sar_33_IRQn	Active	SAR1, Logical channel #1 interrupt
110	pass_0_interrupts_sar_34_IRQn	Active	SAR1, Logical channel #2 interrupt
111	pass_0_interrupts_sar_35_IRQn	Active	SAR1, Logical channel #3 interrupt
112	pass_0_interrupts_sar_36_IRQn	Active	SAR1, Logical channel #4 interrupt
113	pass_0_interrupts_sar_37_IRQn	Active	SAR1, Logical channel #5 interrupt
114	pass_0_interrupts_sar_38_IRQn	Active	SAR1, Logical channel #6 interrupt
115	pass_0_interrupts_sar_39_IRQn	Active	SAR1, Logical channel #7 interrupt
116	pass_0_interrupts_sar_40_IRQn	Active	SAR1, Logical channel #8 interrupt
117	pass_0_interrupts_sar_41_IRQn	Active	SAR1, Logical channel #9 interrupt
118	pass_0_interrupts_sar_42_IRQn	Active	SAR1, Logical channel #10 interrupt
119	pass_0_interrupts_sar_43_IRQn	Active	SAR1, Logical channel #11 interrupt
120	pass_0_interrupts_sar_44_IRQn	Active	SAR1, Logical channel #12 interrupt
121	pass_0_interrupts_sar_45_IRQn	Active	SAR1, Logical channel #13 interrupt
122	pass_0_interrupts_sar_46_IRQn	Active	SAR1, Logical channel #14 interrupt
123	pass_0_interrupts_sar_47_IRQn	Active	SAR1, Logical channel #15 interrupt
124	pass_0_interrupts_sar_48_IRQn	Active	SAR1, Logical channel #16 interrupt
125	pass_0_interrupts_sar_49_IRQn	Active	SAR1, Logical channel #17 interrupt
126	pass_0_interrupts_sar_50_IRQn	Active	SAR1, Logical channel #18 interrupt
127	pass_0_interrupts_sar_51_IRQn	Active	SAR1, Logical channel #19 interrupt
128	pass_0_interrupts_sar_52_IRQn	Active	SAR1, Logical channel #20 interrupt

(table continues...)

Table 18 (continued) Peripheral interrupt assignments and wake-up sources

Interrupt	Source	Power mode	Description
129	pass_0_interrupts_sar_53_IRQn	Active	SAR1, Logical channel #21 interrupt
130	pass_0_interrupts_sar_54_IRQn	Active	SAR1, Logical channel #22 interrupt
131	pass_0_interrupts_sar_55_IRQn	Active	SAR1, Logical channel #23 interrupt
132	pass_0_interrupts_sar_56_IRQn	Active	SAR1, Logical channel #24 interrupt
133	pass_0_interrupts_sar_57_IRQn	Active	SAR1, Logical channel #25 interrupt
134	pass_0_interrupts_sar_58_IRQn	Active	SAR1, Logical channel #26 interrupt
135	pass_0_interrupts_sar_59_IRQn	Active	SAR1, Logical channel #27 interrupt
136	pass_0_interrupts_sar_60_IRQn	Active	SAR1, Logical channel #28 interrupt
137	pass_0_interrupts_sar_61_IRQn	Active	SAR1, Logical channel #29 interrupt
138	pass_0_interrupts_sar_62_IRQn	Active	SAR1, Logical channel #30 interrupt
139	pass_0_interrupts_sar_63_IRQn	Active	SAR1, Logical channel #31 interrupt
140	pass_0_interrupts_sar_64_IRQn	Active	SAR2, Logical channel #0 interrupt
141	pass_0_interrupts_sar_65_IRQn	Active	SAR2, Logical channel #1 interrupt
142	pass_0_interrupts_sar_66_IRQn	Active	SAR2, Logical channel #2 interrupt
143	pass_0_interrupts_sar_67_IRQn	Active	SAR2, Logical channel #3 interrupt
144	pass_0_interrupts_sar_68_IRQn	Active	SAR2, Logical channel #4 interrupt
145	pass_0_interrupts_sar_69_IRQn	Active	SAR2, Logical channel #5 interrupt
146	pass_0_interrupts_sar_70_IRQn	Active	SAR2, Logical channel #6 interrupt
147	pass_0_interrupts_sar_71_IRQn	Active	SAR2, Logical channel #7 interrupt
148	cpuss_interrupts_dmac_0_IRQn	Active	CPUSS M-DMA0, Channel #0 Interrupt
149	cpuss_interrupts_dmac_1_IRQn	Active	CPUSS M-DMA0, Channel #1 Interrupt
150	cpuss_interrupts_dmac_2_IRQn	Active	CPUSS M-DMA0, Channel #2 Interrupt
151	cpuss_interrupts_dmac_3_IRQn	Active	CPUSS M-DMA0, Channel #3 Interrupt
152	cpuss_interrupts_dw0_0_IRQn	Active	CPUSS P-DMA0, Channel #0 Interrupt
153	cpuss_interrupts_dw0_1_IRQn	Active	CPUSS P-DMA0, Channel #1 Interrupt
154	cpuss_interrupts_dw0_2_IRQn	Active	CPUSS P-DMA0, Channel #2 Interrupt
155	cpuss_interrupts_dw0_3_IRQn	Active	CPUSS P-DMA0, Channel #3 Interrupt
156	cpuss_interrupts_dw0_4_IRQn	Active	CPUSS P-DMA0, Channel #4 Interrupt
157	cpuss_interrupts_dw0_5_IRQn	Active	CPUSS P-DMA0, Channel #5 Interrupt
158	cpuss_interrupts_dw0_6_IRQn	Active	CPUSS P-DMA0, Channel #6 Interrupt
159	cpuss_interrupts_dw0_7_IRQn	Active	CPUSS P-DMA0, Channel #7 Interrupt
160	cpuss_interrupts_dw0_8_IRQn	Active	CPUSS P-DMA0, Channel #8 Interrupt
161	cpuss_interrupts_dw0_9_IRQn	Active	CPUSS P-DMA0, Channel #9 Interrupt

(table continues...)

Table 18 (continued) Peripheral interrupt assignments and wake-up sources

Interrupt	Source	Power mode	Description
162	cpuss_interrupts_dw0_10_IRQn	Active	CPUSS P-DMA0, Channel #10 Interrupt
163	cpuss_interrupts_dw0_11_IRQn	Active	CPUSS P-DMA0, Channel #11 Interrupt
164	cpuss_interrupts_dw0_12_IRQn	Active	CPUSS P-DMA0, Channel #12 Interrupt
165	cpuss_interrupts_dw0_13_IRQn	Active	CPUSS P-DMA0, Channel #13 Interrupt
166	cpuss_interrupts_dw0_14_IRQn	Active	CPUSS P-DMA0, Channel #14 Interrupt
167	cpuss_interrupts_dw0_15_IRQn	Active	CPUSS P-DMA0, Channel #15 Interrupt
168	cpuss_interrupts_dw0_16_IRQn	Active	CPUSS P-DMA0, Channel #16 Interrupt
169	cpuss_interrupts_dw0_17_IRQn	Active	CPUSS P-DMA0, Channel #17 Interrupt
170	cpuss_interrupts_dw0_18_IRQn	Active	CPUSS P-DMA0, Channel #18 Interrupt
171	cpuss_interrupts_dw0_19_IRQn	Active	CPUSS P-DMA0, Channel #19 Interrupt
172	cpuss_interrupts_dw0_20_IRQn	Active	CPUSS P-DMA0, Channel #20 Interrupt
173	cpuss_interrupts_dw0_21_IRQn	Active	CPUSS P-DMA0, Channel #21 Interrupt
174	cpuss_interrupts_dw0_22_IRQn	Active	CPUSS P-DMA0, Channel #22 Interrupt
175	cpuss_interrupts_dw0_23_IRQn	Active	CPUSS P-DMA0, Channel #23 Interrupt
176	cpuss_interrupts_dw0_24_IRQn	Active	CPUSS P-DMA0, Channel #24 Interrupt
177	cpuss_interrupts_dw0_25_IRQn	Active	CPUSS P-DMA0, Channel #25 Interrupt
178	cpuss_interrupts_dw0_26_IRQn	Active	CPUSS P-DMA0, Channel #26 Interrupt
179	cpuss_interrupts_dw0_27_IRQn	Active	CPUSS P-DMA0, Channel #27 Interrupt
180	cpuss_interrupts_dw0_28_IRQn	Active	CPUSS P-DMA0, Channel #28 Interrupt
181	cpuss_interrupts_dw0_29_IRQn	Active	CPUSS P-DMA0, Channel #29 Interrupt
182	cpuss_interrupts_dw0_30_IRQn	Active	CPUSS P-DMA0, Channel #30 Interrupt
183	cpuss_interrupts_dw0_31_IRQn	Active	CPUSS P-DMA0, Channel #31 Interrupt
184	cpuss_interrupts_dw0_32_IRQn	Active	CPUSS P-DMA0, Channel #32 Interrupt
185	cpuss_interrupts_dw0_33_IRQn	Active	CPUSS P-DMA0, Channel #33 Interrupt
186	cpuss_interrupts_dw0_34_IRQn	Active	CPUSS P-DMA0, Channel #34 Interrupt
187	cpuss_interrupts_dw0_35_IRQn	Active	CPUSS P-DMA0, Channel #35 Interrupt
188	cpuss_interrupts_dw0_36_IRQn	Active	CPUSS P-DMA0, Channel #36 Interrupt
189	cpuss_interrupts_dw0_37_IRQn	Active	CPUSS P-DMA0, Channel #37 Interrupt
190	cpuss_interrupts_dw0_38_IRQn	Active	CPUSS P-DMA0, Channel #38 Interrupt
191	cpuss_interrupts_dw0_39_IRQn	Active	CPUSS P-DMA0, Channel #39 Interrupt
192	cpuss_interrupts_dw0_40_IRQn	Active	CPUSS P-DMA0, Channel #40 Interrupt
193	cpuss_interrupts_dw0_41_IRQn	Active	CPUSS P-DMA0, Channel #41 Interrupt
194	cpuss_interrupts_dw0_42_IRQn	Active	CPUSS P-DMA0, Channel #42 Interrupt

(table continues...)

Table 18 (continued) Peripheral interrupt assignments and wake-up sources

Interrupt	Source	Power mode	Description
195	cpuss_interrupts_dw0_43_IRQn	Active	CPUSS P-DMA0, Channel #43 Interrupt
196	cpuss_interrupts_dw0_44_IRQn	Active	CPUSS P-DMA0, Channel #44 Interrupt
197	cpuss_interrupts_dw0_45_IRQn	Active	CPUSS P-DMA0, Channel #45 Interrupt
198	cpuss_interrupts_dw0_46_IRQn	Active	CPUSS P-DMA0, Channel #46 Interrupt
199	cpuss_interrupts_dw0_47_IRQn	Active	CPUSS P-DMA0, Channel #47 Interrupt
200	cpuss_interrupts_dw0_48_IRQn	Active	CPUSS P-DMA0, Channel #48 Interrupt
201	cpuss_interrupts_dw0_49_IRQn	Active	CPUSS P-DMA0, Channel #49 Interrupt
202	cpuss_interrupts_dw0_50_IRQn	Active	CPUSS P-DMA0, Channel #50 Interrupt
203	cpuss_interrupts_dw0_51_IRQn	Active	CPUSS P-DMA0, Channel #51 Interrupt
204	cpuss_interrupts_dw0_52_IRQn	Active	CPUSS P-DMA0, Channel #52 Interrupt
205	cpuss_interrupts_dw0_53_IRQn	Active	CPUSS P-DMA0, Channel #53 Interrupt
206	cpuss_interrupts_dw0_54_IRQn	Active	CPUSS P-DMA0, Channel #54 Interrupt
207	cpuss_interrupts_dw0_55_IRQn	Active	CPUSS P-DMA0, Channel #55 Interrupt
208	cpuss_interrupts_dw0_56_IRQn	Active	CPUSS P-DMA0, Channel #56 Interrupt
209	cpuss_interrupts_dw0_57_IRQn	Active	CPUSS P-DMA0, Channel #57 Interrupt
210	cpuss_interrupts_dw0_58_IRQn	Active	CPUSS P-DMA0, Channel #58 Interrupt
211	cpuss_interrupts_dw0_59_IRQn	Active	CPUSS P-DMA0, Channel #59 Interrupt
212	cpuss_interrupts_dw0_60_IRQn	Active	CPUSS P-DMA0, Channel #60 Interrupt
213	cpuss_interrupts_dw0_61_IRQn	Active	CPUSS P-DMA0, Channel #61 Interrupt
214	cpuss_interrupts_dw0_62_IRQn	Active	CPUSS P-DMA0, Channel #62 Interrupt
215	cpuss_interrupts_dw0_63_IRQn	Active	CPUSS P-DMA0, Channel #63 Interrupt
216	cpuss_interrupts_dw0_64_IRQn	Active	CPUSS P-DMA0, Channel #64 Interrupt
217	cpuss_interrupts_dw0_65_IRQn	Active	CPUSS P-DMA0, Channel #65 Interrupt
218	cpuss_interrupts_dw0_66_IRQn	Active	CPUSS P-DMA0, Channel #66 Interrupt
219	cpuss_interrupts_dw0_67_IRQn	Active	CPUSS P-DMA0, Channel #67 Interrupt
220	cpuss_interrupts_dw0_68_IRQn	Active	CPUSS P-DMA0, Channel #68 Interrupt
221	cpuss_interrupts_dw0_69_IRQn	Active	CPUSS P-DMA0, Channel #69 Interrupt
222	cpuss_interrupts_dw0_70_IRQn	Active	CPUSS P-DMA0, Channel #70 Interrupt
223	cpuss_interrupts_dw0_71_IRQn	Active	CPUSS P-DMA0, Channel #71 Interrupt
224	cpuss_interrupts_dw0_72_IRQn	Active	CPUSS P-DMA0, Channel #72 Interrupt
225	cpuss_interrupts_dw0_73_IRQn	Active	CPUSS P-DMA0, Channel #73 Interrupt
226	cpuss_interrupts_dw0_74_IRQn	Active	CPUSS P-DMA0, Channel #74 Interrupt
227	cpuss_interrupts_dw0_75_IRQn	Active	CPUSS P-DMA0, Channel #75 Interrupt

(table continues...)

Table 18 (continued) Peripheral interrupt assignments and wake-up sources

Interrupt	Source	Power mode	Description
228	cpuss_interrupts_dw0_76_IRQn	Active	CPUSS P-DMA0, Channel #76 Interrupt
229	cpuss_interrupts_dw0_77_IRQn	Active	CPUSS P-DMA0, Channel #77 Interrupt
230	cpuss_interrupts_dw0_78_IRQn	Active	CPUSS P-DMA0, Channel #78 Interrupt
231	cpuss_interrupts_dw0_79_IRQn	Active	CPUSS P-DMA0, Channel #79 Interrupt
232	cpuss_interrupts_dw0_80_IRQn	Active	CPUSS P-DMA0, Channel #80 Interrupt
233	cpuss_interrupts_dw0_81_IRQn	Active	CPUSS P-DMA0, Channel #81 Interrupt
234	cpuss_interrupts_dw0_82_IRQn	Active	CPUSS P-DMA0, Channel #82 Interrupt
235	cpuss_interrupts_dw0_83_IRQn	Active	CPUSS P-DMA0, Channel #83 Interrupt
236	cpuss_interrupts_dw0_84_IRQn	Active	CPUSS P-DMA0, Channel #84 Interrupt
237	cpuss_interrupts_dw0_85_IRQn	Active	CPUSS P-DMA0, Channel #85 Interrupt
238	cpuss_interrupts_dw0_86_IRQn	Active	CPUSS P-DMA0, Channel #86 Interrupt
239	cpuss_interrupts_dw0_87_IRQn	Active	CPUSS P-DMA0, Channel #87 Interrupt
240	cpuss_interrupts_dw0_88_IRQn	Active	CPUSS P-DMA0, Channel #88 Interrupt
241	cpuss_interrupts_dw1_0_IRQn	Active	CPUSS P-DMA1, Channel #0 Interrupt
242	cpuss_interrupts_dw1_1_IRQn	Active	CPUSS P-DMA1, Channel #1 Interrupt
243	cpuss_interrupts_dw1_2_IRQn	Active	CPUSS P-DMA1, Channel #2 Interrupt
244	cpuss_interrupts_dw1_3_IRQn	Active	CPUSS P-DMA1, Channel #3 Interrupt
245	cpuss_interrupts_dw1_4_IRQn	Active	CPUSS P-DMA1, Channel #4 Interrupt
246	cpuss_interrupts_dw1_5_IRQn	Active	CPUSS P-DMA1, Channel #5 Interrupt
247	cpuss_interrupts_dw1_6_IRQn	Active	CPUSS P-DMA1, Channel #6 Interrupt
248	cpuss_interrupts_dw1_7_IRQn	Active	CPUSS P-DMA1, Channel #7 Interrupt
249	cpuss_interrupts_dw1_8_IRQn	Active	CPUSS P-DMA1, Channel #8 Interrupt
250	cpuss_interrupts_dw1_9_IRQn	Active	CPUSS P-DMA1, Channel #9 Interrupt
251	cpuss_interrupts_dw1_10_IRQn	Active	CPUSS P-DMA1, Channel #10 Interrupt
252	cpuss_interrupts_dw1_11_IRQn	Active	CPUSS P-DMA1, Channel #11 Interrupt
253	cpuss_interrupts_dw1_12_IRQn	Active	CPUSS P-DMA1, Channel #12 Interrupt
254	cpuss_interrupts_dw1_13_IRQn	Active	CPUSS P-DMA1, Channel #13 Interrupt
255	cpuss_interrupts_dw1_14_IRQn	Active	CPUSS P-DMA1, Channel #14 Interrupt
256	cpuss_interrupts_dw1_15_IRQn	Active	CPUSS P-DMA1, Channel #15 Interrupt
257	cpuss_interrupts_dw1_16_IRQn	Active	CPUSS P-DMA1, Channel #16 Interrupt
258	cpuss_interrupts_dw1_17_IRQn	Active	CPUSS P-DMA1, Channel #17 Interrupt
259	cpuss_interrupts_dw1_18_IRQn	Active	CPUSS P-DMA1, Channel #18 Interrupt
260	cpuss_interrupts_dw1_19_IRQn	Active	CPUSS P-DMA1, Channel #19 Interrupt

(table continues...)

Table 18 (continued) Peripheral interrupt assignments and wake-up sources

Interrupt	Source	Power mode	Description
261	cpuss_interrupts_dw1_20_IRQn	Active	CPUSS P-DMA1, Channel #20 Interrupt
262	cpuss_interrupts_dw1_21_IRQn	Active	CPUSS P-DMA1, Channel #21 Interrupt
263	cpuss_interrupts_dw1_22_IRQn	Active	CPUSS P-DMA1, Channel #22 Interrupt
264	cpuss_interrupts_dw1_23_IRQn	Active	CPUSS P-DMA1, Channel #23 Interrupt
265	cpuss_interrupts_dw1_24_IRQn	Active	CPUSS P-DMA1, Channel #24 Interrupt
266	cpuss_interrupts_dw1_25_IRQn	Active	CPUSS P-DMA1, Channel #25 Interrupt
267	cpuss_interrupts_dw1_26_IRQn	Active	CPUSS P-DMA1, Channel #26 Interrupt
268	cpuss_interrupts_dw1_27_IRQn	Active	CPUSS P-DMA1, Channel #27 Interrupt
269	cpuss_interrupts_dw1_28_IRQn	Active	CPUSS P-DMA1, Channel #28 Interrupt
270	cpuss_interrupts_dw1_29_IRQn	Active	CPUSS P-DMA1, Channel #29 Interrupt
271	cpuss_interrupts_dw1_30_IRQn	Active	CPUSS P-DMA1, Channel #30 Interrupt
272	cpuss_interrupts_dw1_31_IRQn	Active	CPUSS P-DMA1, Channel #31 Interrupt
273	cpuss_interrupts_dw1_32_IRQn	Active	CPUSS P-DMA1, Channel #32 Interrupt
274	tcpwm_0_interrupts_0_IRQn	Active	TCPWM0 Group #0, Counter #0 Interrupt
275	tcpwm_0_interrupts_1_IRQn	Active	TCPWM0 Group #0, Counter #1 Interrupt
276	tcpwm_0_interrupts_2_IRQn	Active	TCPWM0 Group #0, Counter #2 Interrupt
277	tcpwm_0_interrupts_3_IRQn	Active	TCPWM0 Group #0, Counter #3 Interrupt
278	tcpwm_0_interrupts_4_IRQn	Active	TCPWM0 Group #0, Counter #4 Interrupt
279	tcpwm_0_interrupts_5_IRQn	Active	TCPWM0 Group #0, Counter #5 Interrupt
280	tcpwm_0_interrupts_6_IRQn	Active	TCPWM0 Group #0, Counter #6 Interrupt
281	tcpwm_0_interrupts_7_IRQn	Active	TCPWM0 Group #0, Counter #7 Interrupt
282	tcpwm_0_interrupts_8_IRQn	Active	TCPWM0 Group #0, Counter #8 Interrupt
283	tcpwm_0_interrupts_9_IRQn	Active	TCPWM0 Group #0, Counter #9 Interrupt
284	tcpwm_0_interrupts_10_IRQn	Active	TCPWM0 Group #0, Counter #10 Interrupt
285	tcpwm_0_interrupts_11_IRQn	Active	TCPWM0 Group #0, Counter #11 Interrupt
286	tcpwm_0_interrupts_12_IRQn	Active	TCPWM0 Group #0, Counter #12 Interrupt
287	tcpwm_0_interrupts_13_IRQn	Active	TCPWM0 Group #0, Counter #13 Interrupt
288	tcpwm_0_interrupts_14_IRQn	Active	TCPWM0 Group #0, Counter #14 Interrupt
289	tcpwm_0_interrupts_15_IRQn	Active	TCPWM0 Group #0, Counter #15 Interrupt
290	tcpwm_0_interrupts_16_IRQn	Active	TCPWM0 Group #0, Counter #16 Interrupt
291	tcpwm_0_interrupts_17_IRQn	Active	TCPWM0 Group #0, Counter #17 Interrupt
292	tcpwm_0_interrupts_18_IRQn	Active	TCPWM0 Group #0, Counter #18 Interrupt
293	tcpwm_0_interrupts_19_IRQn	Active	TCPWM0 Group #0, Counter #19 Interrupt

(table continues...)

Table 18 (continued) Peripheral interrupt assignments and wake-up sources

Interrupt	Source	Power mode	Description
294	tcpwm_0_interrupts_20_IRQn	Active	TCPWM0 Group #0, Counter #20 Interrupt
295	tcpwm_0_interrupts_21_IRQn	Active	TCPWM0 Group #0, Counter #21 Interrupt
296	tcpwm_0_interrupts_22_IRQn	Active	TCPWM0 Group #0, Counter #22 Interrupt
297	tcpwm_0_interrupts_23_IRQn	Active	TCPWM0 Group #0, Counter #23 Interrupt
298	tcpwm_0_interrupts_24_IRQn	Active	TCPWM0 Group #0, Counter #24 Interrupt
299	tcpwm_0_interrupts_25_IRQn	Active	TCPWM0 Group #0, Counter #25 Interrupt
300	tcpwm_0_interrupts_26_IRQn	Active	TCPWM0 Group #0, Counter #26 Interrupt
301	tcpwm_0_interrupts_27_IRQn	Active	TCPWM0 Group #0, Counter #27 Interrupt
302	tcpwm_0_interrupts_28_IRQn	Active	TCPWM0 Group #0, Counter #28 Interrupt
303	tcpwm_0_interrupts_29_IRQn	Active	TCPWM0 Group #0, Counter #29 Interrupt
304	tcpwm_0_interrupts_30_IRQn	Active	TCPWM0 Group #0, Counter #30 Interrupt
305	tcpwm_0_interrupts_31_IRQn	Active	TCPWM0 Group #0, Counter #31 Interrupt
306	tcpwm_0_interrupts_32_IRQn	Active	TCPWM0 Group #0, Counter #32 Interrupt
307	tcpwm_0_interrupts_33_IRQn	Active	TCPWM0 Group #0, Counter #33 Interrupt
308	tcpwm_0_interrupts_34_IRQn	Active	TCPWM0 Group #0, Counter #34 Interrupt
309	tcpwm_0_interrupts_35_IRQn	Active	TCPWM0 Group #0, Counter #35 Interrupt
310	tcpwm_0_interrupts_36_IRQn	Active	TCPWM0 Group #0, Counter #36 Interrupt
311	tcpwm_0_interrupts_37_IRQn	Active	TCPWM0 Group #0, Counter #37 Interrupt
312	tcpwm_0_interrupts_38_IRQn	Active	TCPWM0 Group #0, Counter #38 Interrupt
313	tcpwm_0_interrupts_39_IRQn	Active	TCPWM0 Group #0, Counter #39 Interrupt
314	tcpwm_0_interrupts_40_IRQn	Active	TCPWM0 Group #0, Counter #40 Interrupt
315	tcpwm_0_interrupts_41_IRQn	Active	TCPWM0 Group #0, Counter #41 Interrupt
316	tcpwm_0_interrupts_42_IRQn	Active	TCPWM0 Group #0, Counter #42 Interrupt
317	tcpwm_0_interrupts_43_IRQn	Active	TCPWM0 Group #0, Counter #43 Interrupt
318	tcpwm_0_interrupts_44_IRQn	Active	TCPWM0 Group #0, Counter #44 Interrupt
319	tcpwm_0_interrupts_45_IRQn	Active	TCPWM0 Group #0, Counter #45 Interrupt
320	tcpwm_0_interrupts_46_IRQn	Active	TCPWM0 Group #0, Counter #46 Interrupt
321	tcpwm_0_interrupts_47_IRQn	Active	TCPWM0 Group #0, Counter #47 Interrupt
322	tcpwm_0_interrupts_48_IRQn	Active	TCPWM0 Group #0, Counter #48 Interrupt
323	tcpwm_0_interrupts_49_IRQn	Active	TCPWM0 Group #0, Counter #49 Interrupt
324	tcpwm_0_interrupts_50_IRQn	Active	TCPWM0 Group #0, Counter #50 Interrupt
325	tcpwm_0_interrupts_51_IRQn	Active	TCPWM0 Group #0, Counter #51 Interrupt
326	tcpwm_0_interrupts_52_IRQn	Active	TCPWM0 Group #0, Counter #52 Interrupt

(table continues...)

Table 18 (continued) Peripheral interrupt assignments and wake-up sources

Interrupt	Source	Power mode	Description
327	tcpwm_0_interrupts_53_IRQn	Active	TCPWM0 Group #0, Counter #53 Interrupt
328	tcpwm_0_interrupts_54_IRQn	Active	TCPWM0 Group #0, Counter #54 Interrupt
329	tcpwm_0_interrupts_55_IRQn	Active	TCPWM0 Group #0, Counter #55 Interrupt
330	tcpwm_0_interrupts_56_IRQn	Active	TCPWM0 Group #0, Counter #56 Interrupt
331	tcpwm_0_interrupts_57_IRQn	Active	TCPWM0 Group #0, Counter #57 Interrupt
332	tcpwm_0_interrupts_58_IRQn	Active	TCPWM0 Group #0, Counter #58 Interrupt
333	tcpwm_0_interrupts_59_IRQn	Active	TCPWM0 Group #0, Counter #59 Interrupt
334	tcpwm_0_interrupts_60_IRQn	Active	TCPWM0 Group #0, Counter #60 Interrupt
335	tcpwm_0_interrupts_61_IRQn	Active	TCPWM0 Group #0, Counter #61 Interrupt
336	tcpwm_0_interrupts_62_IRQn	Active	TCPWM0 Group #0, Counter #62 Interrupt
337	tcpwm_0_interrupts_256_IRQn	Active	TCPWM0 Group #1, Counter #0 Interrupt
338	tcpwm_0_interrupts_257_IRQn	Active	TCPWM0 Group #1, Counter #1 Interrupt
339	tcpwm_0_interrupts_258_IRQn	Active	TCPWM0 Group #1, Counter #2 Interrupt
340	tcpwm_0_interrupts_259_IRQn	Active	TCPWM0 Group #1, Counter #3 Interrupt
341	tcpwm_0_interrupts_260_IRQn	Active	TCPWM0 Group #1, Counter #4 Interrupt
342	tcpwm_0_interrupts_261_IRQn	Active	TCPWM0 Group #1, Counter #5 Interrupt
343	tcpwm_0_interrupts_262_IRQn	Active	TCPWM0 Group #1, Counter #6 Interrupt
344	tcpwm_0_interrupts_263_IRQn	Active	TCPWM0 Group #1, Counter #7 Interrupt
345	tcpwm_0_interrupts_264_IRQn	Active	TCPWM0 Group #1, Counter #8 Interrupt
346	tcpwm_0_interrupts_265_IRQn	Active	TCPWM0 Group #1, Counter #9 Interrupt
347	tcpwm_0_interrupts_266_IRQn	Active	TCPWM0 Group #1, Counter #10 Interrupt
348	tcpwm_0_interrupts_267_IRQn	Active	TCPWM0 Group #1, Counter #11 Interrupt
349	tcpwm_0_interrupts_512_IRQn	Active	TCPWM0 Group #2, Counter #0 Interrupt
350	tcpwm_0_interrupts_513_IRQn	Active	TCPWM0 Group #2, Counter #1 Interrupt
351	tcpwm_0_interrupts_514_IRQn	Active	TCPWM0 Group #2, Counter #2 Interrupt
352	tcpwm_0_interrupts_515_IRQn	Active	TCPWM0 Group #2, Counter #3 Interrupt

15 Core interrupt types

Table 19 Core interrupt types

Interrupt	Source	Power mode	Description
0	CPUIntIdx0_IRQn[22)]	DeepSleep	CPU user interrupt #0
1	CPUIntIdx1_IRQn[22)]	DeepSleep	CPU user interrupt #1
2	CPUIntIdx2_IRQn	DeepSleep	CPU user interrupt #2
3	CPUIntIdx3_IRQn	DeepSleep	CPU user interrupt #3
4	CPUIntIdx4_IRQn	DeepSleep	CPU user interrupt #4
5	CPUIntIdx5_IRQn	DeepSleep	CPU user interrupt #5
6	CPUIntIdx6_IRQn	DeepSleep	CPU user interrupt #6
7	CPUIntIdx7_IRQn	DeepSleep	CPU user interrupt #7
8	Internal0_IRQn	Active	Internal software interrupt #0
9	Internal1_IRQn	Active	Internal software interrupt #1
10	Internal2_IRQn	Active	Internal software interrupt #2
11	Internal3_IRQn	Active	Internal software interrupt #3
12	Internal4_IRQn	Active	Internal software interrupt #4
13	Internal5_IRQn	Active	Internal software interrupt #5
14	Internal6_IRQn	Active	Internal software interrupt #6
15	Internal7_IRQn	Active	Internal software interrupt #7

22 User interrupt cannot be used for CM0+ application, as it is used internally by system calls. Note, this does not impact CM4 application.

16 Trigger multiplexer

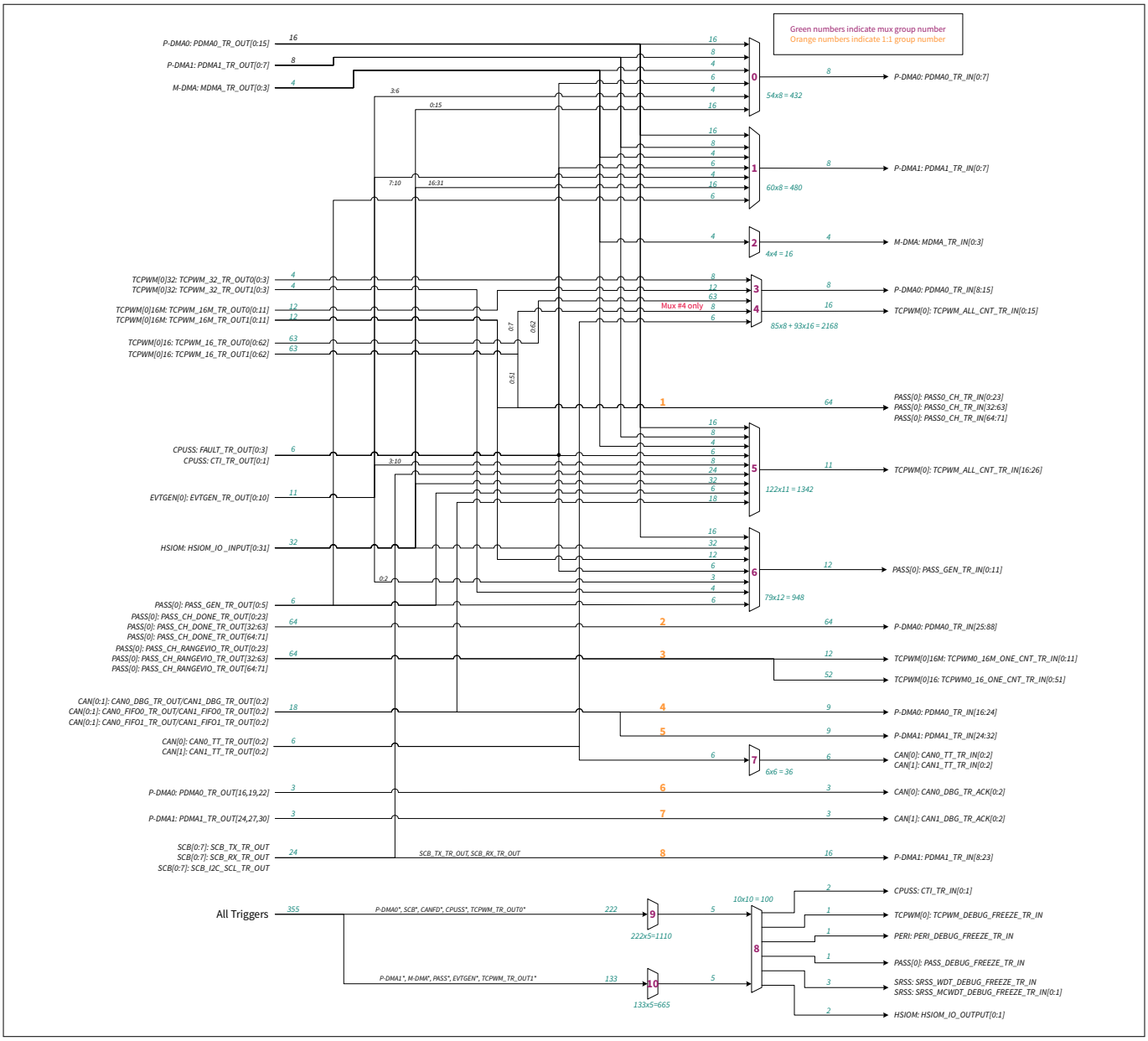


Figure 10 Trigger multiplexer

***Note**: The diagram shows only the TRIG_LABEL, final trigger formation based on the formula TRIG_{PREFIX(IN/OUT)}_{MUX_x}_{TRIG_LABEL} / TRIG_{PREFIX(IN_1TO1/OUT_1TO1)}_{x}_{TRIG_LABEL} (see Table 20, Table 21, and Table 22.)*

17 Triggers group inputs

Table 20 Trigger inputs

Input	Trigger	Description
MUX Group 0: PDMA0_TR (P-DMA0 trigger multiplexer)		
1:16[23]	PDMA0_TR_OUT[0:15]	Allow P-DMA0 to chain to itself, useful for triggering once per row for 2D transfer
17:24	PDMA1_TR_OUT[0:7]	Cross connections from P-DMA1 to P-DMA0, Channels 0-7 are used
25:28	MDMA_TR_OUT[0:3]	Cross connections from M-DMA0 to P-DMA0
29:32	FAULT_TR_OUT[0:3]	Allow faults to initiate data transfer for debug purposes
33:34	CTI_TR_OUT[0:1]	Trace events
35:38	EVTGEN_TR_OUT[3:6]	EVTGEN triggers
39:54	HSIOM_IO_INPUT[0:15]	I/O inputs
MUX Group 1: PDMA1_TR (P-DMA1 trigger multiplexer)		
1:16	PDMA0_TR_OUT[0:15]	Allow P-DMA0 to trigger P-DMA1
17:24	PDMA1_TR_OUT[0:7]	Allow P-DMA1 to chain to itself, useful for triggering once per row for 2D transfer
25:28	MDMA_TR_OUT[0:3]	Allow M-DMA0 to trigger P-DMA0
29:32	FAULT_TR_OUT[0:3]	Allow faults to initiate data transfer for debug purposes
33:34	CTI_TR_OUT[0:1]	Trace events
35:38	EVTGEN_TR_OUT[7:10]	EVTGEN triggers
39:54	HSIOM_IO_INPUT[16:31]	I/O inputs
55:60	PASS_GEN_TR_OUT[0:5]	PASS SAR events
MUX Group 2: MDMA (M-DMA0 trigger multiplexer)		
1:4	MDMA_TR_OUT[0:3]	Allow M-DMA0 to trigger itself
MUX Group 3: TCPWM_TO_PDMA0 (TCPWM0 to P-DMA0 trigger multiplexer)		
1:4	TCPWM_32_TR_OUT0[0:3]	32-bit TCPWM0 counters
5:16	TCPWM_16M_TR_OUT0[0:11]	16-bit motor enhanced TCPWM0 counters
17:79	TCPWM_16_TR_OUT0[0:62]	16-bit TCPWM0 counters
80:82	CAN0_TT_TR_OUT[0:2]	CAN0 TT sync outputs
83:85	CAN1_TT_TR_OUT[0:2]	CAN1 TT sync outputs
MUX Group 4: TCPWM_OUT (TCPWM0 loop back multiplexer)		
1:4	TCPWM_32_TR_OUT0[0:3]	32-bit TCPWM0 counters

(table continues...)

[23] "x:y" depicts a range starting from 'x' through 'y'.

Table 20 (continued) Trigger inputs

Input	Trigger	Description
5:16	TCPWM_16M_TR_OUT0[0:11]	16-bit Motor enhanced TCPWM0 counters
17:79	TCPWM_16_TR_OUT0[0:62]	16-bit TCPWM0 counters
80:87	TCPWM_16_TR_OUT1[0:7]	Allows feedback of two signals from same counters
88:90	CAN0_TT_TR_OUT[0:2]	CAN0 TT sync outputs
91:93	CAN1_TT_TR_OUT[0:2]	CAN1 TT sync outputs
MUX Group 5: TCPWM_IN (TCPWM0 Trigger Multiplexer)		
1:16	PDMA0_TR_OUT[0:15]	General-purpose P-DMA0 triggers
17:24	PDMA1_TR_OUT[0:7]	General-purpose P-DMA1 triggers
25:28	MDMA_TR_OUT[0:3]	M-DMA0 triggers
29:30	CTI_TR_OUT[0:1]	Trace events
31:34	FAULT_TR_OUT[0:3]	Fault events
35:40	PASS_GEN_TR_OUT[0:5]	PASS SAR events
41:72	HSIOM_IO_INPUT[0:31]	I/O inputs
73	SCB_TX_TR_OUT[0]	SCB0 TX trigger
74	SCB_RX_TR_OUT[0]	SCB0 RX trigger
75	SCB_I2C_SCL_TR_OUT[0]	SCB0 I2C trigger
76	SCB_TX_TR_OUT[1]	SCB1 TX trigger
77	SCB_RX_TR_OUT[1]	SCB1 RX trigger
78	SCB_I2C_SCL_TR_OUT[1]	SCB1 I2C trigger
79	SCB_TX_TR_OUT[2]	SCB2 TX trigger
80	SCB_RX_TR_OUT[2]	SCB2 RX trigger
81	SCB_I2C_SCL_TR_OUT[2]	SCB2 I2C trigger
82	SCB_TX_TR_OUT[3]	SCB3 TX trigger
83	SCB_RX_TR_OUT[3]	SCB3 RX trigger
84	SCB_I2C_SCL_TR_OUT[3]	SCB3 I2C trigger
85	SCB_TX_TR_OUT[4]	SCB4 TX trigger
86	SCB_RX_TR_OUT[4]	SCB4 RX trigger
87	SCB_I2C_SCL_TR_OUT[4]	SCB4 I2C trigger
88	SCB_TX_TR_OUT[5]	SCB5 TX trigger
89	SCB_RX_TR_OUT[5]	SCB5 RX trigger
90	SCB_I2C_SCL_TR_OUT[5]	SCB5 I2C trigger
91	SCB_TX_TR_OUT[6]	SCB6 TX trigger
92	SCB_RX_TR_OUT[6]	SCB6 RX trigger

(table continues...)

Table 20 (continued) Trigger inputs

Input	Trigger	Description
93	SCB_I2C_SCL_TR_OUT[6]	SCB6 I2C trigger
94	SCB_TX_TR_OUT[7]	SCB7 TX trigger
95	SCB_RX_TR_OUT[7]	SCB7 RX trigger
96	SCB_I2C_SCL_TR_OUT[7]	SCB7 I2C trigger
97:99	CAN0_DBG_TR_OUT[0:2]	CAN0 M-DMA0 events
100:102	CAN0_FIFO0_TR_OUT[0:2]	CAN0 FIFO0 events
103:105	CAN0_FIFO1_TR_OUT[0:2]	CAN0 FIFO1 events
106:108	CAN1_DBG_TR_OUT[0:2]	CAN1 M-DMA0 events
109:111	CAN1_FIFO0_TR_OUT[0:2]	CAN1 FIFO0 events
112:114	CAN1_FIFO1_TR_OUT[0:2]	CAN1 FIFO1 events
97:99	EVTGEN_TR_OUT[3:10]	EVTGEN triggers
MUX Group 6: PASS (PASS SAR trigger multiplexer)		
1:16	PDMA0_TR_OUT[0:15]	General purpose P-DMA0 triggers
17:18	CTI_TR_OUT[0:1]	Trace events
19:22	FAULT_TR_OUT[0:3]	Fault events
23:25	EVTGEN_TR_OUT[0:2]	EVTGEN triggers
26:31	PASS_GEN_TR_OUT[0:5]	PASS SAR done signals
32:63	HSIOM_IO_INPUT[0:31]	I/O inputs
64:67	TCPWM_32_TR_OUT1[0:3]	32-bit TCPWM0 counters
68:79	TCPWM_16M_TR_OUT1[0:11]	16-bit Motor enhanced TCPWM0 counters
MUX Group 7: CAN TT sync triggers		
1:3	CAN0_TT_TR_OUT[0:2]	CAN0 TT sync outputs
4:6	CAN1_TT_TR_OUT[0:2]	CAN1 TT sync outputs
MUX Group 8: DebugMain (Debug Multiplexer)		
1:5	TR_GROUP9_OUTPUT[0:4]	Output from debug reduction multiplexer #1
6:10	TR_GROUP10_OUTPUT[0:4]	Output from debug reduction multiplexer #2
MUX Group 9: DebugReduction1 (Debug Reduction #1)		
1:89	PDMA0_TR_OUT[0:88]	P-DMA0 triggers
90:97	SCB_TX_TR_OUT[0:7]	SCB tx triggers
98:105	SCB_RX_TR_OUT[0:7]	SCB rx triggers
106:113	SCB_I2C_SCL_TR_OUT[0:7]	SCB I2C triggers
114:116	CAN0_DBG_TR_OUT[0:2]	CAN0 P-DMA
117:119	CAN0_FIFO0_TR_OUT[0:2]	CAN0 FIFO0

(table continues...)

Table 20 (continued) Trigger inputs

Input	Trigger	Description
120:122	CAN0_FIFO1_TR_OUT[0:2]	CAN0 FIFO1
123:125	CAN0_TT_TR_OUT[0:2]	CAN TT sync outputs
126:128	CAN1_DBG_TR_OUT[0:2]	CAN1 P-DMA
129:131	CAN1_FIFO0_TR_OUT[0:2]	CAN1 FIFO0
132:134	CAN1_FIFO1_TR_OUT[0:2]	CAN1 FIFO1
135:137	CAN1_TT_TR_OUT[0:2]	CAN TT sync outputs
138:139	CTI_TR_OUT[0:1]	Trace events
140:143	FAULT_TR_OUT[0:3]	Fault events
144:147	TCPWM_32_TR_OUT0[0:7]	32-bit TCPWM0 counters
148:159	TCPWM_16M_TR_OUT0[0:11]	16-bit Motor enhanced TCPWM0 counters
160:222	TCPWM_16_TR_OUT0[0:62]	16-bit TCPWM0 counters
MUX Group 10: DebugReduction2 (Debug Reduction #2)		
1:33	PDMA1_TR_OUT[0:32]	P-DMA1 triggers
34:37	MDMA_TR_OUT[0:3]	M-DMA triggers
38:41	TCPWM_32_TR_OUT1[0:3]	32-bit TCPWM0 counters
42:53	TCPWM_16M_TR_OUT1[0:11]	16-bit Motor enhanced TCPWM0 counters
54:116	TCPWM_16_TR_OUT1[0:62]	16-bit TCPWM0 counters
117:122	PASS_GEN_TR_OUT[0:5]	PASS SAR conversion complete events
123:133	EVTGEN_TR_OUT[0:10]	EVTGEN Triggers

18 Triggers group outputs

Table 21 Trigger outputs

Output	Trigger	Description
MUX Group 0: PDMA0_TR (P-DMA0 trigger multiplexer)		
0:7	PDMA0_TR_IN[0:7]	Triggers to P-DMA0[0:7]
MUX Group 1: PDMA1_TR (P-DMA1 trigger multiplexer)		
0:7	PDMA1_TR_IN[0:7]	Triggers to P-DMA1[0:7]
MUX Group 2: MDMA (M-DMA0 trigger multiplexer)		
0:3	MDMA_TR_IN[0:3]	Triggers to M-DMA0
MUX Group 3: TCPWM_TO_PDMA0 (TCPWM0 to P-DMA0 trigger multiplexer)		
0:7	PDMA0_TR_IN[8:15]	Triggers to P-DMA0[8:15]
MUX Group 4: TCPWM_OUT (TCPWM0 loop back multiplexer)		
0:15	TCPWM_ALL_CNT_TR_IN[0:15]	All counters trigger input
MUX Group 5: TCPWM_IN (TCPWM0 Trigger Multiplexer)		
0:10	TCPWM_ALL_CNT_TR_IN[16:26]	Triggers to TCPWM0
MUX Group 6: PASS (PASS SAR trigger multiplexer)		
0:11	PASS_GEN_TR_IN[0:11]	Triggers to SAR ADCs
MUX Group 7: CAN TT sync triggers		
0:2	CAN0_TT_TR_IN[0:2]	CAN0 TT Sync Inputs
3:5	CAN1_TT_TR_IN[0:2]	CAN1 TT Sync Inputs
MUX Group 8: DebugMain (Debug Multiplexer)		
0:1	HSIOM_IO_OUTPUT[0:1]	To HSIOM as an output
2:3	CTI_TR_IN[0:1]	To CPU Cross Trigger system
4	PERI_DEBUG_FREEZE_TR_IN	Signal to Freeze PERI operation
5	PASS_DEBUG_FREEZE_TR_IN	Signal to Freeze SAR ADC operation
6	SRSS_WDT_DEBUG_FREEZE_TR_IN	Signal to Freeze WDT operation
7:8	SRSS_MCWDT_DEBUG_FREEZE_TR_IN[0:1]	Signal to Freeze MCWDT operation
9	TCPWM_DEBUG_FREEZE_TR_IN	Signal to Freeze TCPWM0 operation
MUX Group 9: DebugReduction1 (Debug Reduction #1)		
0:4	TR_GROUP8_INPUT[1:5]	To main debug multiplexer
MUX Group 10: DebugReduction2 (Debug Reduction #2)		
0:4	TR_GROUP8_INPUT[6:10]	To main debug multiplexer

19 Triggers one-to-one

Table 22 Triggers 1:1

Input	Trigger in	Trigger out	Description
MUX Group 1: TCPWM0 to PASS SARx direct connect			
0	TCPWM0_16M_TR_OUT1[0]	PASS0_CH_TR_IN[0]	TCPWM0 Group #1 Counter #00 (PWM0_M_0) to SAR0 ch#0
1	TCPWM0_16M_TR_OUT1[1]	PASS0_CH_TR_IN[1]	TCPWM0 Group #1 Counter #03 (PWM0_M_3) to SAR0 ch#1
2	TCPWM0_16M_TR_OUT1[2]	PASS0_CH_TR_IN[2]	TCPWM0 Group #1 Counter #06 (PWM0_M_6) to SAR0 ch#2
3	TCPWM0_16M_TR_OUT1[3]	PASS0_CH_TR_IN[3]	TCPWM0 Group #1 Counter #09 (PWM0_M_9) to SAR0 ch#3
4:23	TCPWM0_16_TR_OUT1[0:19]	PASS0_CH_TR_IN[4:23]	TCPWM0 Group #0 Counter #00 through 19 (PWM0_0 to PWM0_19) to SAR0 ch#4 through SAR0 ch#23
24	TCPWM0_16M_TR_OUT1[4]	PASS0_CH_TR_IN[32]	TCPWM0 Group #1 Counter #01 (PWM0_M_1) to SAR1 ch#0
25	TCPWM0_16M_TR_OUT1[5]	PASS0_CH_TR_IN[33]	TCPWM0 Group #1 Counter #04 (PWM0_M_4) to SAR1 ch#1
26	TCPWM0_16M_TR_OUT1[6]	PASS0_CH_TR_IN[34]	TCPWM0 Group #1 Counter #07 (PWM0_M_7) to SAR1 ch#2
27	TCPWM0_16M_TR_OUT1[7]	PASS0_CH_TR_IN[35]	TCPWM0 Group #1 Counter #10 (PWM0_M_10) to SAR1 ch#3
28:55	TCPWM0_16_TR_OUT1[20:47]	PASS0_CH_TR_IN[36:63]	TCPWM0 Group #0 Counter #20 through 47 (PWM0_20 to PWM0_47) to SAR1 ch#4 through SAR1 ch#31
56	TCPWM0_16M_TR_OUT1[8]	PASS0_CH_TR_IN[64]	TCPWM0 Group #1 Counter #02 (PWM0_M_2) to SAR2 ch#0
57	TCPWM0_16M_TR_OUT1[9]	PASS0_CH_TR_IN[65]	TCPWM0 Group #1 Counter #05 (PWM0_M_5) to SAR2 ch#1
58	TCPWM0_16M_TR_OUT1[10]	PASS0_CH_TR_IN[66]	TCPWM0 Group #1 Counter #08 (PWM0_M_8) to SAR2 ch#2
59	TCPWM0_16M_TR_OUT1[11]	PASS0_CH_TR_IN[67]	TCPWM0 Group #1 Counter #11 (PWM0_M_11) to SAR2 ch#3
60:63	TCPWM0_16_TR_OUT1[48:51]	PASS0_CH_TR_IN[68:71]	TCPWM0 Group #0 Counter #48 through 51 (PWM0_48 to PWM0_51) to SAR2 ch#4 through SAR2 ch#7
MUX Group 2: PASS SARx to P-DMA0 direct connect			
0:23	PASS0_CH_DONE_TR_OUT[0:23]	PDMA0_TR_IN[25:48]	PASS SAR0 [0:23] to P-DMA0 direct connect

(table continues...)

Table 22 (continued) Triggers 1:1

Input	Trigger in	Trigger out	Description
24:55	PASS0_CH_DONE_TR_OUT[32:63]	PDMA0_TR_IN[49:80]	PASS SAR1 [0:31] to P-DMA0 direct connect
56:63	PASS0_CH_DONE_TR_OUT[64:71]	PDMA0_TR_IN[81:88]	PASS SAR2 [0:7] to P-DMA0 direct connect
MUX Group 3: PASS SARx to TCPWM0 direct connect			
0	PASS0_CH_RANGEVIO_TR_OUT[0]	TCPWM0_16M_ONE_CNT_TR_IN[0]	SAR0 ch#0[24], range violation to TCPWM0 Group #1 Counter #00 trig=4
1	PASS0_CH_RANGEVIO_TR_OUT[1]	TCPWM0_16M_ONE_CNT_TR_IN[3]	SAR0 ch#1, range violation to TCPWM0 Group #1 Counter #03 trig=4
2	PASS0_CH_RANGEVIO_TR_OUT[2]	TCPWM0_16M_ONE_CNT_TR_IN[6]	SAR0 ch#2, range violation to TCPWM0 Group #1 Counter #06 trig=4
3	PASS0_CH_RANGEVIO_TR_OUT[3]	TCPWM0_16M_ONE_CNT_TR_IN[9]	SAR0 ch#3, range violation to TCPWM0 Group #1 Counter #09 trig=4
4	PASS0_CH_RANGEVIO_TR_OUT[4]	TCPWM0_16_ONE_CNT_TR_IN[0]	SAR0 ch#4, range violation to TCPWM0 Group #0 Counter #00 trig=4
5	PASS0_CH_RANGEVIO_TR_OUT[5]	TCPWM0_16_ONE_CNT_TR_IN[1]	SAR0 ch#5, range violation to TCPWM0 Group #0 Counter #01 trig=4
6	PASS0_CH_RANGEVIO_TR_OUT[6]	TCPWM0_16_ONE_CNT_TR_IN[2]	SAR0 ch#6, range violation to TCPWM0 Group #0 Counter #02 trig=4
7	PASS0_CH_RANGEVIO_TR_OUT[7]	TCPWM0_16_ONE_CNT_TR_IN[3]	SAR0 ch#7, range violation to TCPWM0 Group #0 Counter #03 trig=4
8	PASS0_CH_RANGEVIO_TR_OUT[8]	TCPWM0_16_ONE_CNT_TR_IN[4]	SAR0 ch#8, range violation to TCPWM0 Group #0 Counter #04 trig=4
9	PASS0_CH_RANGEVIO_TR_OUT[9]	TCPWM0_16_ONE_CNT_TR_IN[5]	SAR0 ch#9, range violation to TCPWM0 Group #0 Counter #05 trig=4
10	PASS0_CH_RANGEVIO_TR_OUT[10]	TCPWM0_16_ONE_CNT_TR_IN[6]	SAR0 ch#10, range violation to TCPWM0 Group #0 Counter #06 trig=4
11	PASS0_CH_RANGEVIO_TR_OUT[11]	TCPWM0_16_ONE_CNT_TR_IN[7]	SAR0 ch#11, range violation to TCPWM0 Group #0 Counter #07 trig=4
12	PASS0_CH_RANGEVIO_TR_OUT[12]	TCPWM0_16_ONE_CNT_TR_IN[8]	SAR0 ch#12, range violation to TCPWM0 Group #0 Counter #08 trig=4
13	PASS0_CH_RANGEVIO_TR_OUT[13]	TCPWM0_16_ONE_CNT_TR_IN[9]	SAR0 ch#13, range violation to TCPWM0 Group #0 Counter #09 trig=4
14	PASS0_CH_RANGEVIO_TR_OUT[14]	TCPWM0_16_ONE_CNT_TR_IN[10]	SAR0 ch#14, range violation to TCPWM0 Group #0 Counter #10 trig=4
15	PASS0_CH_RANGEVIO_TR_OUT[15]	TCPWM0_16_ONE_CNT_TR_IN[11]	SAR0 ch#15, range violation to TCPWM0 Group #0 Counter #11 trig=4

(table continues...)

[24] Each logical channel of SAR ADC[x] can be connected to any of the SAR ADC[x]_y external pin. (x = 0, or 1, or, 2 and y=0 to maximum 31)

Table 22 (continued) Triggers 1:1

Input	Trigger in	Trigger out	Description
16	PASS0_CH_RANGEVIO_TR_OUT[16]	TCPWM0_16_ONE_CNT_TR_IN[12]	SAR0 ch#16, range violation to TCPWM0 Group #0 Counter #12 trig=4
17	PASS0_CH_RANGEVIO_TR_OUT[17]	TCPWM0_16_ONE_CNT_TR_IN[13]	SAR0 ch#17, range violation to TCPWM0 Group #0 Counter #13 trig=4
18	PASS0_CH_RANGEVIO_TR_OUT[18]	TCPWM0_16_ONE_CNT_TR_IN[14]	SAR0 ch#18, range violation to TCPWM0 Group #0 Counter #14 trig=4
19	PASS0_CH_RANGEVIO_TR_OUT[19]	TCPWM0_16_ONE_CNT_TR_IN[15]	SAR0 ch#19, range violation to TCPWM0 Group #0 Counter #15 trig=4
20	PASS0_CH_RANGEVIO_TR_OUT[20]	TCPWM0_16_ONE_CNT_TR_IN[16]	SAR0 ch#20, range violation to TCPWM0 Group #0 Counter #16 trig=4
21	PASS0_CH_RANGEVIO_TR_OUT[21]	TCPWM0_16_ONE_CNT_TR_IN[17]	SAR0 ch#21, range violation to TCPWM0 Group #0 Counter #17 trig=4
22	PASS0_CH_RANGEVIO_TR_OUT[22]	TCPWM0_16_ONE_CNT_TR_IN[18]	SAR0 ch#22, range violation to TCPWM0 Group #0 Counter #18 trig=4
23	PASS0_CH_RANGEVIO_TR_OUT[23]	TCPWM0_16_ONE_CNT_TR_IN[19]	SAR0 ch#23, range violation to TCPWM0 Group #0 Counter #19 trig=4
24	PASS0_CH_RANGEVIO_TR_OUT[32]	TCPWM0_16M_ONE_CNT_TR_IN[1]	SAR1 ch#0, range violation to TCPWM0 Group #1 Counter #01 trig=4
25	PASS0_CH_RANGEVIO_TR_OUT[33]	TCPWM0_16M_ONE_CNT_TR_IN[4]	SAR1 ch#1, range violation to TCPWM0 Group #1 Counter #04 trig=4
26	PASS0_CH_RANGEVIO_TR_OUT[34]	TCPWM0_16M_ONE_CNT_TR_IN[7]	SAR1 ch#2, range violation to TCPWM0 Group #1 Counter #07 trig=4
27	PASS0_CH_RANGEVIO_TR_OUT[35]	TCPWM0_16M_ONE_CNT_TR_IN[10]	SAR1 ch#3, range violation to TCPWM0 Group #1 Counter #10 trig=4
28	PASS0_CH_RANGEVIO_TR_OUT[36]	TCPWM0_16_ONE_CNT_TR_IN[20]	SAR1 ch#4, range violation to TCPWM0 Group #0 Counter #20 trig=4
29	PASS0_CH_RANGEVIO_TR_OUT[37]	TCPWM0_16_ONE_CNT_TR_IN[21]	SAR1 ch#5, range violation to TCPWM0 Group #0 Counter #21 trig=4
30	PASS0_CH_RANGEVIO_TR_OUT[38]	TCPWM0_16_ONE_CNT_TR_IN[22]	SAR1 ch#6, range violation to TCPWM0 Group #0 Counter #22 trig=4
31	PASS0_CH_RANGEVIO_TR_OUT[39]	TCPWM0_16_ONE_CNT_TR_IN[23]	SAR1 ch#7, range violation to TCPWM0 Group #0 Counter #23 trig=4
32	PASS0_CH_RANGEVIO_TR_OUT[40]	TCPWM0_16_ONE_CNT_TR_IN[24]	SAR1 ch#8, range violation to TCPWM0 Group #0 Counter #24 trig=4
33	PASS0_CH_RANGEVIO_TR_OUT[41]	TCPWM0_16_ONE_CNT_TR_IN[25]	SAR1 ch#9, range violation to TCPWM0 Group #0 Counter #25 trig=4
34	PASS0_CH_RANGEVIO_TR_OUT[42]	TCPWM0_16_ONE_CNT_TR_IN[26]	SAR1 ch#10, range violation to TCPWM0 Group #0 Counter #26 trig=4

(table continues...)

Table 22 (continued) Triggers 1:1

Input	Trigger in	Trigger out	Description
35	PASS0_CH_RANGEVIO_TR_OUT[43]	TCPWM0_16_ONE_CNT_TR_IN[27]	SAR1 ch#11, range violation to TCPWM0 Group #0 Counter #27 trig=4
36	PASS0_CH_RANGEVIO_TR_OUT[44]	TCPWM0_16_ONE_CNT_TR_IN[28]	SAR1 ch#12, range violation to TCPWM0 Group #0 Counter #28 trig=4
37	PASS0_CH_RANGEVIO_TR_OUT[45]	TCPWM0_16_ONE_CNT_TR_IN[29]	SAR1 ch#13, range violation to TCPWM0 Group #0 Counter #29 trig=4
38	PASS0_CH_RANGEVIO_TR_OUT[46]	TCPWM0_16_ONE_CNT_TR_IN[30]	SAR1 ch#14, range violation to TCPWM0 Group #0 Counter #30 trig=4
39	PASS0_CH_RANGEVIO_TR_OUT[47]	TCPWM0_16_ONE_CNT_TR_IN[31]	SAR1 ch#15, range violation to TCPWM0 Group #0 Counter #31 trig=4
40	PASS0_CH_RANGEVIO_TR_OUT[48]	TCPWM0_16_ONE_CNT_TR_IN[32]	SAR1 ch#16, range violation to TCPWM0 Group #0 Counter #32 trig=4
41	PASS0_CH_RANGEVIO_TR_OUT[49]	TCPWM0_16_ONE_CNT_TR_IN[33]	SAR1 ch#17, range violation to TCPWM0 Group #0 Counter #33 trig=4
42	PASS0_CH_RANGEVIO_TR_OUT[50]	TCPWM0_16_ONE_CNT_TR_IN[34]	SAR1 ch#18, range violation to TCPWM0 Group #0 Counter #34 trig=4
43	PASS0_CH_RANGEVIO_TR_OUT[51]	TCPWM0_16_ONE_CNT_TR_IN[35]	SAR1 ch#19, range violation to TCPWM0 Group #0 Counter #35 trig=4
44	PASS0_CH_RANGEVIO_TR_OUT[52]	TCPWM0_16_ONE_CNT_TR_IN[36]	SAR1 ch#20, range violation to TCPWM0 Group #0 Counter #36 trig=4
45	PASS0_CH_RANGEVIO_TR_OUT[53]	TCPWM0_16_ONE_CNT_TR_IN[37]	SAR1 ch#21, range violation to TCPWM0 Group #0 Counter #37 trig=4
46	PASS0_CH_RANGEVIO_TR_OUT[54]	TCPWM0_16_ONE_CNT_TR_IN[38]	SAR1 ch#22, range violation to TCPWM0 Group #0 Counter #38 trig=4
47	PASS0_CH_RANGEVIO_TR_OUT[55]	TCPWM0_16_ONE_CNT_TR_IN[39]	SAR1 ch#23, range violation to TCPWM0 Group #0 Counter #39 trig=4
48	PASS0_CH_RANGEVIO_TR_OUT[56]	TCPWM0_16_ONE_CNT_TR_IN[40]	SAR1 ch#24, range violation to TCPWM0 Group #0 Counter #40 trig=4
49	PASS0_CH_RANGEVIO_TR_OUT[57]	TCPWM0_16_ONE_CNT_TR_IN[41]	SAR1 ch#25, range violation to TCPWM0 Group #0 Counter #41 trig=4
50	PASS0_CH_RANGEVIO_TR_OUT[58]	TCPWM0_16_ONE_CNT_TR_IN[42]	SAR1 ch#26, range violation to TCPWM0 Group #0 Counter #42 trig=4
51	PASS0_CH_RANGEVIO_TR_OUT[59]	TCPWM0_16_ONE_CNT_TR_IN[43]	SAR1 ch#27, range violation to TCPWM0 Group #0 Counter #43 trig=4
52	PASS0_CH_RANGEVIO_TR_OUT[60]	TCPWM0_16_ONE_CNT_TR_IN[44]	SAR1 ch#28, range violation to TCPWM0 Group #0 Counter #44 trig=4
53	PASS0_CH_RANGEVIO_TR_OUT[61]	TCPWM0_16_ONE_CNT_TR_IN[45]	SAR1 ch#29, range violation to TCPWM0 Group #0 Counter #45 trig=4

(table continues...)

Table 22 (continued) Triggers 1:1

Input	Trigger in	Trigger out	Description
54	PASS0_CH_RANGEVIO_TR_OUT[62]	TCPWM0_16_ONE_CNT_TR_IN[46]	SAR1 ch#30, range violation to TCPWM0 Group #0 Counter #46 trig=4
55	PASS0_CH_RANGEVIO_TR_OUT[63]	TCPWM0_16_ONE_CNT_TR_IN[47]	SAR1 ch#31, range violation to TCPWM0 Group #0 Counter #47 trig=4
56	PASS0_CH_RANGEVIO_TR_OUT[64]	TCPWM0_16M_ONE_CNT_TR_IN[2]	SAR2 ch#0, range violation to TCPWM0 Group #1 Counter #02 trig=4
57	PASS0_CH_RANGEVIO_TR_OUT[65]	TCPWM0_16M_ONE_CNT_TR_IN[5]	SAR2 ch#1, range violation to TCPWM0 Group #1 Counter #05 trig=4
58	PASS0_CH_RANGEVIO_TR_OUT[66]	TCPWM0_16M_ONE_CNT_TR_IN[8]	SAR2 ch#2, range violation to TCPWM0 Group #1 Counter #08 trig=4
59	PASS0_CH_RANGEVIO_TR_OUT[67]	TCPWM0_16M_ONE_CNT_TR_IN[11]	SAR2 ch#3, range violation to TCPWM0 Group #1 Counter #11 trig=4
60	PASS0_CH_RANGEVIO_TR_OUT[68]	TCPWM0_16_ONE_CNT_TR_IN[48]	SAR2 ch#4, range violation to TCPWM0 Group #0 Counter #48 trig=4
61	PASS0_CH_RANGEVIO_TR_OUT[69]	TCPWM0_16_ONE_CNT_TR_IN[49]	SAR2 ch#5, range violation to TCPWM0 Group #0 Counter #49 trig=4
62	PASS0_CH_RANGEVIO_TR_OUT[70]	TCPWM0_16_ONE_CNT_TR_IN[50]	SAR2 ch#6, range violation to TCPWM0 Group #0 Counter #50 trig=4
63	PASS0_CH_RANGEVIO_TR_OUT[71]	TCPWM0_16_ONE_CNT_TR_IN[51]	SAR2 ch#7, range violation to TCPWM0 Group #0 Counter #51 trig=4
MUX Group 4: CAN0 to P-DMA0 Triggers			
0	CAN0_DBG_TR_OUT[0]	PDMA0_TR_IN[16]	CAN0, Channel #0 P-DMA0 trigger
1	CAN0_FIFO0_TR_OUT[0]	PDMA0_TR_IN[17]	CAN0, Channel #0 FIFO0 trigger
2	CAN0_FIFO1_TR_OUT[0]	PDMA0_TR_IN[18]	CAN0, Channel #0 FIFO1 trigger
3	CAN0_DBG_TR_OUT[1]	PDMA0_TR_IN[19]	CAN0, Channel #1 P-DMA0 trigger
4	CAN0_FIFO0_TR_OUT[1]	PDMA0_TR_IN[20]	CAN0, Channel #1 FIFO0 trigger
5	CAN0_FIFO1_TR_OUT[1]	PDMA0_TR_IN[21]	CAN0, Channel #1 FIFO1 trigger
6	CAN0_DBG_TR_OUT[2]	PDMA0_TR_IN[22]	CAN0, Channel #2 P-DMA0 trigger
7	CAN0_FIFO0_TR_OUT[2]	PDMA0_TR_IN[23]	CAN0, Channel #2 FIFO0 trigger
8	CAN0_FIFO1_TR_OUT[2]	PDMA0_TR_IN[24]	CAN0, Channel #2 FIFO1 trigger
MUX Group 5: CAN1 to P-DMA1 triggers			
0	CAN1_DBG_TR_OUT[0]	PDMA1_TR_IN[24]	CAN1, Channel #0 P-DMA01 trigger
1	CAN1_FIFO0_TR_OUT[0]	PDMA1_TR_IN[25]	CAN1, Channel #0 FIFO0 trigger
2	CAN1_FIFO1_TR_OUT[0]	PDMA1_TR_IN[26]	CAN1, Channel #0 FIFO1 trigger
3	CAN1_DBG_TR_OUT[1]	PDMA1_TR_IN[27]	CAN1, Channel #1 P-DMA1 trigger
4	CAN1_FIFO0_TR_OUT[1]	PDMA1_TR_IN[28]	CAN1, Channel #1 FIFO0 trigger

(table continues...)

Table 22 (continued) Triggers 1:1

Input	Trigger in	Trigger out	Description
5	CAN1_FIFO1_TR_OUT[1]	PDMA1_TR_IN[29]	CAN1, Channel #1 FIFO1 trigger
6	CAN1_DBG_TR_OUT[2]	PDMA1_TR_IN[30]	CAN1, Channel #2 P-DMA1 trigger
7	CAN1_FIFO0_TR_OUT[2]	PDMA1_TR_IN[31]	CAN1, Channel #2 FIFO0 trigger
8	CAN1_FIFO1_TR_OUT[2]	PDMA1_TR_IN[32]	CAN1, Channel #2 FIFO1 trigger
MUX Group 6:Acknowledge triggers from P-DMA0 to CAN0			
0	PDMA0_TR_OUT[16]	CAN0_DBG_TR_ACK[0]	CAN0, Channel #0 P-DMA0 acknowledge
1	PDMA0_TR_OUT[19]	CAN0_DBG_TR_ACK[1]	CAN0, Channel #1 P-DMA0 acknowledge
2	PDMA0_TR_OUT[22]	CAN0_DBG_TR_ACK[2]	CAN0, Channel #2 P-DMA0 acknowledge
MUX Group 7: Acknowledge triggers from P-DMA1 to CAN1			
0	PDMA1_TR_OUT[24]	CAN1_DBG_TR_ACK[0]	CAN1, Channel #0 P-DMA1 acknowledge
1	PDMA1_TR_OUT[27]	CAN1_DBG_TR_ACK[1]	CAN1, Channel #1 P-DMA1 acknowledge
2	PDMA1_TR_OUT[30]	CAN1_DBG_TR_ACK[2]	CAN1, Channel #2 P-DMA1 acknowledge
MUX Group 8: SCBx to P-DMA1 Triggers			
0	SCB0_TX_TR_OUT	PDMA1_TR_IN[8]	SCB0 TX to P-DMA1 Trigger
1	SCB0_RX_TR_OUT	PDMA1_TR_IN[9]	SCB0 RX to P-DMA1 Trigger
2	SCB1_TX_TR_OUT	PDMA1_TR_IN[10]	SCB1 TX to P-DMA1 Trigger
3	SCB1_RX_TR_OUT	PDMA1_TR_IN[11]	SCB1 RX to P-DMA1 Trigger
4	SCB2_TX_TR_OUT	PDMA1_TR_IN[12]	SCB2 TX to P-DMA1 Trigger
5	SCB2_RX_TR_OUT	PDMA1_TR_IN[13]	SCB2 RX to P-DMA1 Trigger
6	SCB3_TX_TR_OUT	PDMA1_TR_IN[14]	SCB3 TX to P-DMA1 Trigger
7	SCB3_RX_TR_OUT	PDMA1_TR_IN[15]	SCB3 RX to P-DMA1 Trigger
8	SCB4_TX_TR_OUT	PDMA1_TR_IN[16]	SCB4 TX to P-DMA1 Trigger
9	SCB4_RX_TR_OUT	PDMA1_TR_IN[17]	SCB4 RX to P-DMA1 Trigger
10	SCB5_TX_TR_OUT	PDMA1_TR_IN[18]	SCB5 TX to P-DMA1 Trigger
11	SCB5_RX_TR_OUT	PDMA1_TR_IN[19]	SCB5 RX to P-DMA1 Trigger
12	SCB6_TX_TR_OUT	PDMA1_TR_IN[20]	SCB6 TX to P-DMA1 Trigger
13	SCB6_RX_TR_OUT	PDMA1_TR_IN[21]	SCB6 RX to P-DMA1 Trigger
14	SCB7_TX_TR_OUT	PDMA1_TR_IN[22]	SCB7 TX to P-DMA1 Trigger
15	SCB7_RX_TR_OUT	PDMA1_TR_IN[23]	SCB7 RX to P-DMA1 Trigger

20 Peripheral clocks

Table 23 Peripheral clock assignments

Output	Destination	Description
0	PCLK_CPUSS_CLOCK_TRACE_IN	Trace clock
1	PCLK_SMARTIO12_CLOCK	SMART I/O #12
2	PCLK_SMARTIO13_CLOCK	SMART I/O #13
3	PCLK_SMARTIO14_CLOCK	SMART I/O #14
4	PCLK_SMARTIO15_CLOCK	SMART I/O #15
5	PCLK_SMARTIO16_CLOCK	SMART I/O #16
6	PCLK_CANFD0_CLOCK_CAN0	CAN0, Channel #0
7	PCLK_CANFD0_CLOCK_CAN1	CAN0, Channel #1
8	PCLK_CANFD0_CLOCK_CAN2	CAN0, Channel #2
9	PCLK_CANFD1_CLOCK_CAN0	CAN1, Channel #0
10	PCLK_CANFD1_CLOCK_CAN1	CAN1, Channel #1
11	PCLK_CANFD1_CLOCK_CAN2	CAN1, Channel #2
20	PCLK_SCB0_CLOCK	SCB0
21	PCLK_SCB1_CLOCK	SCB1
22	PCLK_SCB2_CLOCK	SCB2
23	PCLK_SCB3_CLOCK	SCB3
24	PCLK_SCB4_CLOCK	SCB4
25	PCLK_SCB5_CLOCK	SCB5
26	PCLK_SCB6_CLOCK	SCB6
27	PCLK_SCB7_CLOCK	SCB7
28	PCLK_PASS0_CLOCK_SAR0	SAR0
29	PCLK_PASS0_CLOCK_SAR1	SAR1
30	PCLK_PASS0_CLOCK_SAR2	SAR2
31	PCLK_TCPWM0_CLOCKS0	TCPWM0 Group #0, Counter #0
32	PCLK_TCPWM0_CLOCKS1	TCPWM0 Group #0, Counter #1
33	PCLK_TCPWM0_CLOCKS2	TCPWM0 Group #0, Counter #2
34	PCLK_TCPWM0_CLOCKS3	TCPWM0 Group #0, Counter #3
35	PCLK_TCPWM0_CLOCKS4	TCPWM0 Group #0, Counter #4
36	PCLK_TCPWM0_CLOCKS5	TCPWM0 Group #0, Counter #5
37	PCLK_TCPWM0_CLOCKS6	TCPWM0 Group #0, Counter #6
38	PCLK_TCPWM0_CLOCKS7	TCPWM0 Group #0, Counter #7
39	PCLK_TCPWM0_CLOCKS8	TCPWM0 Group #0, Counter #8

(table continues...)

Table 23 (continued) Peripheral clock assignments

Output	Destination	Description
40	PCLK_TCPWM0_CLOCKS9	TCPWM0 Group #0, Counter #9
41	PCLK_TCPWM0_CLOCKS10	TCPWM0 Group #0, Counter #10
42	PCLK_TCPWM0_CLOCKS11	TCPWM0 Group #0, Counter #11
43	PCLK_TCPWM0_CLOCKS12	TCPWM0 Group #0, Counter #12
44	PCLK_TCPWM0_CLOCKS13	TCPWM0 Group #0, Counter #13
45	PCLK_TCPWM0_CLOCKS14	TCPWM0 Group #0, Counter #14
46	PCLK_TCPWM0_CLOCKS15	TCPWM0 Group #0, Counter #15
47	PCLK_TCPWM0_CLOCKS16	TCPWM0 Group #0, Counter #16
48	PCLK_TCPWM0_CLOCKS17	TCPWM0 Group #0, Counter #17
49	PCLK_TCPWM0_CLOCKS18	TCPWM0 Group #0, Counter #18
50	PCLK_TCPWM0_CLOCKS19	TCPWM0 Group #0, Counter #19
51	PCLK_TCPWM0_CLOCKS20	TCPWM0 Group #0, Counter #20
52	PCLK_TCPWM0_CLOCKS21	TCPWM0 Group #0, Counter #21
53	PCLK_TCPWM0_CLOCKS22	TCPWM0 Group #0, Counter #22
54	PCLK_TCPWM0_CLOCKS23	TCPWM0 Group #0, Counter #23
55	PCLK_TCPWM0_CLOCKS24	TCPWM0 Group #0, Counter #24
56	PCLK_TCPWM0_CLOCKS25	TCPWM0 Group #0, Counter #25
57	PCLK_TCPWM0_CLOCKS26	TCPWM0 Group #0, Counter #26
58	PCLK_TCPWM0_CLOCKS27	TCPWM0 Group #0, Counter #27
59	PCLK_TCPWM0_CLOCKS28	TCPWM0 Group #0, Counter #28
60	PCLK_TCPWM0_CLOCKS29	TCPWM0 Group #0, Counter #29
61	PCLK_TCPWM0_CLOCKS30	TCPWM0 Group #0, Counter #30
62	PCLK_TCPWM0_CLOCKS31	TCPWM0 Group #0, Counter #31
63	PCLK_TCPWM0_CLOCKS32	TCPWM0 Group #0, Counter #32
64	PCLK_TCPWM0_CLOCKS33	TCPWM0 Group #0, Counter #33
65	PCLK_TCPWM0_CLOCKS34	TCPWM0 Group #0, Counter #34
66	PCLK_TCPWM0_CLOCKS35	TCPWM0 Group #0, Counter #35
67	PCLK_TCPWM0_CLOCKS36	TCPWM0 Group #0, Counter #36
68	PCLK_TCPWM0_CLOCKS37	TCPWM0 Group #0, Counter #37
69	PCLK_TCPWM0_CLOCKS38	TCPWM0 Group #0, Counter #38
70	PCLK_TCPWM0_CLOCKS39	TCPWM0 Group #0, Counter #39
71	PCLK_TCPWM0_CLOCKS40	TCPWM0 Group #0, Counter #40
72	PCLK_TCPWM0_CLOCKS41	TCPWM0 Group #0, Counter #41

(table continues...)

Table 23 (continued) Peripheral clock assignments

Output	Destination	Description
73	PCLK_TCPWM0_CLOCKS42	TCPWM0 Group #0, Counter #42
74	PCLK_TCPWM0_CLOCKS43	TCPWM0 Group #0, Counter #43
75	PCLK_TCPWM0_CLOCKS44	TCPWM0 Group #0, Counter #44
76	PCLK_TCPWM0_CLOCKS45	TCPWM0 Group #0, Counter #45
77	PCLK_TCPWM0_CLOCKS46	TCPWM0 Group #0, Counter #46
78	PCLK_TCPWM0_CLOCKS47	TCPWM0 Group #0, Counter #47
79	PCLK_TCPWM0_CLOCKS48	TCPWM0 Group #0, Counter #48
80	PCLK_TCPWM0_CLOCKS49	TCPWM0 Group #0, Counter #49
81	PCLK_TCPWM0_CLOCKS50	TCPWM0 Group #0, Counter #50
82	PCLK_TCPWM0_CLOCKS51	TCPWM0 Group #0, Counter #51
83	PCLK_TCPWM0_CLOCKS52	TCPWM0 Group #0, Counter #52
84	PCLK_TCPWM0_CLOCKS53	TCPWM0 Group #0, Counter #53
85	PCLK_TCPWM0_CLOCKS54	TCPWM0 Group #0, Counter #54
86	PCLK_TCPWM0_CLOCKS55	TCPWM0 Group #0, Counter #55
87	PCLK_TCPWM0_CLOCKS56	TCPWM0 Group #0, Counter #56
88	PCLK_TCPWM0_CLOCKS57	TCPWM0 Group #0, Counter #57
89	PCLK_TCPWM0_CLOCKS58	TCPWM0 Group #0, Counter #58
90	PCLK_TCPWM0_CLOCKS59	TCPWM0 Group #0, Counter #59
91	PCLK_TCPWM0_CLOCKS60	TCPWM0 Group #0, Counter #60
92	PCLK_TCPWM0_CLOCKS61	TCPWM0 Group #0, Counter #61
93	PCLK_TCPWM0_CLOCKS62	TCPWM0 Group #0, Counter #62
94	PCLK_TCPWM0_CLOCKS256	TCPWM0 Group #1, Counter #0
95	PCLK_TCPWM0_CLOCKS257	TCPWM0 Group #1, Counter #1
96	PCLK_TCPWM0_CLOCKS258	TCPWM0 Group #1, Counter #2
97	PCLK_TCPWM0_CLOCKS259	TCPWM0 Group #1, Counter #3
98	PCLK_TCPWM0_CLOCKS260	TCPWM0 Group #1, Counter #4
99	PCLK_TCPWM0_CLOCKS261	TCPWM0 Group #1, Counter #5
100	PCLK_TCPWM0_CLOCKS262	TCPWM0 Group #1, Counter #6
101	PCLK_TCPWM0_CLOCKS263	TCPWM0 Group #1, Counter #7
102	PCLK_TCPWM0_CLOCKS264	TCPWM0 Group #1, Counter #8
103	PCLK_TCPWM0_CLOCKS265	TCPWM0 Group #1, Counter #9
104	PCLK_TCPWM0_CLOCKS266	TCPWM0 Group #1, Counter #10
105	PCLK_TCPWM0_CLOCKS267	TCPWM0 Group #1, Counter #11

(table continues...)

Table 23 (continued) Peripheral clock assignments

Output	Destination	Description
106	PCLK_TCPWM0_CLOCKS512	TCPWM0 Group #2, Counter #0
107	PCLK_TCPWM0_CLOCKS513	TCPWM0 Group #2, Counter #1
108	PCLK_TCPWM0_CLOCKS514	TCPWM0 Group #2, Counter #2
109	PCLK_TCPWM0_CLOCKS515	TCPWM0 Group #2, Counter #3

21 Faults

Table 24 Fault assignments

Fault	Source	Description
0	CPUSS_MPU_VIO_0	CM0+ SMPU violation DATA0[31:0]: Violating address. DATA1[0]: User read. DATA1[1]: User write. DATA1[2]: User execute. DATA1[3]: Privileged read. DATA1[4]: Privileged write. DATA1[5]: Privileged execute. DATA1[6]: Non-secure. DATA1[11:8]: Master identifier. DATA1[15:12]: Protection context identifier. DATA1[31]: '0' MPU violation; '1': SMPU violation.
1	CPUSS_MPU_VIO_1	CRYPTO SMPU violation. See CPUSS_MPU_VIO_0 description.
2	CPUSS_MPU_VIO_2	P-DMA0 MPU/SMPU violation. See CPUSS_MPU_VIO_0 description.
3	CPUSS_MPU_VIO_3	P-DMA1 MPU/SMPU violation. See CPUSS_MPU_VIO_0 description.
4	CPUSS_MPU_VIO_4	M-DMA0 MPU/SMPU violation. See CPUSS_MPU_VIO_0 description.
15	CPUSS_MPU_VIO_15	Test Controller MPU/SMPU violation. See CPUSS_MPU_VIO_0 description.
16	CPUSS_MPU_VIO_16	CM4 system bus AHB-Lite interface MPU violation. See CPUSS_MPU_VIO_0 description
17	CPUSS_MPU_VIO_17	CM4 code bus AHB-Lite interface MPU violation for non flash controller accesses. See CPUSS_MPU_VIO_0 description.
18	CPUSS_MPU_VIO_18	CM4 code bus AHB-Lite interface MPU violation for flash controller accesses. See CPUSS_MPU_VIO_0 description.
26	PERI_PERI_C_ECC	Peripheral protection SRAM correctable ECC violation DATA0[10:0]: Violating address. DATA1[7:0]: Syndrome of SRAM word.
27	PERI_PERI_NC_ECC	Peripheral protection SRAM non-correctable ECC violation

(table continues...)

Table 24 (continued) Fault assignments

Fault	Source	Description
28	PERI_MS_VIO_0	CM0+ Peripheral Master Interface PPU violation DATA0[31:0]: Violating address. DATA1[0]: User read. DATA1[1]: User write. DATA1[2]: User execute. DATA1[3]: Privileged read. DATA1[4]: Privileged write. DATA1[5]: Privileged execute. DATA1[6]: Non-secure. DATA1[11:8]: Master identifier. DATA1[15:12]: Protection context identifier. DATA1[31:28]: "0": master interface, PPU violation, "1": timeout detected, "2": bus error, other: undefined.
29	PERI_MS_VIO_1	CM4 Peripheral Master Interface PPU violation. See PERI_MS_VIO_0 description.
30	PERI_MS_VIO_2	P-DMA0 Peripheral Master Interface PPU violation. See PERI_MS_VIO_0 description.
31	PERI_MS_VIO_3	P-DMA1 Peripheral Master Interface PPU violation. See PERI_MS_VIO_0 description.
32	PERI_GROUP_VIO_0	Peripheral Group #0 violation. DATA0[31:0]: Violating address. DATA1[0]: User read. DATA1[1]: User write. DATA1[2]: User execute. DATA1[3]: Privileged read. DATA1[4]: Privileged write. DATA1[5]: Privileged execute. DATA1[6]: Non-secure. DATA1[11:8]: Master identifier. DATA1[15:12]: Protection context identifier. DATA1[31:28]: “0”: decoder or peripheral bus error, other: undefined.
33	PERI_GROUP_VIO_1	Peripheral Group #1 violation. See PERI_GROUP_VIO_0 description.
34	PERI_GROUP_VIO_2	Peripheral Group #2 violation. See PERI_GROUP_VIO_0 description.
35	PERI_GROUP_VIO_3	Peripheral Group #3 violation. See PERI_GROUP_VIO_0 description.

(table continues...)

Table 24 (continued) Fault assignments

Fault	Source	Description
37	PERI_GROUP_VIO_5	Peripheral Group #5 violation. See PERI_GROUP_VIO_0 description.
38	PERI_GROUP_VIO_6	Peripheral Group #6 violation. See PERI_GROUP_VIO_0 description.
41	PERI_GROUP_VIO_9	Peripheral Group #9 violation. See PERI_GROUP_VIO_0 description.
48	CPUSS_FLASHC_MAIN_BUS_ERRO R	Flash controller main flash bus error FAULT_DATA0[26:0]: Violating address. Append 5'b00010 as most significant bits to derive 32-bit system address. FAULT_DATA1[11:8]: Master identifier.
49	CPUSS_FLASHC_MAIN_C_ECC	Flash controller main flash correctable ECC violation DATA[26:0]: Violating address. Append 5'b00010 as most significant bits to derive 32-bit system address. DATA1[7:0]: Syndrome of 64-bit word (at address offset 0x00). DATA1[15:8]: Syndrome of 64-bit word (at address offset 0x08). DATA1[23:16]: Syndrome of 64-bit word (at address offset 0x10). DATA1[31:24]: Syndrome of 64-bit word (at address offset 0x18).
50	CPUSS_FLASHC_MAIN_NC_ECC	Flash controller main flash non-correctable ECC violation. See CPUSS_FLASHC_MAIN_C_ECC description.
51	CPUSS_FLASHC_WORK_BUS_ERRO R	Flash controller work-flash bus error. See CPUSS_FLASHC_MAIN_BUS_ERR description.
52	CPUSS_FLASHC_WORK_C_ECC	Flash controller work flash correctable ECC violation. DATA0[26:0]: Violating address. Append 5'b00010 as most significant bits to derive 32-bit system address. DATA1[6:0]: Syndrome of 32-bit word.
53	CPUSS_FLASHC_WORK_NC_ECC	Flash controller work-flash non-correctable ECC violation. See CPUSS_FLASHC_WORK_C_ECC description.
54	CPUSS_FLASHC_CM0_CA_C_ECC	Flash controller CM0+ cache correctable ECC violation. DATA0[26:0]: Violating address. DATA1[6:0]: Syndrome of 32-bit SRAM word (at address offset 0x0). DATA1[14:8]: Syndrome of 32-bit SRAM word (at address offset 0x4). DATA1[22:16]: Syndrome of 32-bit SRAM word (at address offset 0x8). DATA1[30:24]: Syndrome of 32-bit SRAM word (at address offset 0xc).
55	CPUSS_FLASHC_CM0_CA_NC_ECC	Flash controller CM0+ cache non-correctable ECC violation. See CPUSS_FLASHC_CM0_CA_C_ECC description.

(table continues...)

Table 24 (continued) Fault assignments

Fault	Source	Description
56	CPUSS_FLASHC_CM4_CA_C_ECC	Flash controller CM4 cache correctable ECC violation. See CPUSS_FLASHC_CM0_CA_C_ECC description.
57	CPUSS_FLASHC_CM4_CA_NC_ECC	Flash controller CM4 cache non-correctable ECC violation. See CPUSS_FLASHC_CM0_CA_C_ECC description.
58	CPUSS_RAMC0_C_ECC	System memory controller 0 correctable ECC violation: DATA0[31:0]: Violating address. DATA1[6:0]: Syndrome of 32-bit SRAM code word.
59	CPUSS_RAMC0_NC_ECC	System memory controller 0 non-correctable ECC violation. See CPUSS_RAMC0_C_ECC description.
60	CPUSS_RAMC1_C_ECC	System memory controller 1 correctable ECC violation. See CPUSS_RAMC0_C_ECC description.
61	CPUSS_RAMC1_NC_ECC	System memory controller 1 non-correctable ECC violation. See CPUSS_RAMC0_C_ECC description.
64	CPUSS_CRYPTO_C_ECC	Crypto memory correctable ECC violation. DATA0[31:0]: Violating address. DATA1[6:0]: Syndrome of Least Significant 32-bit SRAM. DATA1[14:8]: Syndrome of Most Significant 32-bit SRAM.
65	CPUSS_CRYPTO_NC_ECC	CRYPTO memory non-correctable ECC violation. See CPUSS_CRYPTO_C_ECC description.
70	CPUSS_DW0_C_ECC	P-DMA0 memory correctable ECC violation: DATA0[11:0]: Violating DW SRAM address (word address, assuming byte addressable). DATA1[6:0]: Syndrome of 32-bit SRAM code word.
71	CPUSS_DW0_NC_ECC	P-DMA0 memory non-correctable ECC violation. See CPUSS_DW0_C_ECC description.
72	CPUSS_DW1_C_ECC	P-DMA1 memory correctable ECC violation. See CPUSS_DW0_C_ECC description.
73	CPUSS_DW1_NC_ECC	P-DMA1 memory non-correctable ECC violation. See CPUSS_DW0_C_ECC description.
74	CPUSS_FM_SRAM_C_ECC	Flash code storage SRAM memory correctable ECC violation: DATA0[15:0]: Address location in the eCT Flash SRAM. DATA1[6:0]: Syndrome of 32-bit SRAM word.
75	CPUSS_FM_SRAM_NC_ECC	Flash code storage SRAM memory non-correctable ECC violation: See CPUSS_FM_SRAMC_C_ECC description.

(table continues...)

Table 24 (continued) Fault assignments

Fault	Source	Description
80	CANFD_0_CAN_C_ECC	CAN0 message buffer correctable ECC violation: DATA0[15:0]: Violating address. DATA0[22:16]: ECC violating data[38:32] from MRAM. DATA0[27:24]: Master ID: 0-7 = CAN channel ID within mxttcanfd cluster, 8 = AHB I/F DATA1[31:0]: ECC violating data[31:0] from MRAM.
81	CANFD_0_CAN_NC_ECC	CAN0 message buffer non-correctable ECC violation: DATA0[15:0]: Violating address. DATA0[22:16]: ECC violating data[38:32] from MRAM (not for Address Error). DATA0[27:24]: Master ID: 0-7 = CAN channel ID within mxttcanfd cluster, 8 = AHB I/F DATA0[30]: Write access, only possible for Address Error DATA0[31]: Address Error: a CAN channel did an MRAM access above MRAM_SIZE DATA1[31:0]: ECC violating data[31:0] from MRAM (not for Address Error).
82	CANFD_1_CAN_C_ECC	CAN1 message buffer correctable ECC violation. See CANFD_0_CAN_C_ECC description.
83	CANFD_1_CAN_NC_ECC	CAN1 message buffer non-correctable ECC violation. See CANFD_0_CAN_NC_ECC description.
90	SRSS_FAULT_CSV	Consolidated fault output for clock supervisors. Multiple CSV can detect a violation at the same time. DATA0[15:0]: CLK_HF* root CSV violation flags. DATA0[24]: CLK_REF CSV violation flag (reference clock for CLK_HF CSVs) DATA0[25]: CLK_LF CSV violation flag DATA0[26]: CLK_HVILO CSV violation flag
91	SRSS_FAULT_SSV	Consolidated fault output for supply supervisors. Multiple CSV can detect a violation at the same time. DATA0[0]: BOD on VDDA DATA[1]: OVD on VDDA DATA[16]: LVD/HVD #1 DATA0[17]: LVD/HVD #2
92	SRSS_FAULT_MCWDT0	Fault output for MCWDT0 (all sub-counters) Multiple counters can detect a violation at the same time. DATA0[0]: MCWDT sub counter 0 LOWER_LIMIT DATA0[1]: MCWDT sub counter 0 UPPER_LIMIT DATA0[2]: MCWDT sub counter 1 LOWER_LIMIT DATA0[3]: MCWDT sub counter 1 UPPER_LIMIT

(table continues...)

Table 24 (continued) Fault assignments

Fault	Source	Description
93	SRSS_FAULT_MCWDT1	Fault output for MCWDT1 (all sub-counters). See SRSS_FAULT_MCWDT0 description.

22 Peripheral protection unit fixed structure pairs

Protection pair is a pair PPU structures, a master and a slave structure. The master structure protects the slave structure, and the slave structure protects resources such as peripheral registers, or the peripheral itself.

Table 25 PPU fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
0	PERI_MS_PPU_FX_PERI_MAIN	0x40000000	0x00002000	Peripheral interconnect main
1	PERI_MS_PPU_FX_PERI_SECURE	0x40002000	0x00000004	Peripheral interconnect secure
2	PERI_MS_PPU_FX_PERI_GR0_GROUP	0x40004010	0x00000004	Peripheral group #0 main
3	PERI_MS_PPU_FX_PERI_GR1_GROUP	0x40004030	0x00000004	Peripheral group #1 main
4	PERI_MS_PPU_FX_PERI_GR2_GROUP	0x40004050	0x00000004	Peripheral group #2 main
5	PERI_MS_PPU_FX_PERI_GR3_GROUP	0x40004060	0x00000020	Peripheral group #3 main
6	PERI_MS_PPU_FX_PERI_GR5_GROUP	0x400040A0	0x00000020	Peripheral group #5 main
7	PERI_MS_PPU_FX_PERI_GR6_GROUP	0x400040C0	0x00000020	Peripheral group #6 main
8	PERI_MS_PPU_FX_PERI_GR9_GROUP	0x40004120	0x00000020	Peripheral group #9 main
9	PERI_MS_PPU_FX_PERI_TR	0x40008000	0x00008000	Peripheral trigger multiplexer
10	PERI_MS_PPU_FX_CRYPTO_MAIN	0x40100000	0x00000400	Crypto main
11	PERI_MS_PPU_FX_CRYPTO_CRYPTO	0x40101000	0x00000800	Crypto MMIO (memory mapped I/O)
12	PERI_MS_PPU_FX_CRYPTO_BOOT	0x40102000	0x00000100	Crypto boot
13	PERI_MS_PPU_FX_CRYPTO_KEY0	0x40102100	0x00000004	Crypto Key #0
14	PERI_MS_PPU_FX_CRYPTO_KEY1	0x40102120	0x00000004	Crypto Key #1
15	PERI_MS_PPU_FX_CRYPTO_BUF	0x40108000	0x00002000	Crypto buffer
16	PERI_MS_PPU_FX_CPUSS_CM4	0x40200000	0x00000400	CM4 CPU core
17	PERI_MS_PPU_FX_CPUSS_CM0	0x40201000	0x00001000	CM0+ CPU core

(table continues...)

Table 25 (continued) PPU fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
18	PERI_MS_PPU_FX_CPUSS_BOOT[25]	0x40202000	0x00000200	CPUSS boot
19	PERI_MS_PPU_FX_CPUSS_CM0_INT	0x40208000	0x00000800	CPUSS CM0+ interrupts
20	PERI_MS_PPU_FX_CPUSS_CM4_INT	0x4020A000	0x00000800	CPUSS CM4 interrupts
21	PERI_MS_PPU_FX_FAULT_STRUCT0_MAIN	0x40210000	0x00000100	CPUSS fault structure #0 main
22	PERI_MS_PPU_FX_FAULT_STRUCT1_MAIN	0x40210100	0x00000100	CPUSS fault structure #1 main
23	PERI_MS_PPU_FX_FAULT_STRUCT2_MAIN	0x40210200	0x00000100	CPUSS fault structure #2 main
24	PERI_MS_PPU_FX_FAULT_STRUCT3_MAIN	0x40210300	0x00000100	CPUSS fault structure #3 main
25	PERI_MS_PPU_FX_IPC_STRUCT0_IPC	0x40220000	0x00000020	CPUSS IPC structure #0
26	PERI_MS_PPU_FX_IPC_STRUCT1_IPC	0x40220020	0x00000020	CPUSS IPC structure #1
27	PERI_MS_PPU_FX_IPC_STRUCT2_IPC	0x40220040	0x00000020	CPUSS IPC structure #2
28	PERI_MS_PPU_FX_IPC_STRUCT3_IPC	0x40220060	0x00000020	CPUSS IPC structure #3
29	PERI_MS_PPU_FX_IPC_STRUCT4_IPC	0x40220080	0x00000020	CPUSS IPC structure #4
30	PERI_MS_PPU_FX_IPC_STRUCT5_IPC	0x402200A0	0x00000020	CPUSS IPC structure #5
31	PERI_MS_PPU_FX_IPC_STRUCT6_IPC	0x402200C0	0x00000020	CPUSS IPC structure #6
32	PERI_MS_PPU_FX_IPC_STRUCT7_IPC	0x402200E0	0x00000020	CPUSS IPC structure #7
33	PERI_MS_PPU_FX_IPC_INTR_STRUCT0_INTR	0x40221000	0x00000010	CPUSS IPC interrupt structure #0
34	PERI_MS_PPU_FX_IPC_INTR_STRUCT1_INTR	0x40221020	0x00000010	CPUSS IPC interrupt structure #1
35	PERI_MS_PPU_FX_IPC_INTR_STRUCT2_INTR	0x40221040	0x00000010	CPUSS IPC interrupt structure #2
36	PERI_MS_PPU_FX_IPC_INTR_STRUCT3_INTR	0x40221060	0x00000010	CPUSS IPC interrupt structure #3

(table continues...)

[25] Fixed PPU is configured inside the Boot and user is not allowed to change the attributes of this PPU.

Table 25 (continued) PPU fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
37	PERI_MS_PPU_FX_IPC_INTR_STRUCT4_INTR	0x40221080	0x00000010	CPUSS IPC interrupt structure #4
38	PERI_MS_PPU_FX_IPC_INTR_STRUCT5_INTR	0x402210A0	0x00000010	CPUSS IPC interrupt structure #5
39	PERI_MS_PPU_FX_IPC_INTR_STRUCT6_INTR	0x402210C0	0x00000010	CPUSS IPC Interrupt Structure #6
40	PERI_MS_PPU_FX_IPC_INTR_STRUCT7_INTR	0x402210E0	0x00000010	CPUSS IPC Interrupt Structure #7
41	PERI_MS_PPU_FX_PROT_SMPU_MAIN	0x40230000	0x00000040	Peripheral protection SMPU main
42	PERI_MS_PPU_FX_PROT_MPU0_MAIN	0x40234000	0x00000004	Peripheral protection MPU #0 main
43	PERI_MS_PPU_FX_PROT_MPU14_MAIN	0x40237800	0x00000004	Peripheral protection MPU #14 main
44	PERI_MS_PPU_FX_PROT_MPU15_MAIN	0x40237C00	0x00000400	Peripheral protection MPU #15 main
45	PERI_MS_PPU_FX_FLASHC_MAIN	0x40240000	0x00000008	Flash controller main
46	PERI_MS_PPU_FX_FLASHC_CMD	0x40240008	0x00000004	Flash controller command
47	PERI_MS_PPU_FX_FLASHC_DFT	0x40240200	0x00000100	Flash controller tests
48	PERI_MS_PPU_FX_FLASHC_CM0	0x40240400	0x00000080	Flash controller CM0+
49	PERI_MS_PPU_FX_FLASHC_CM4	0x40240480	0x00000080	Flash controller CM4
50	PERI_MS_PPU_FX_FLASHC_CRYPTO	0x40240500	0x00000004	Flash controller Crypto
51	PERI_MS_PPU_FX_FLASHC_DW0	0x40240580	0x00000004	Flash controller P-DMA0
52	PERI_MS_PPU_FX_FLASHC_DW1	0x40240600	0x00000004	Flash controller P-DMA1
53	PERI_MS_PPU_FX_FLASHC_DMAC	0x40240680	0x00000004	Flash controller M-DMA0
54	PERI_MS_PPU_FX_FLASHC_FlashMgmt[25]	0x4024F000	0x00000080	Flash management
55	PERI_MS_PPU_FX_FLASHC_MainSafety	0x4024F400	0x00000008	Flash controller code-flash safety

(table continues...)

25 Fixed PPU is configured inside the Boot and user is not allowed to change the attributes of this PPU.

Table 25 (continued) PPU fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
56	PERI_MS_PPU_FX_FLASHC_WorkSafety	0x4024F500	0x00000004	Flash controller work-flash safety
57	PERI_MS_PPU_FX_SRSS_GENERAL	0x40260000	0x00000400	SRSS general
58	PERI_MS_PPU_FX_SRSS_MAIN	0x40261000	0x00001000	SRSS main
59	PERI_MS_PPU_FX_SRSS_SECURE	0x40262000	0x00002000	SRSS secure
60	PERI_MS_PPU_FX_MCWDT0_CONFIG	0x40268000	0x00000080	MCWDT #0 configuration
61	PERI_MS_PPU_FX_MCWDT1_CONFIG	0x40268100	0x00000080	MCWDT #1 configuration
62	PERI_MS_PPU_FX_MCWDT0_MAIN	0x40268080	0x00000040	MCWDT #0 main
63	PERI_MS_PPU_FX_MCWDT1_MAIN	0x40268180	0x00000040	MCWDT #1 main
64	PERI_MS_PPU_FX_WDT_CONFIG	0x4026C000	0x00000020	System WDT configuration
65	PERI_MS_PPU_FX_WDT_MAIN	0x4026C040	0x00000020	System WDT main
66	PERI_MS_PPU_FX_BACKUP_BACKUP	0x40270000	0x00010000	SRSS backup
67	PERI_MS_PPU_FX_DW0_DW	0x40280000	0x00000100	P-DMA0 main
68	PERI_MS_PPU_FX_DW1_DW	0x40290000	0x00000100	P-DMA1 main
69	PERI_MS_PPU_FX_DW0_DW_CRC	0x40280100	0x00000080	P-DMA0 CRC
70	PERI_MS_PPU_FX_DW1_DW_CRC	0x40290100	0x00000080	P-DMA1 CRC
71	PERI_MS_PPU_FX_DW0_CH_STRUCT0_CH	0x40288000	0x00000040	P-DMA0 Channel #0
72	PERI_MS_PPU_FX_DW0_CH_STRUCT1_CH	0x40288040	0x00000040	P-DMA0 Channel #1
73	PERI_MS_PPU_FX_DW0_CH_STRUCT2_CH	0x40288080	0x00000040	P-DMA0 Channel #2
74	PERI_MS_PPU_FX_DW0_CH_STRUCT3_CH	0x402880C0	0x00000040	P-DMA0 Channel #3

(table continues...)

Table 25 (continued) PPU fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
75	PERI_MS_PPU_FX_DW0_CH_STRUCT4_CH	0x40288100	0x00000040	P-DMA0 Channel #4
76	PERI_MS_PPU_FX_DW0_CH_STRUCT5_CH	0x40288140	0x00000040	P-DMA0 Channel #5
77	PERI_MS_PPU_FX_DW0_CH_STRUCT6_CH	0x40288180	0x00000040	P-DMA0 Channel #6
78	PERI_MS_PPU_FX_DW0_CH_STRUCT7_CH	0x402881C0	0x00000040	P-DMA0 Channel #7
79	PERI_MS_PPU_FX_DW0_CH_STRUCT8_CH	0x40288200	0x00000040	P-DMA0 Channel #8
80	PERI_MS_PPU_FX_DW0_CH_STRUCT9_CH	0x40288240	0x00000040	P-DMA0 Channel #9
81	PERI_MS_PPU_FX_DW0_CH_STRUCT10_CH	0x40288280	0x00000040	P-DMA0 Channel #10
82	PERI_MS_PPU_FX_DW0_CH_STRUCT11_CH	0x402882C0	0x00000040	P-DMA0 Channel #11
83	PERI_MS_PPU_FX_DW0_CH_STRUCT12_CH	0x40288300	0x00000040	P-DMA0 Channel #12
84	PERI_MS_PPU_FX_DW0_CH_STRUCT13_CH	0x40288340	0x00000040	P-DMA0 Channel #13
85	PERI_MS_PPU_FX_DW0_CH_STRUCT14_CH	0x40288380	0x00000040	P-DMA0 Channel #14
86	PERI_MS_PPU_FX_DW0_CH_STRUCT15_CH	0x402883C0	0x00000040	P-DMA0 Channel #15
87	PERI_MS_PPU_FX_DW0_CH_STRUCT16_CH	0x40288400	0x00000040	P-DMA0 Channel #16
88	PERI_MS_PPU_FX_DW0_CH_STRUCT17_CH	0x40288440	0x00000040	P-DMA0 Channel #17
89	PERI_MS_PPU_FX_DW0_CH_STRUCT18_CH	0x40288480	0x00000040	P-DMA0 Channel #18
90	PERI_MS_PPU_FX_DW0_CH_STRUCT19_CH	0x402884C0	0x00000040	P-DMA0 Channel #19
91	PERI_MS_PPU_FX_DW0_CH_STRUCT20_CH	0x40288500	0x00000040	P-DMA0 Channel #20
92	PERI_MS_PPU_FX_DW0_CH_STRUCT21_CH	0x40288540	0x00000040	P-DMA0 Channel #21
93	PERI_MS_PPU_FX_DW0_CH_STRUCT22_CH	0x40288580	0x00000040	P-DMA0 Channel #22

(table continues...)

Table 25 (continued) PPU fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
94	PERI_MS_PPU_FX_DW0_CH_STRUCT23_CH	0x402885C0	0x00000040	P-DMA0 Channel #23
95	PERI_MS_PPU_FX_DW0_CH_STRUCT24_CH	0x40288600	0x00000040	P-DMA0 Channel #24
96	PERI_MS_PPU_FX_DW0_CH_STRUCT25_CH	0x40288640	0x00000040	P-DMA0 Channel #25
97	PERI_MS_PPU_FX_DW0_CH_STRUCT26_CH	0x40288680	0x00000040	P-DMA0 Channel #26
98	PERI_MS_PPU_FX_DW0_CH_STRUCT27_CH	0x402886C0	0x00000040	P-DMA0 Channel #27
99	PERI_MS_PPU_FX_DW0_CH_STRUCT28_CH	0x40288700	0x00000040	P-DMA0 Channel #28
100	PERI_MS_PPU_FX_DW0_CH_STRUCT29_CH	0x40288740	0x00000040	P-DMA0 Channel #29
101	PERI_MS_PPU_FX_DW0_CH_STRUCT30_CH	0x40288780	0x00000040	P-DMA0 Channel #30
102	PERI_MS_PPU_FX_DW0_CH_STRUCT31_CH	0x402887C0	0x00000040	P-DMA0 Channel #31
103	PERI_MS_PPU_FX_DW0_CH_STRUCT32_CH	0x40288800	0x00000040	P-DMA0 Channel #32
104	PERI_MS_PPU_FX_DW0_CH_STRUCT33_CH	0x40288840	0x00000040	P-DMA0 Channel #33
105	PERI_MS_PPU_FX_DW0_CH_STRUCT34_CH	0x40288880	0x00000040	P-DMA0 Channel #34
106	PERI_MS_PPU_FX_DW0_CH_STRUCT35_CH	0x402888C0	0x00000040	P-DMA0 Channel #35
107	PERI_MS_PPU_FX_DW0_CH_STRUCT36_CH	0x40288900	0x00000040	P-DMA0 Channel #36
108	PERI_MS_PPU_FX_DW0_CH_STRUCT37_CH	0x40288940	0x00000040	P-DMA0 Channel #37
109	PERI_MS_PPU_FX_DW0_CH_STRUCT38_CH	0x40288980	0x00000040	P-DMA0 Channel #38
110	PERI_MS_PPU_FX_DW0_CH_STRUCT39_CH	0x402889C0	0x00000040	P-DMA0 Channel #39
111	PERI_MS_PPU_FX_DW0_CH_STRUCT40_CH	0x40288A00	0x00000040	P-DMA0 Channel #40
112	PERI_MS_PPU_FX_DW0_CH_STRUCT41_CH	0x40288A40	0x00000040	P-DMA0 Channel #41

(table continues...)

Table 25 (continued) PPU fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
113	PERI_MS_PPU_FX_DW0_CH_STRUCT42_CH	0x40288A80	0x00000040	P-DMA0 Channel #42
114	PERI_MS_PPU_FX_DW0_CH_STRUCT43_CH	0x40288AC0	0x00000040	P-DMA0 Channel #43
115	PERI_MS_PPU_FX_DW0_CH_STRUCT44_CH	0x40288B00	0x00000040	P-DMA0 Channel #44
116	PERI_MS_PPU_FX_DW0_CH_STRUCT45_CH	0x40288B40	0x00000040	P-DMA0 Channel #45
117	PERI_MS_PPU_FX_DW0_CH_STRUCT46_CH	0x40288B80	0x00000040	P-DMA0 Channel #46
118	PERI_MS_PPU_FX_DW0_CH_STRUCT47_CH	0x40288BC0	0x00000040	P-DMA0 Channel #47
119	PERI_MS_PPU_FX_DW0_CH_STRUCT48_CH	0x40288C00	0x00000040	P-DMA0 Channel #48
120	PERI_MS_PPU_FX_DW0_CH_STRUCT49_CH	0x40288C40	0x00000040	P-DMA0 Channel #49
121	PERI_MS_PPU_FX_DW0_CH_STRUCT50_CH	0x40288C80	0x00000040	P-DMA0 Channel #50
122	PERI_MS_PPU_FX_DW0_CH_STRUCT51_CH	0x40288CC0	0x00000040	P-DMA0 Channel #51
123	PERI_MS_PPU_FX_DW0_CH_STRUCT52_CH	0x40288D00	0x00000040	P-DMA0 Channel #52
124	PERI_MS_PPU_FX_DW0_CH_STRUCT53_CH	0x40288D40	0x00000040	P-DMA0 Channel #53
125	PERI_MS_PPU_FX_DW0_CH_STRUCT54_CH	0x40288D80	0x00000040	P-DMA0 Channel #54
126	PERI_MS_PPU_FX_DW0_CH_STRUCT55_CH	0x40288DC0	0x00000040	P-DMA0 Channel #55
127	PERI_MS_PPU_FX_DW0_CH_STRUCT56_CH	0x40288E00	0x00000040	P-DMA0 Channel #56
128	PERI_MS_PPU_FX_DW0_CH_STRUCT57_CH	0x40288E40	0x00000040	P-DMA0 Channel #57
129	PERI_MS_PPU_FX_DW0_CH_STRUCT58_CH	0x40288E80	0x00000040	P-DMA0 Channel #58
130	PERI_MS_PPU_FX_DW0_CH_STRUCT59_CH	0x40288EC0	0x00000040	P-DMA0 Channel #59
131	PERI_MS_PPU_FX_DW0_CH_STRUCT60_CH	0x40288F00	0x00000040	P-DMA0 Channel #60

(table continues...)

Table 25 (continued) PPU fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
132	PERI_MS_PPU_FX_DW0_CH_STRUCT61_CH	0x40288F40	0x00000040	P-DMA0 Channel #61
133	PERI_MS_PPU_FX_DW0_CH_STRUCT62_CH	0x40288F80	0x00000040	P-DMA0 Channel #62
134	PERI_MS_PPU_FX_DW0_CH_STRUCT63_CH	0x40288FC0	0x00000040	P-DMA0 Channel #63
135	PERI_MS_PPU_FX_DW0_CH_STRUCT64_CH	0x40289000	0x00000040	P-DMA0 Channel #64
136	PERI_MS_PPU_FX_DW0_CH_STRUCT65_CH	0x40289040	0x00000040	P-DMA0 Channel #65
137	PERI_MS_PPU_FX_DW0_CH_STRUCT66_CH	0x40289080	0x00000040	P-DMA0 Channel #66
138	PERI_MS_PPU_FX_DW0_CH_STRUCT67_CH	0x402890C0	0x00000040	P-DMA0 Channel #67
139	PERI_MS_PPU_FX_DW0_CH_STRUCT68_CH	0x40289100	0x00000040	P-DMA0 Channel #68
140	PERI_MS_PPU_FX_DW0_CH_STRUCT69_CH	0x40289140	0x00000040	P-DMA0 Channel #69
141	PERI_MS_PPU_FX_DW0_CH_STRUCT70_CH	0x40289180	0x00000040	P-DMA0 Channel #70
142	PERI_MS_PPU_FX_DW0_CH_STRUCT71_CH	0x402891C0	0x00000040	P-DMA0 Channel #71
143	PERI_MS_PPU_FX_DW0_CH_STRUCT72_CH	0x40289200	0x00000040	P-DMA0 Channel #72
144	PERI_MS_PPU_FX_DW0_CH_STRUCT73_CH	0x40289240	0x00000040	P-DMA0 Channel #73
145	PERI_MS_PPU_FX_DW0_CH_STRUCT74_CH	0x40289280	0x00000040	P-DMA0 Channel #74
146	PERI_MS_PPU_FX_DW0_CH_STRUCT75_CH	0x402892C0	0x00000040	P-DMA0 Channel #75
147	PERI_MS_PPU_FX_DW0_CH_STRUCT76_CH	0x40289300	0x00000040	P-DMA0 Channel #76
148	PERI_MS_PPU_FX_DW0_CH_STRUCT77_CH	0x40289340	0x00000040	P-DMA0 Channel #77
149	PERI_MS_PPU_FX_DW0_CH_STRUCT78_CH	0x40289380	0x00000040	P-DMA0 Channel #78
150	PERI_MS_PPU_FX_DW0_CH_STRUCT79_CH	0x402893C0	0x00000040	P-DMA0 Channel #79

(table continues...)

Table 25 (continued) PPU fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
151	PERI_MS_PPU_FX_DW0_CH_STRUCT80_CH	0x40289400	0x00000040	P-DMA0 Channel #80
152	PERI_MS_PPU_FX_DW0_CH_STRUCT81_CH	0x40289440	0x00000040	P-DMA0 Channel #81
153	PERI_MS_PPU_FX_DW0_CH_STRUCT82_CH	0x40289480	0x00000040	P-DMA0 Channel #82
154	PERI_MS_PPU_FX_DW0_CH_STRUCT83_CH	0x402894C0	0x00000040	P-DMA0 Channel #83
155	PERI_MS_PPU_FX_DW0_CH_STRUCT84_CH	0x40289500	0x00000040	P-DMA0 Channel #84
156	PERI_MS_PPU_FX_DW0_CH_STRUCT85_CH	0x40289540	0x00000040	P-DMA0 Channel #85
157	PERI_MS_PPU_FX_DW0_CH_STRUCT86_CH	0x40289580	0x00000040	P-DMA0 Channel #86
158	PERI_MS_PPU_FX_DW0_CH_STRUCT87_CH	0x402895C0	0x00000040	P-DMA0 Channel #87
159	PERI_MS_PPU_FX_DW0_CH_STRUCT88_CH	0x40289600	0x00000040	P-DMA0 Channel #88
Pair no.	**PPU fixed structure pair**	**Address**	**Size**	**Description**
160	PERI_MS_PPU_FX_DW1_CH_STRUCT0_CH	0x40298000	0x00000040	P-DMA1 Channel #0
161	PERI_MS_PPU_FX_DW1_CH_STRUCT1_CH	0x40298040	0x00000040	P-DMA1 Channel #1
162	PERI_MS_PPU_FX_DW1_CH_STRUCT2_CH	0x40298080	0x00000040	P-DMA1 Channel #2
163	PERI_MS_PPU_FX_DW1_CH_STRUCT3_CH	0x402980C0	0x00000040	P-DMA1 Channel #3
164	PERI_MS_PPU_FX_DW1_CH_STRUCT4_CH	0x40298100	0x00000040	P-DMA1 Channel #4
165	PERI_MS_PPU_FX_DW1_CH_STRUCT5_CH	0x40298140	0x00000040	P-DMA1 Channel #5
166	PERI_MS_PPU_FX_DW1_CH_STRUCT6_CH	0x40298180	0x00000040	P-DMA1 Channel #6
167	PERI_MS_PPU_FX_DW1_CH_STRUCT7_CH	0x402981C0	0x00000040	P-DMA1 Channel #7
168	PERI_MS_PPU_FX_DW1_CH_STRUCT8_CH	0x40298200	0x00000040	P-DMA1 Channel #8
169	PERI_MS_PPU_FX_DW1_CH_STRUCT9_CH	0x40298240	0x00000040	P-DMA1 Channel #9

(table continues...)

22 Peripheral protection unit fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
170	PERI_MS_PPU_FX_DW1_CH_STRUCT10_CH	0x40298280	0x00000040	P-DMA1 Channel #10
171	PERI_MS_PPU_FX_DW1_CH_STRUCT11_CH	0x402982C0	0x00000040	P-DMA1 Channel #11
172	PERI_MS_PPU_FX_DW1_CH_STRUCT12_CH	0x40298300	0x00000040	P-DMA1 Channel #12
173	PERI_MS_PPU_FX_DW1_CH_STRUCT13_CH	0x40298340	0x00000040	P-DMA1 Channel #13
174	PERI_MS_PPU_FX_DW1_CH_STRUCT14_CH	0x40298380	0x00000040	P-DMA1 Channel #14
175	PERI_MS_PPU_FX_DW1_CH_STRUCT15_CH	0x402983C0	0x00000040	P-DMA1 Channel #15
176	PERI_MS_PPU_FX_DW1_CH_STRUCT16_CH	0x40298400	0x00000040	P-DMA1 Channel #16
177	PERI_MS_PPU_FX_DW1_CH_STRUCT17_CH	0x40298440	0x00000040	P-DMA1 Channel #17
178	PERI_MS_PPU_FX_DW1_CH_STRUCT18_CH	0x40298480	0x00000040	P-DMA1 Channel #18
179	PERI_MS_PPU_FX_DW1_CH_STRUCT19_CH	0x402984C0	0x00000040	P-DMA1 Channel #19
180	PERI_MS_PPU_FX_DW1_CH_STRUCT20_CH	0x40298500	0x00000040	P-DMA1 Channel #20
181	PERI_MS_PPU_FX_DW1_CH_STRUCT21_CH	0x40298540	0x00000040	P-DMA1 Channel #21
182	PERI_MS_PPU_FX_DW1_CH_STRUCT22_CH	0x40298580	0x00000040	P-DMA1 Channel #22
183	PERI_MS_PPU_FX_DW1_CH_STRUCT23_CH	0x402985C0	0x00000040	P-DMA1 Channel #23
184	PERI_MS_PPU_FX_DW1_CH_STRUCT24_CH	0x40298600	0x00000040	P-DMA1 Channel #24
185	PERI_MS_PPU_FX_DW1_CH_STRUCT25_CH	0x40298640	0x00000040	P-DMA1 Channel #25
186	PERI_MS_PPU_FX_DW1_CH_STRUCT26_CH	0x40298680	0x00000040	P-DMA1 Channel #26
187	PERI_MS_PPU_FX_DW1_CH_STRUCT27_CH	0x402986C0	0x00000040	P-DMA1 Channel #27
188	PERI_MS_PPU_FX_DW1_CH_STRUCT28_CH	0x40298700	0x00000040	P-DMA1 Channel #28
189	PERI_MS_PPU_FX_DW1_CH_STRUCT29_CH	0x40298740	0x00000040	P-DMA1 Channel #29

(table continues...)

22 Peripheral protection unit fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
190	PERI_MS_PPU_FX_DW1_CH_STRUCT30_CH	0x40298780	0x00000040	P-DMA1 Channel #30
191	PERI_MS_PPU_FX_DW1_CH_STRUCT31_CH	0x402987C0	0x00000040	P-DMA1 Channel #31
192	PERI_MS_PPU_FX_DW1_CH_STRUCT32_CH	0x40298800	0x00000040	P-DMA1 Channel #32
193	PERI_MS_PPU_FX_DMAC_TOP	0x402A0000	0x00000010	M-DMA0 main
194	PERI_MS_PPU_FX_DMAC_CH0_CH	0x402A1000	0x00000100	M-DMA0 Channel #0
195	PERI_MS_PPU_FX_DMAC_CH1_CH	0x402A1100	0x00000100	M-DMA0 Channel #1
196	PERI_MS_PPU_FX_DMAC_CH2_CH	0x402A1200	0x00000100	M-DMA0 Channel #2
197	PERI_MS_PPU_FX_DMAC_CH3_CH	0x402A1300	0x00000100	M-DMA0 Channel #3
198	PERI_MS_PPU_FX_EFUSE_CTL	0x402C0000	0x00000200	EFUSE control
199	PERI_MS_PPU_FX_EFUSE_DATA	0x402C0800	0x00000200	EFUSE data
200	PERI_MS_PPU_FX_BIST	0x402F0000	0x00001000	Built-in self test
201	PERI_MS_PPU_FX_HSIOM_PRT0_PRT	0x40300000	0x00000008	HSIOm Port #0
202	PERI_MS_PPU_FX_HSIOM_PRT1_PRT	0x40300010	0x00000008	HSIOm Port #1
203	PERI_MS_PPU_FX_HSIOM_PRT2_PRT	0x40300020	0x00000008	HSIOm Port #2
204	PERI_MS_PPU_FX_HSIOM_PRT3_PRT	0x40300030	0x00000008	HSIOm Port #3
205	PERI_MS_PPU_FX_HSIOM_PRT4_PRT	0x40300040	0x00000008	HSIOm Port #4
206	PERI_MS_PPU_FX_HSIOM_PRT5_PRT	0x40300050	0x00000008	HSIOm Port #5
207	PERI_MS_PPU_FX_HSIOM_PRT6_PRT	0x40300060	0x00000008	HSIOm Port #6
208	PERI_MS_PPU_FX_HSIOM_PRT7_PRT	0x40300070	0x00000008	HSIOm Port #7
209	PERI_MS_PPU_FX_HSIOM_PRT8_PRT	0x40300080	0x00000008	HSIOm Port #8

(table continues...)

22 Peripheral protection unit fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
210	PERI_MS_PPU_FX_HSIOM_PRT9_PRT	0x40300090	0x00000008	HSIOm Port #9
211	PERI_MS_PPU_FX_HSIOM_PRT10_PRT	0x403000A0	0x00000008	HSIOm Port #10
212	PERI_MS_PPU_FX_HSIOM_PRT11_PRT	0x403000B0	0x00000008	HSIOm Port #11
213	PERI_MS_PPU_FX_HSIOM_PRT12_PRT	0x403000C0	0x00000008	HSIOm Port #12
214	PERI_MS_PPU_FX_HSIOM_PRT13_PRT	0x403000D0	0x00000008	HSIOm Port #13
215	PERI_MS_PPU_FX_HSIOM_PRT14_PRT	0x403000E0	0x00000008	HSIOm Port #14
216	PERI_MS_PPU_FX_HSIOM_PRT15_PRT	0x403000F0	0x00000008	HSIOm Port #15
217	PERI_MS_PPU_FX_HSIOM_PRT16_PRT	0x40300100	0x00000008	HSIOm Port #16
218	PERI_MS_PPU_FX_HSIOM_PRT17_PRT	0x40300110	0x00000008	HSIOm Port #17
219	PERI_MS_PPU_FX_HSIOM_PRT18_PRT	0x40300120	0x00000008	HSIOm Port #18
220	PERI_MS_PPU_FX_HSIOM_PRT19_PRT	0x40300130	0x00000008	HSIOm Port #19
221	PERI_MS_PPU_FX_HSIOM_PRT20_PRT	0x40300140	0x00000008	HSIOm Port #20
222	PERI_MS_PPU_FX_HSIOM_PRT21_PRT	0x40300150	0x00000008	HSIOm Port #21
223	PERI_MS_PPU_FX_HSIOM_PRT22_PRT	0x40300160	0x00000008	HSIOm Port #22
224	PERI_MS_PPU_FX_HSIOM_PRT23_PRT	0x40300170	0x00000008	HSIOm Port #23
225	PERI_MS_PPU_FX_HSIOM_AMUX	0x40302000	0x00000010	HSIOm Analog multiplexer
226	PERI_MS_PPU_FX_HSIOM_MON	0x40302200	0x00000010	HSIOm monitor
227	PERI_MS_PPU_FX_HSIOM_ALTJTAG	0x40302240	0x00000004	HSIOm Alternate JTAG
228	PERI_MS_PPU_FX_GPIO_PRT0_PRT	0x40310000	0x00000040	GPIO_ENH Port #0
229	PERI_MS_PPU_FX_GPIO_PRT1_PRT	0x40310080	0x00000040	GPIO_STD Port #1

(table continues...)

22 Peripheral protection unit fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
230	PERI_MS_PPU_FX_GPIO_PRT2_PRT	0x40310100	0x00000040	GPIO_STD Port #2
231	PERI_MS_PPU_FX_GPIO_PRT3_PRT	0x40310180	0x00000040	GPIO_STD Port #3
232	PERI_MS_PPU_FX_GPIO_PRT4_PRT	0x40310200	0x00000040	GPIO_STD Port #4
233	PERI_MS_PPU_FX_GPIO_PRT5_PRT	0x40310280	0x00000040	GPIO_STD Port #5
234	PERI_MS_PPU_FX_GPIO_PRT6_PRT	0x40310300	0x00000040	GPIO_STD Port #6
235	PERI_MS_PPU_FX_GPIO_PRT7_PRT	0x40310380	0x00000040	GPIO_STD Port #7
236	PERI_MS_PPU_FX_GPIO_PRT8_PRT	0x40310400	0x00000040	GPIO_STD Port #8
237	PERI_MS_PPU_FX_GPIO_PRT9_PRT	0x40310480	0x00000040	GPIO_STD Port #9
238	PERI_MS_PPU_FX_GPIO_PRT10_PRT	0x40310500	0x00000040	GPIO_STD Port #10
239	PERI_MS_PPU_FX_GPIO_PRT11_PRT	0x40310580	0x00000040	GPIO_STD Port #11
240	PERI_MS_PPU_FX_GPIO_PRT12_PRT	0x40310600	0x00000040	GPIO_STD Port #12
241	PERI_MS_PPU_FX_GPIO_PRT13_PRT	0x40310680	0x00000040	GPIO_STD Port #13
242	PERI_MS_PPU_FX_GPIO_PRT14_PRT	0x40310700	0x00000040	GPIO_STD Port #14
243	PERI_MS_PPU_FX_GPIO_PRT15_PRT	0x40310780	0x00000040	GPIO_STD Port #15
244	PERI_MS_PPU_FX_GPIO_PRT16_PRT	0x40310800	0x00000040	GPIO_STD Port #16
245	PERI_MS_PPU_FX_GPIO_PRT17_PRT	0x40310880	0x00000040	GPIO_STD Port #17
246	PERI_MS_PPU_FX_GPIO_PRT18_PRT	0x40310900	0x00000040	GPIO_STD Port #18
247	PERI_MS_PPU_FX_GPIO_PRT19_PRT	0x40310980	0x00000040	GPIO_STD Port #19
248	PERI_MS_PPU_FX_GPIO_PRT20_PRT	0x40310A00	0x00000040	GPIO_STD Port #20
249	PERI_MS_PPU_FX_GPIO_PRT21_PRT	0x40310A80	0x00000040	GPIO_STD Port #21

(table continues...)

22 Peripheral protection unit fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
250	PERI_MS_PPU_FX_GPIO_PRT22_PRT	0x40310B00	0x00000040	GPIO_STD Port #22
251	PERI_MS_PPU_FX_GPIO_PRT23_PRT	0x40310B80	0x00000040	GPIO_STD Port #23
252	PERI_MS_PPU_FX_GPIO_PRT0_CFG	0x40310040	0x00000020	GPIO_ENH Port #0 configuration
253	PERI_MS_PPU_FX_GPIO_PRT1_CFG	0x403100C0	0x00000020	GPIO_STD Port #1 configuration
254	PERI_MS_PPU_FX_GPIO_PRT2_CFG	0x40310140	0x00000020	GPIO_STD Port #2 configuration
255	PERI_MS_PPU_FX_GPIO_PRT3_CFG	0x403101C0	0x00000020	GPIO_STD Port #3 configuration
256	PERI_MS_PPU_FX_GPIO_PRT4_CFG	0x40310240	0x00000020	GPIO_STD Port #4 configuration
257	PERI_MS_PPU_FX_GPIO_PRT5_CFG	0x403102C0	0x00000020	GPIO_STD Port #5 configuration
258	PERI_MS_PPU_FX_GPIO_PRT6_CFG	0x40310340	0x00000020	GPIO_STD Port #6 configuration
259	PERI_MS_PPU_FX_GPIO_PRT7_CFG	0x403103C0	0x00000020	GPIO_STD Port #7 configuration
260	PERI_MS_PPU_FX_GPIO_PRT8_CFG	0x40310440	0x00000020	GPIO_STD Port #8 configuration
261	PERI_MS_PPU_FX_GPIO_PRT9_CFG	0x403104C0	0x00000020	GPIO_STD Port #9 configuration
262	PERI_MS_PPU_FX_GPIO_PRT10_CFG	0x40310540	0x00000020	GPIO_STD Port #10 configuration
263	PERI_MS_PPU_FX_GPIO_PRT11_CFG	0x403105C0	0x00000020	GPIO_STD Port #11 configuration
264	PERI_MS_PPU_FX_GPIO_PRT12_CFG	0x40310640	0x00000020	GPIO_STD Port #12 configuration
265	PERI_MS_PPU_FX_GPIO_PRT13_CFG	0x403106C0	0x00000020	GPIO_STD Port #13 configuration
266	PERI_MS_PPU_FX_GPIO_PRT14_CFG	0x40310740	0x00000020	GPIO_STD Port #14 configuration
267	PERI_MS_PPU_FX_GPIO_PRT15_CFG	0x403107C0	0x00000020	GPIO_STD Port #15 configuration
268	PERI_MS_PPU_FX_GPIO_PRT16_CFG	0x40310840	0x00000020	GPIO_STD Port #16 configuration
269	PERI_MS_PPU_FX_GPIO_PRT17_CFG	0x403108C0	0x00000020	GPIO_STD Port #17 configuration

(table continues...)

22 Peripheral protection unit fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
270	PERI_MS_PPU_FX_GPIO_PRT18_CFG	0x40310940	0x00000020	GPIO_STD Port #18 configuration
271	PERI_MS_PPU_FX_GPIO_PRT19_CFG	0x403109C0	0x00000020	GPIO_STD Port #19 configuration
272	PERI_MS_PPU_FX_GPIO_PRT20_CFG	0x40310A40	0x00000020	GPIO_STD Port #20 configuration
273	PERI_MS_PPU_FX_GPIO_PRT21_CFG	0x40310AC0	0x00000020	GPIO_STD Port #21 configuration
274	PERI_MS_PPU_FX_GPIO_PRT22_CFG	0x40310B40	0x00000020	GPIO_STD Port #22 configuration
275	PERI_MS_PPU_FX_GPIO_PRT23_CFG	0x40310BC0	0x00000020	GPIO_STD Port #23 configuration
276	PERI_MS_PPU_FX_GPIO_GPIO	0x40314000	0x00000040	GPIO main
277	PERI_MS_PPU_FX_GPIO_TEST	0x40315000	0x00000008	GPIO test
278	PERI_MS_PPU_FX_SMARTIO_PRT12_PRT	0x40320C00	0x00000100	SMART I/O #12
279	PERI_MS_PPU_FX_SMARTIO_PRT13_PRT	0x40320D00	0x00000100	SMART I/O #13
280	PERI_MS_PPU_FX_SMARTIO_PRT14_PRT	0x40320E00	0x00000100	SMART I/O #14
281	PERI_MS_PPU_FX_SMARTIO_PRT15_PRT	0x40320F00	0x00000100	SMART I/O #15
282	PERI_MS_PPU_FX_SMARTIO_PRT17_PRT	0x40321100	0x00000100	SMART I/O #17
283	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT0_CNT	0x40380000	0x00000080	TCPWM0 Group #0, Counter #0
284	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT1_CNT	0x40380080	0x00000080	TCPWM0 Group #0, Counter #1
285	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT2_CNT	0x40380100	0x00000080	TCPWM0 Group #0, Counter #2
286	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT3_CNT	0x40380180	0x00000080	TCPWM0 Group #0, Counter #3
287	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT4_CNT	0x40380200	0x00000080	TCPWM0 Group #0, Counter #4
288	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT5_CNT	0x40380280	0x00000080	TCPWM0 Group #0, Counter #5
289	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT6_CNT	0x40380300	0x00000080	TCPWM0 Group #0, Counter #6

(table continues...)

22 Peripheral protection unit fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
290	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT7_CNT	0x40380380	0x00000080	TCPWM0 Group #0, Counter #7
291	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT8_CNT	0x40380400	0x00000080	TCPWM0 Group #0, Counter #8
292	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT9_CNT	0x40380480	0x00000080	TCPWM0 Group #0, Counter #9
293	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT10_CNT	0x40380500	0x00000080	TCPWM0 Group #0, Counter #10
294	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT11_CNT	0x40380580	0x00000080	TCPWM0 Group #0, Counter #11
295	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT12_CNT	0x40380600	0x00000080	TCPWM0 Group #0, Counter #12
296	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT13_CNT	0x40380680	0x00000080	TCPWM0 Group #0, Counter #13
297	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT14_CNT	0x40380700	0x00000080	TCPWM0 Group #0, Counter #14
298	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT15_CNT	0x40380780	0x00000080	TCPWM0 Group #0, Counter #15
299	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT16_CNT	0x40380800	0x00000080	TCPWM0 Group #0, Counter #16
300	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT17_CNT	0x40380880	0x00000080	TCPWM0 Group #0, Counter #17
301	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT18_CNT	0x40380900	0x00000080	TCPWM0 Group #0, Counter #18
302	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT19_CNT	0x40380980	0x00000080	TCPWM0 Group #0, Counter #19
303	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT20_CNT	0x40380A00	0x00000080	TCPWM0 Group #0, Counter #20
304	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT21_CNT	0x40380A80	0x00000080	TCPWM0 Group #0, Counter #21
305	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT22_CNT	0x40380B00	0x00000080	TCPWM0 Group #0, Counter #22
306	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT23_CNT	0x40380B80	0x00000080	TCPWM0 Group #0, Counter #23
307	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT24_CNT	0x40380C00	0x00000080	TCPWM0 Group #0, Counter #24
308	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT25_CNT	0x40380C80	0x00000080	TCPWM0 Group #0, Counter #25
309	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT26_CNT	0x40380D00	0x00000080	TCPWM0 Group #0, Counter #26

(table continues...)

22 Peripheral protection unit fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
310	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT27_CNT	0x40380D80	0x00000080	TCPWM0 Group #0, Counter #27
311	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT28_CNT	0x40380E00	0x00000080	TCPWM0 Group #0, Counter #28
312	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT29_CNT	0x40380E80	0x00000080	TCPWM0 Group #0, Counter #29
313	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT30_CNT	0x40380F00	0x00000080	TCPWM0 Group #0, Counter #30
314	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT31_CNT	0x40380F80	0x00000080	TCPWM0 Group #0, Counter #31
315	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT32_CNT	0x40381000	0x00000080	TCPWM0 Group #0, Counter #32
316	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT33_CNT	0x40381080	0x00000080	TCPWM0 Group #0, Counter #33
317	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT34_CNT	0x40381100	0x00000080	TCPWM0 Group #0, Counter #34
318	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT35_CNT	0x40381180	0x00000080	TCPWM0 Group #0, Counter #35
319	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT36_CNT	0x40381200	0x00000080	TCPWM0 Group #0, Counter #36
320	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT37_CNT	0x40381280	0x00000080	TCPWM0 Group #0, Counter #37
321	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT38_CNT	0x40381300	0x00000080	TCPWM0 Group #0, Counter #38
322	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT39_CNT	0x40381380	0x00000080	TCPWM0 Group #0, Counter #39
323	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT40_CNT	0x40381400	0x00000080	TCPWM0 Group #0, Counter #40
324	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT41_CNT	0x40381480	0x00000080	TCPWM0 Group #0, Counter #41
325	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT42_CNT	0x40381500	0x00000080	TCPWM0 Group #0, Counter #42
326	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT43_CNT	0x40381580	0x00000080	TCPWM0 Group #0, Counter #43
327	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT44_CNT	0x40381600	0x00000080	TCPWM0 Group #0, Counter #44
328	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT45_CNT	0x40381680	0x00000080	TCPWM0 Group #0, Counter #45
329	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT46_CNT	0x40381700	0x00000080	TCPWM0 Group #0, Counter #46

(table continues...)

22 Peripheral protection unit fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
330	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT47_CNT	0x40381780	0x00000080	TCPWM0 Group #0, Counter #47
331	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT48_CNT	0x40381800	0x00000080	TCPWM0 Group #0, Counter #48
332	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT49_CNT	0x40381880	0x00000080	TCPWM0 Group #0, Counter #49
333	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT50_CNT	0x40381900	0x00000080	TCPWM0 Group #0, Counter #50
334	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT51_CNT	0x40381980	0x00000080	TCPWM0 Group #0, Counter #51
335	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT52_CNT	0x40381A00	0x00000080	TCPWM0 Group #0, Counter #52
336	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT53_CNT	0x40381A80	0x00000080	TCPWM0 Group #0, Counter #53
337	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT54_CNT	0x40381B00	0x00000080	TCPWM0 Group #0, Counter #54
338	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT55_CNT	0x40381B80	0x00000080	TCPWM0 Group #0, Counter #55
339	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT56_CNT	0x40381C00	0x00000080	TCPWM0 Group #0, Counter #56
340	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT57_CNT	0x40381C80	0x00000080	TCPWM0 Group #0, Counter #57
341	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT58_CNT	0x40381D00	0x00000080	TCPWM0 Group #0, Counter #58
342	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT59_CNT	0x40381D80	0x00000080	TCPWM0 Group #0, Counter #59
343	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT60_CNT	0x40381E00	0x00000080	TCPWM0 Group #0, Counter #60
344	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT61_CNT	0x40381E80	0x00000080	TCPWM0 Group #0, Counter #61
345	PERI_MS_PPU_FX_TCPWM0_GRP0_CNT62_CNT	0x40381F00	0x00000080	TCPWM0 Group #0, Counter #62
346	PERI_MS_PPU_FX_TCPWM0_GRP1_CNT0_CNT	0x40388000	0x00000080	TCPWM0 Group #1, Counter #0
347	PERI_MS_PPU_FX_TCPWM0_GRP1_CNT1_CNT	0x40388080	0x00000080	TCPWM0 Group #1, Counter #1
348	PERI_MS_PPU_FX_TCPWM0_GRP1_CNT2_CNT	0x40388100	0x00000080	TCPWM0 Group #1, Counter #2
349	PERI_MS_PPU_FX_TCPWM0_GRP1_CNT3_CNT	0x40388180	0x00000080	TCPWM0 Group #1, Counter #3

(table continues...)

22 Peripheral protection unit fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
350	PERI_MS_PPU_FX_TCPWM0_GRP1_CNT4_CNT	0x40388200	0x00000080	TCPWM0 Group #1, Counter #4
351	PERI_MS_PPU_FX_TCPWM0_GRP1_CNT5_CNT	0x40388280	0x00000080	TCPWM0 Group #1, Counter #5
352	PERI_MS_PPU_FX_TCPWM0_GRP1_CNT6_CNT	0x40388300	0x00000080	TCPWM0 Group #1, Counter #6
353	PERI_MS_PPU_FX_TCPWM0_GRP1_CNT7_CNT	0x40388380	0x00000080	TCPWM0 Group #1, Counter #7
354	PERI_MS_PPU_FX_TCPWM0_GRP1_CNT8_CNT	0x40388400	0x00000080	TCPWM0 Group #1, Counter #8
355	PERI_MS_PPU_FX_TCPWM0_GRP1_CNT9_CNT	0x40388480	0x00000080	TCPWM0 Group #1, Counter #9
356	PERI_MS_PPU_FX_TCPWM0_GRP1_CNT10_CNT	0x40388500	0x00000080	TCPWM0 Group #1, Counter #10
357	PERI_MS_PPU_FX_TCPWM0_GRP1_CNT11_CNT	0x40388580	0x00000080	TCPWM0 Group #1, Counter #11
358	PERI_MS_PPU_FX_TCPWM0_GRP2_CNT0_CNT	0x40390000	0x00000080	TCPWM0 Group #2, Counter #0
359	PERI_MS_PPU_FX_TCPWM0_GRP2_CNT1_CNT	0x40390080	0x00000080	TCPWM0 Group #2, Counter #1
360	PERI_MS_PPU_FX_TCPWM0_GRP2_CNT2_CNT	0x40390100	0x00000080	TCPWM0 Group #2, Counter #2
361	PERI_MS_PPU_FX_TCPWM0_GRP2_CNT3_CNT	0x40390180	0x00000080	TCPWM0 Group #2, Counter #3
362	PERI_MS_PPU_FX_EVTGEN0	0x403F0000	0x00001000	Event generator #0
372	PERI_MS_PPU_FX_CANFD0_CH0_CH	0x40520000	0x00000200	CANFD0, Channel #0
373	PERI_MS_PPU_FX_CANFD0_CH1_CH	0x40520200	0x00000200	CANFD0, Channel #1
374	PERI_MS_PPU_FX_CANFD0_CH2_CH	0x40520400	0x00000200	CANFD0, Channel #2
375	PERI_MS_PPU_FX_CANFD1_CH0_CH	0x40540000	0x00000200	CANFD1, Channel #0
376	PERI_MS_PPU_FX_CANFD1_CH1_CH	0x40540200	0x00000200	CANFD1, Channel #1
377	PERI_MS_PPU_FX_CANFD1_CH2_CH	0x40540400	0x00000200	CANFD1, Channel #2
378	PERI_MS_PPU_FX_CANFD0_MAIN	0x40521000	0x00000100	CANFD0, main

(table continues...)

22 Peripheral protection unit fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
379	PERI_MS_PPU_FX_CANFD1_MAIN	0x40541000	0x00000100	CANFD1, main
380	PERI_MS_PPU_FX_CANFD0_BUF	0x40530000	0x00010000	CANFD0, buffer
381	PERI_MS_PPU_FX_CANFD1_BUF	0x40550000	0x00010000	CANFD1, buffer
382	PERI_MS_PPU_FX_SCB0	0x40600000	0x00010000	SCB0
383	PERI_MS_PPU_FX_SCB1	0x40610000	0x00010000	SCB1
384	PERI_MS_PPU_FX_SCB2	0x40620000	0x00010000	SCB2
385	PERI_MS_PPU_FX_SCB3	0x40630000	0x00010000	SCB3
386	PERI_MS_PPU_FX_SCB4	0x40640000	0x00010000	SCB4
387	PERI_MS_PPU_FX_SCB5	0x40650000	0x00010000	SCB5
388	PERI_MS_PPU_FX_SCB6	0x40660000	0x00010000	SCB6
389	PERI_MS_PPU_FX_SCB7	0x40670000	0x00010000	SCB7
390	PERI_MS_PPU_FX_PASS0_SAR0_SAR	0x40900000	0x00000400	PASS0, SAR0
391	PERI_MS_PPU_FX_PASS0_SAR1_SAR	0x40901000	0x00000400	PASS0, SAR1
392	PERI_MS_PPU_FX_PASS0_SAR2_SAR	0x40902000	0x00000400	PASS0, SAR2
393	PERI_MS_PPU_FX_PASS0_SAR0_CH0_CH	0x40900800	0x00000040	SAR0, Channel #0
394	PERI_MS_PPU_FX_PASS0_SAR0_CH1_CH	0x40900840	0x00000040	SAR0, Channel #1
395	PERI_MS_PPU_FX_PASS0_SAR0_CH2_CH	0x40900880	0x00000040	SAR0, Channel #2
396	PERI_MS_PPU_FX_PASS0_SAR0_CH3_CH	0x409008C0	0x00000040	SAR0, Channel #3
397	PERI_MS_PPU_FX_PASS0_SAR0_CH4_CH	0x40900900	0x00000040	SAR0, Channel #4
398	PERI_MS_PPU_FX_PASS0_SAR0_CH5_CH	0x40900940	0x00000040	SAR0, Channel #5

(table continues...)

22 Peripheral protection unit fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
399	PERI_MS_PPU_FX_PASS0_SAR0_CH6_CH	0x40900980	0x00000040	SAR0, Channel #6
400	PERI_MS_PPU_FX_PASS0_SAR0_CH7_CH	0x409009C0	0x00000040	SAR0, Channel #7
401	PERI_MS_PPU_FX_PASS0_SAR0_CH8_CH	0x40900A00	0x00000040	SAR0, Channel #8
402	PERI_MS_PPU_FX_PASS0_SAR0_CH9_CH	0x40900A40	0x00000040	SAR0, Channel #9
403	PERI_MS_PPU_FX_PASS0_SAR0_CH10_CH	0x40900A80	0x00000040	SAR0, Channel #10
404	PERI_MS_PPU_FX_PASS0_SAR0_CH11_CH	0x40900AC0	0x00000040	SAR0, Channel #11
405	PERI_MS_PPU_FX_PASS0_SAR0_CH12_CH	0x40900B00	0x00000040	SAR0, Channel #12
406	PERI_MS_PPU_FX_PASS0_SAR0_CH13_CH	0x40900B40	0x00000040	SAR0, Channel #13
407	PERI_MS_PPU_FX_PASS0_SAR0_CH14_CH	0x40900B80	0x00000040	SAR0, Channel #14
408	PERI_MS_PPU_FX_PASS0_SAR0_CH15_CH	0x40900BC0	0x00000040	SAR0, Channel #15
409	PERI_MS_PPU_FX_PASS0_SAR0_CH16_CH	0x40900C00	0x00000040	SAR0, Channel #16
410	PERI_MS_PPU_FX_PASS0_SAR0_CH17_CH	0x40900C40	0x00000040	SAR0, Channel #17
411	PERI_MS_PPU_FX_PASS0_SAR0_CH18_CH	0x40900C80	0x00000040	SAR0, Channel #18
412	PERI_MS_PPU_FX_PASS0_SAR0_CH19_CH	0x40900CC0	0x00000040	SAR0, Channel #19
413	PERI_MS_PPU_FX_PASS0_SAR0_CH20_CH	0x40900D00	0x00000040	SAR0, Channel #20
414	PERI_MS_PPU_FX_PASS0_SAR0_CH21_CH	0x40900D40	0x00000040	SAR0, Channel #21
415	PERI_MS_PPU_FX_PASS0_SAR0_CH22_CH	0x40900D80	0x00000040	SAR0, Channel #22
416	PERI_MS_PPU_FX_PASS0_SAR0_CH23_CH	0x40900DC0	0x00000040	SAR0, Channel #23
417	PERI_MS_PPU_FX_PASS0_SAR1_CH0_CH	0x40901800	0x00000040	SAR1, Channel #0
418	PERI_MS_PPU_FX_PASS0_SAR1_CH1_CH	0x40901840	0x00000040	SAR1, Channel #1

(table continues...)

22 Peripheral protection unit fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
419	PERI_MS_PPU_FX_PASS0_SAR1_CH2_CH	0x40901880	0x00000040	SAR1, Channel #2
420	PERI_MS_PPU_FX_PASS0_SAR1_CH3_CH	0x409018C0	0x00000040	SAR1, Channel #3
421	PERI_MS_PPU_FX_PASS0_SAR1_CH4_CH	0x40901900	0x00000040	SAR1, Channel #4
422	PERI_MS_PPU_FX_PASS0_SAR1_CH5_CH	0x40901940	0x00000040	SAR1, Channel #5
423	PERI_MS_PPU_FX_PASS0_SAR1_CH6_CH	0x40901980	0x00000040	SAR1, Channel #6
424	PERI_MS_PPU_FX_PASS0_SAR1_CH7_CH	0x409019C0	0x00000040	SAR1, Channel #7
425	PERI_MS_PPU_FX_PASS0_SAR1_CH8_CH	0x40901A00	0x00000040	SAR1, Channel #8
426	PERI_MS_PPU_FX_PASS0_SAR1_CH9_CH	0x40901A40	0x00000040	SAR1, Channel #9
427	PERI_MS_PPU_FX_PASS0_SAR1_CH10_CH	0x40901A80	0x00000040	SAR1, Channel #10
428	PERI_MS_PPU_FX_PASS0_SAR1_CH11_CH	0x40901AC0	0x00000040	SAR1, Channel #11
429	PERI_MS_PPU_FX_PASS0_SAR1_CH12_CH	0x40901B00	0x00000040	SAR1, Channel #12
430	PERI_MS_PPU_FX_PASS0_SAR1_CH13_CH	0x40901B40	0x00000040	SAR1, Channel #13
431	PERI_MS_PPU_FX_PASS0_SAR1_CH14_CH	0x40901B80	0x00000040	SAR1, Channel #14
432	PERI_MS_PPU_FX_PASS0_SAR1_CH15_CH	0x40901BC0	0x00000040	SAR1, Channel #15
433	PERI_MS_PPU_FX_PASS0_SAR1_CH16_CH	0x40901C00	0x00000040	SAR1, Channel #16
434	PERI_MS_PPU_FX_PASS0_SAR1_CH17_CH	0x40901C40	0x00000040	SAR1, Channel #17
435	PERI_MS_PPU_FX_PASS0_SAR1_CH18_CH	0x40901C80	0x00000040	SAR1, Channel #18
436	PERI_MS_PPU_FX_PASS0_SAR1_CH19_CH	0x40901CC0	0x00000040	SAR1, Channel #19
437	PERI_MS_PPU_FX_PASS0_SAR1_CH20_CH	0x40901D00	0x00000040	SAR1, Channel #20
438	PERI_MS_PPU_FX_PASS0_SAR1_CH21_CH	0x40901D40	0x00000040	SAR1, Channel #21

(table continues...)

22 Peripheral protection unit fixed structure pairs

Pair no.	PPU fixed structure pair	Address	Size	Description
439	PERI_MS_PPU_FX_PASS0_SAR1_CH22_CH	0x40901D80	0x00000040	SAR1, Channel #22
440	PERI_MS_PPU_FX_PASS0_SAR1_CH23_CH	0x40901DC0	0x00000040	SAR1, Channel #23
441	PERI_MS_PPU_FX_PASS0_SAR1_CH24_CH	0x40901E00	0x00000040	SAR1, Channel #24
442	PERI_MS_PPU_FX_PASS0_SAR1_CH25_CH	0x40901E40	0x00000040	SAR1, Channel #25
443	PERI_MS_PPU_FX_PASS0_SAR1_CH26_CH	0x40901E80	0x00000040	SAR1, Channel #26
444	PERI_MS_PPU_FX_PASS0_SAR1_CH27_CH	0x40901EC0	0x00000040	SAR1, Channel #27
445	PERI_MS_PPU_FX_PASS0_SAR1_CH28_CH	0x40901F00	0x00000040	SAR1, Channel #28
446	PERI_MS_PPU_FX_PASS0_SAR1_CH29_CH	0x40901F40	0x00000040	SAR1, Channel #29
447	PERI_MS_PPU_FX_PASS0_SAR1_CH30_CH	0x40901F80	0x00000040	SAR1, Channel #30
448	PERI_MS_PPU_FX_PASS0_SAR1_CH31_CH	0x40901FC0	0x00000040	SAR1, Channel #31
449	PERI_MS_PPU_FX_PASS0_SAR2_CH0_CH	0x40902800	0x00000040	SAR2, Channel #0
450	PERI_MS_PPU_FX_PASS0_SAR2_CH1_CH	0x40902840	0x00000040	SAR2, Channel #1
451	PERI_MS_PPU_FX_PASS0_SAR2_CH2_CH	0x40902880	0x00000040	SAR2, Channel #2
452	PERI_MS_PPU_FX_PASS0_SAR2_CH3_CH	0x409028C0	0x00000040	SAR2, Channel #3
453	PERI_MS_PPU_FX_PASS0_SAR2_CH4_CH	0x40902900	0x00000040	SAR2, Channel #4
454	PERI_MS_PPU_FX_PASS0_SAR2_CH5_CH	0x40902940	0x00000040	SAR2, Channel #5
455	PERI_MS_PPU_FX_PASS0_SAR2_CH6_CH	0x40902980	0x00000040	SAR2, Channel #6
456	PERI_MS_PPU_FX_PASS0_SAR2_CH7_CH	0x409029C0	0x00000040	SAR2, Channel #7
457	PERI_MS_PPU_FX_PASS0_TOP	0x409F0000	0x00001000	PASS0 SAR main

23 Bus masters

The Arbiter (part of flash controller) performs priority-based arbitration based on the master identifier. Each bus master has a dedicated 4-bit master identifier. This master identifier is used for bus arbitration and IPC functionality.

Table 26 Bus masters for access and protection control

ID No.	Master ID	Description
0	CPUSS_MS_ID_CM0	Master ID for Cortex®-M0+ CPU
1	CPUSS_MS_ID_CRYPTO	Master ID for Crypto
2	CPUSS_MS_ID_DW0	Master ID for P-DMA 0
3	CPUSS_MS_ID_DW1	Master ID for P-DMA 1
4	CPUSS_MS_ID_DMAC	Master ID for M-DMA0
14	CPUSS_MS_ID_CM4	Master ID for Cortex®-M4F CPU
15	CPUSS_MS_ID_TC	Master ID for DAP Tap controller

24 Miscellaneous configuration

Table 27 Miscellaneous configuration for devices

Sl. no.	Configuration	Number/ instances	Description
0	SRSS_NUM_CLKPATH	4	Number of clock paths. One for each of FLL, PLL, Direct and CSV
1	SRSS_NUM_HFROOT	3	Number of CLK_HFs roots present
2	PERI_PC_NR	8	Number of protection contexts
3	PERI_CLOCK_NR	110	Number of programmable clocks (outputs)
4	PERI_DIV_8_NR	32	Number of divide-by-8 clock dividers
5	PERI_DIV_16_NR	16	Number of divide-by-16 clock dividers
6	PERI_DIV_24_5_NR	8	Number of divide-by-24.5 clock dividers
7	CPUSS_CM0P_MPU_NR	8	Number of MPU regions in CM0+
8	CPUSS_CM4_MPU_NR	8	Number of MPU regions in CM4
9	CPUSS_CRYPTO_BUFF_SIZE	2048	Number of 32-bit words in the IP internal memory buffer (to allow for a 256-B, 512-B, 1-KB, 2-KB, 4-KB, 8-KB, 16-KB, and 32-KB memory buffer)
10	CPUSS_FAULT_FAULT_NR	4	Number of fault structures
11	CPUSS_IPC_IPC_NR	8	Number of IPC structures 0 - Reserved for CM0+ access 1 - Reserved for CM4 access 2 - Reserved for DAP access Remaining for user purposes
12	SCBx_EZ_DATA_NR	256	Number of EZ memory bytes. This memory is used in EZ mode, CMD_RESP mode and FIFO mode. ***Note**: Only SCB0 supports CMD_RESP mode*
13	CPUSS_PROT_SMPU_STRUCT_NR	16	Number of SMPU protection structures
14	TCPWM_TR_ONE_CNT_NR	3	Number of input triggers per counter, routed to one counter
15	TCPWM_TR_ALL_CNT_NR	27	Number of input triggers routed to all counters, based on the pin package
16	TCPWM_GRP_NR	3	Number of TCPWM0 counter groups
17	TCPWM_GRP_NR0_GRP_GRP_CNT_NR	63	Number of counters per TCPWM0 Group #0
18	TCPWM_GRP_NR0_CNT_GRP_CNT_WIDTH	16	Counter width in number of bits per TCPWM0 Group #0

(table continues...)

Table 27 (continued) Miscellaneous configuration for devices

Sl. no.	Configuration	Number/ instances	Description
19	TCPWM_GRP_NR1_GRP_GRP_CNT_NR	12	Number of counters per TCPWM0 Group #1
20	TCPWM_GRP_NR1_CNT_GRP_CNT_WIDTH	16	Counter width in number of bits per TCPWM0 Group #1
21	TCPWM_GRP_NR2_GRP_GRP_CNT_NR	4	Number of counters per TCPWM0 Group #2
22	TCPWM_GRP_NR2_CNT_GRP_CNT_WIDTH	32	Counter width in number of bits per TCPWM0 Group #2
23	CANFD0_MRAM_SIZE/ CANFD1_MRAM_SIZE	24	Message RAM size in KB shared by all the channels
24	EVTGEN_COMP_STRUCT_NR	11	Number of event generator comparator structures

25 Development support

XMC5200 has a rich set of documentation, programming tools, and online resources to assist during the development process. Visit www.infineon.com to find out more.

25.1 Documentation

A suite of documentation supports XMC5200 to ensure that you can find answers to your questions quickly. This section contains a list of some of the key documents.

25.1.1 Software user guide

A step-by-step guide for using the sample driver library along with third-party IDEs such as IAR EWARM.

25.1.2 Technical reference manual

The Technical Reference Manual (TRM) contains all the technical detail needed to use the XMC5200 device, including a complete description of all registers. The TRM is available in the documentation section at www.infineon.com.

25.2 Tools

XMC5200 is supported on Infineon ModusToolbox™ software that gives user experience with either a local or GitHub-hosted set of software repos. XMC5200 is also supported on third-party development tool ecosystems such as IAR and supported by Infineon programming utilities for programming, erasing, or reading using Infineon's MiniProg4 or Segger J-link. More details are available in the documentation section at www.infineon.com.

26 Electrical specifications

26.1 Absolute maximum ratings

Using this device under conditions outside the specified minimum and maximum limits listed in Table 28 may result in permanent damage to the device. Prolonged exposure to conditions within the specified limits of Table 28, but beyond those defined for normal operation, may impact the reliability of the device.

The maximum storage temperature is 150°C, as per the JEDEC Standard JESD22-A103 for High Temperature Storage Life. When the device is operated under conditions within the specified limits of Table 28 but beyond normal operating conditions, it may not perform according to its specifications.

Power considerations

The average chip-junction temperature, T_J, in °C, may be calculated using (1):

$$T_J = T_A + (P_D \times \theta_{JA}) \quad (1)$$

Where:

T_A is the ambient temperature in °C.

θ_{JA} is the package junction-to-ambient thermal resistance, in °C/W.

P_D is the sum of P_{INT} and P_{IO} ($P_D = P_{INT} + P_{IO}$).

P_{INT} is the chip internal power. ($P_{INT} = V_{DDD} \times I_{DD} + V_{DDA} \times I_A$)

P_{IO} represents the power dissipation on input and output pins; user determined.

For most applications, $P_{IO} < P_{INT}$ and may be neglected.

On the other hand, P_{IO} may be significant if the device is configured to continuously drive external modules and/or memories.

Table 28 Absolute maximum ratings

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID10	V_{DDD_ABS}	V_{DDD} power supply voltage[26]	V_{SSD} – 0.3	–	V_{SSD} + 6.0	V	For ports 0, 1, 2, 3, 4, 5, 16, 17, 18, 19, 20, 21, 22, 23
SID10B	$V_{DDIO\_1\_ABS}$	V_{DDIO_1} power supply voltage[26]	V_{SSD} – 0.3	–	V_{SSD} + 6.0	V	$V_{DDIO\_1} \geq V_{DDD}$ For ports 6, 7, 8, 9[27]
SID10C1	$V_{DDIO\_2\_ABS}$	V_{DDIO_2} power supply voltage[26]	V_{SSD} – 0.3	–	V_{SSD} + 6.0	V	For ports 10, 11, 12, 13, 14, 15
SID11	V_{DDA_ABS}	V_{DDA} analog power supply voltage[26]	V_{SSA} – 0.3	–	V_{SSA} + 6.0	V	$V_{DDIO\_2} = V_{DDA}$
SID12	V_{REFH_ABS}	Analog reference voltage, HIGH[26]	V_{SSA} – 0.3	–	V_{SSA} + 6.0	V	$V_{REFH} \leq V_{DDA}$ + 0.3 V
SID12A	V_{REFL_ABS}	Analog reference voltage, LOW[26]	V_{SSA} – 0.3	–	V_{SSA} + 0.3	V	

(table continues...)

26 These parameters are based on the condition that $V_{SSD} = V_{SSA} = 0.0$ V.

27 The I/Os in the VDDIO_1 domain refer to the V_{DDD} domain in the 64-LQFP package.

Table 28 (continued) Absolute maximum ratings

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID15A	V_{I0_ABS0}	Input voltage[26]	V_{SSD} – 0.5	–	V_{DDD} + 0.5	V	For ports 0, 1, 2, 3, 4, 5, 16, 17, 18, 19, 20, 21, 22, 23
SID15B	V_{I1_ABS1}	Input voltage[26]	V_{SSD} – 0.5	–	V_{DDIO_1} + 0.5	V	For ports 6, 7, 8, 9[27]
SID15C	V_{I2_ABS2}	Input voltage[26]	V_{SSD} – 0.5	–	V_{DDIO_2} + 0.5	V	For ports 10, 11, 12, 13, 14, 15
SID16	V_{IA_ABS}	Analog input voltage[26]	V_{SSA} – 0.3	–	V_{DDA} + 0.3	V	
SID17A	V_{O0_ABS0}	Output voltage[26]	V_{SSD} – 0.3	–	V_{DDD} + 0.3	V	For ports 0, 1, 2, 3, 4, 5, 16, 17, 18, 19, 20, 21, 22, 23
SID17B	V_{O1_ABS1}	Output voltage[26]	V_{SSD} – 0.3	–	V_{DDIO_1} + 0.3	V	For ports 6, 7, 8, 9[27]
SID17C	V_{O2_ABS2}	Output voltage[26]	V_{SSD} – 0.3	–	V_{DDIO_2} + 0.3	V	For ports 10, 11, 12, 13, 14, 15
SID18	I_{CLAMP_ABS}	Maximum clamp current [28] [29] [30]	–5	–	5	mA	
SID18A	$I_{CLAMP\_SUPPLY\_POS\_ABS}$	Maximum positive clamp current per I/O supply pin. Limit applies to I/O supply pin closest to the B+ injected current[31]	–	–	10	mA	+B injected DC currents are not allowed for Ports 11 and 21 .
SID18B	$I_{CLAMP\_SUPPLY\_NEG\_ABS}$	Maximum negative clamp current per I/O ground pin. Limit applies to I/O supply pin closest to the B+ injected current[31]	–	–	10	mA	+B injected DC currents are not allowed for Ports 11 and 21 .
SID18C	$I_{CLAMP\_TOTAL\_POS\_ABS}$	Maximum positive clamp current per I/O supply, if not limited by the per supply pin (based on SID18A).	–	–	50	mA	

(table continues...)

26 These parameters are based on the condition that $V_{SSD} = V_{SSA} = 0.0$ V.

27 The I/Os in the VDDIO_1 domain refer to the V_{DDD} domain in the 64-LQFP package.

28 A current-limiting resistor must be provided such that the current at the I/O pin does not exceed rated values at any time, including during power transients. See Figure 11 for more information on the recommended circuit.

29 V_{DDD} and V_{DDIO} must be sufficiently loaded or protected to prevent them from being pulled out of the recommended operating range by the clamp current.

30 When the conditions of , and SID18A/B/C/D are met, $|I_{CLAMP\_ABS}|$ supersedes V_{IA_ABS} and V_{I_ABS}.

31 The definition of “closer” depends on the package. In LQFP packaging, “closest” is determined by counting pins. For example, in a 100-LQFP package, P5.2 (pin 16) is closer to the VDDD on pin 12 than on pin 24. Ports 11 and 21 should not be used for injection currents. The impact of injection currents is only defined for GPIO_STD/GPIO_ENH type I/Os.

Table 28 (continued) Absolute maximum ratings

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID18D	$I_{CLAMP\_TOTAL\_NEG\_ABS}$	Maximum negative clamp current per I/O ground, if not limited by the per supply pin (based on SID18B).	–	–	50	mA	
SID20A	I_{OL1A_ABS}	LOW-level maximum output current[32]	–	–	6	mA	For GPIO_STD, configured for drive_sel<1:0>= 0b0X
SID20B	I_{OL1B_ABS}	LOW-level maximum output current[32]	–	–	2	mA	For GPIO_STD, configured for drive_sel<1:0>= 0b10
SID20C	I_{OL1C_ABS}	LOW-level maximum output current[32]	–	–	1	mA	For GPIO_STD, configured for drive_sel<1:0>= 0b11
SID21A	I_{OL2A_ABS}	LOW-level maximum output current[32]	–	–	6	mA	For GPIO_ENH, configured for drive_sel<1:0>= 0b0X
SID21B	I_{OL2B_ABS}	LOW-level maximum output current[32]	–	–	2	mA	For GPIO_ENH, configured for drive_sel<1:0>= 0b10
SID21C	I_{OL2C_ABS}	LOW-level maximum output current[32]	–	–	1	mA	For GPIO_ENH, configured for drive_sel<1:0>= 0b11
SID26A	$\Sigma I_{OL\_ABS\_GPIO}$	LOW-level total output current[33]	–	–	50	mA	
SID27A	I_{OH1A_ABS}	HIGH-level maximum output current[32]	–	–	–5	mA	For GPIO_STD, configured for drive_sel<1:0>= 0b0X
SID27B	I_{OH1B_ABS}	HIGH-level maximum output current[32]	–	–	–2	mA	For GPIO_STD, configured for drive_sel<1:0>= 0b10

(table continues...)

32 The maximum output current is the peak current flowing through any one GPIO.
33 The total output current is the maximum current flowing through all I/Os (GPIO_STD, and GPIO_ENH).

Table 28 (continued) Absolute maximum ratings

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID27C	I_{OH1C_ABS}	HIGH-level maximum output current[32]	–	–	–1	mA	For GPIO_STD, configured for drive_sel<1:0>= 0b11
SID28A	I_{OH2A_ABS}	HIGH-level maximum output current[32]	–	–	–5	mA	For GPIO_ENH, configured for drive_sel<1:0>= 0b0X
SID28B	I_{OH2B_ABS}	HIGH-level maximum output current[32]	–	–	–2	mA	For GPIO_ENH, configured for drive_sel<1:0>= 0b10
SID28C	I_{OH2C_ABS}	HIGH-level maximum output current[32]	–	–	–1	mA	For GPIO_ENH, configured for drive_sel<1:0> = 0b11
SID33A	$\Sigma I_{OH\_ABS\_GPIO}$	HIGH-level total output current[33]	–	–	–50	mA	
SID34	P_D	Power dissipation	–	–	1000	mW	T_J should not exceed 150 °C
SID36	T_A	Ambient temperature	–40	–	125	°C	
SID37	T_{STG}	Storage temperature	–55	–	150	°C	
SID38	T_J	Operating junction temperature	–40	–	150	°C	
SID39A	V_{ESD_HBM}	Electrostatic discharge human body model	2000	–	–	V	
SID39B1	V_{ESD_CDM1}	Electrostatic discharge charged device model for corner pins	750	–	–	V	
SID39B2	V_{ESD_CDM2}	Electrostatic discharge charged device model for all other pins	500	–	–	V	
SID39C	I_{LU}	The maximum pin current the device can tolerate before triggering a latch-up	–100	–	100	mA	

32 The maximum output current is the peak current flowing through any one GPIO.
33 The total output current is the maximum current flowing through all I/Os (GPIO_STD, and GPIO_ENH).

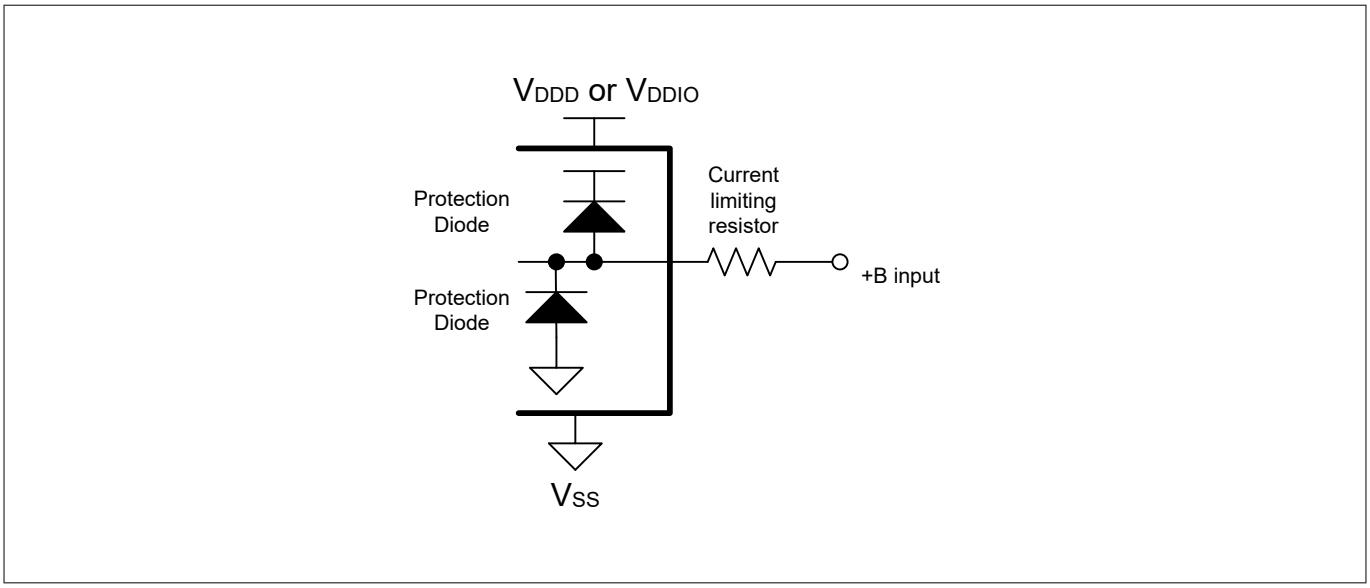


Figure 11 Example of a recommended circuit

***Note**: +B is the positive battery voltage around 45 V.*

Warning**: **Semiconductor devices can be permanently damaged if subjected to stress such as excessive voltage, current, or temperature beyond their absolute maximum ratings. Ensure that none of these ratings are exceeded under any circumstances.

26.2 Device-level specifications

Table 29 Recommended operating conditions

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
Recommended operating conditions							
SID40	V_{DDD}, V_{DDA}, V_{DDIO_1}, V_{DDIO_2}	Power supply voltage[34]	2.7[35]	–	5.5[36]	V	
SID40A	$V_{DDIO\_1\_EFP}$	Power supply voltage for eFuse programming[37]	3	–	5.5	V	
SID41	C_{S1}	Smoothing capacitor[38], [39]	3.76	–	11	µF	

34 V_{DDD}, V_{DDIO_1}, V_{DDIO_2}, and V_{DDA} do not have any sequencing limitation and can establish in any order. These supplies (except for V_{DDA} and V_{DDIO_2}) are independent in voltage level. See 12-Bit SAR ADC DC Specifications when using ADC units.

35 3.0 V ±10% is supported with a lower BOD setting option for V_{DDD} and V_{DDA}. This setting provides robust protection for internal timing but BOD reset occurs at a voltage below the specified operating conditions. A higher BOD setting option is available (consistent with down to 3.0 V) and guarantees that all operating conditions are met.

36 5.0 V ±10% is supported with a higher OVD setting option for V_{DDD} and V_{DDA}. This setting provides robust protection for internal and interface timing, but OVD reset occurs at a voltage above the specified operating conditions. A lower OVD setting option is available (consistent with up to 5.0 V) and guarantees that all operating conditions are met. Voltage overshoot to a higher OVD setting range for VDDD and VDDA is permissible, provided the duration is less than 2 hours cumulated. Note that during overshoot voltage condition electrical parameters are not guaranteed.

37 eFuse programming must be executed with the part in a “quiet” state, with minimal activity (preferably only JTAG or a single CAN channel on VDDD domain, no activity on V_{DDIO_1}).

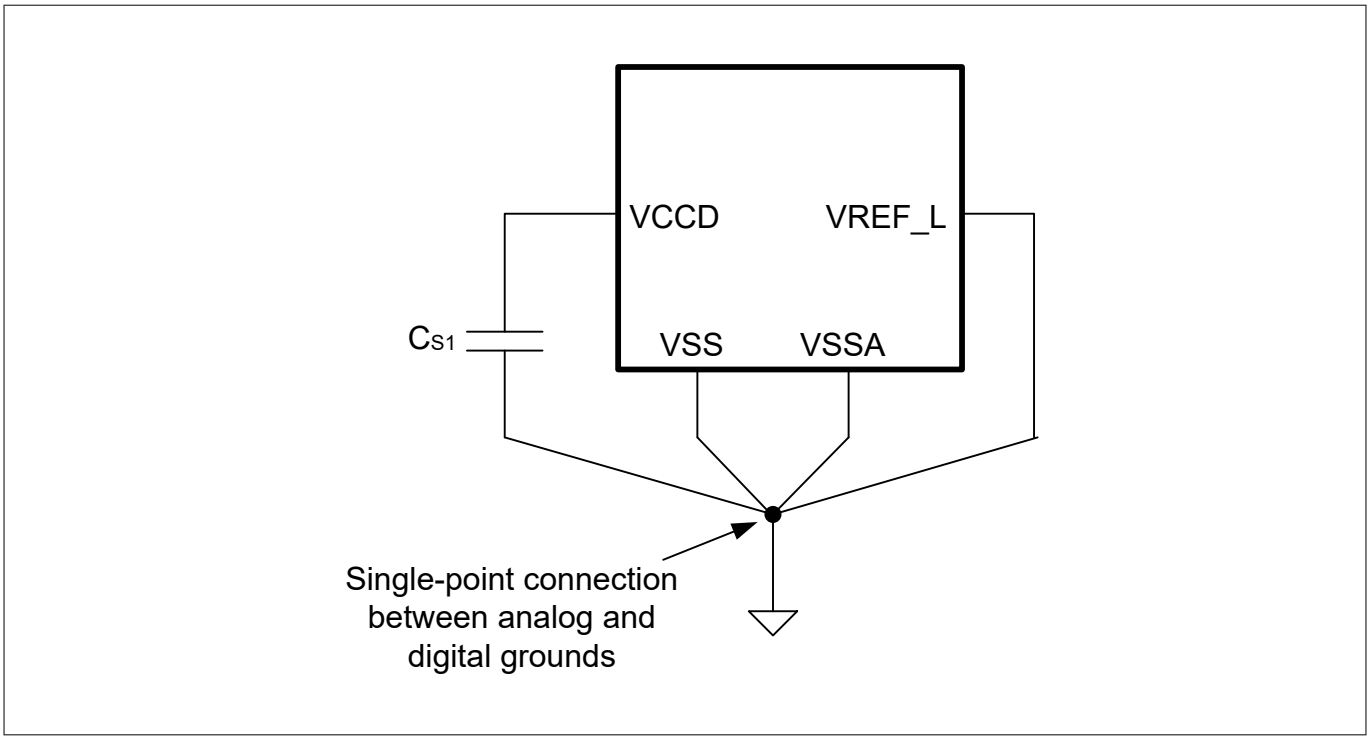


Figure 12 Smoothing capacitor

Smoothing capacitor should be placed as close as possible to the V_{CCD} pin.

38 Smoothing capacitor, C_{S1} is required per chip (not per V_{CCD} pin). The V_{CCD} pins must be connected together to ensure a low-impedance connection (see the requirement in Figure 12).

39 Capacitors used for power supply decoupling or filtering are operated under a continuous DC-bias. Many capacitors used with DC power across them provide less than their target capacitance, and their capacitance is not constant across their working voltage range. When selecting capacitors for use with this device, ensure that the selected components provide the required capacitance under the specific operating conditions of temperature and voltage used in your design. While the temperature coefficient is normally found within a parts catalog (such as, X7R, C0G, Y5V), the matching voltage coefficient may only be available on the component datasheet or direct from the manufacturer. Use of components that do not provide the required capacitance under the actual operating conditions may cause the device to operate to less than datasheet specifications.

26.3 DC specifications

Table 30 DC specifications, CPU current, and transition time specifications

All specifications are valid for –40°C ≤ T_A ≤ 125°C and for 2.7 V to 5.5 V except where noted.

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID49C1A	$I_{DD1\_CM04\_8\_1A}$	LP Active mode (CM4 and CM0+ at 8 MHz, all peripherals are disabled)	–	4	9	mA	CM0+ and CM4 clocked at 8 MHz with IMO. All peripherals are disabled. No I/O toggling. TYP: T_A = 25°C, V_{DDD} = 5.0 V, process typ (TT), CM0+ and CM4 executing Dhrystone from flash with cache enabled MAX: T_A = 25°C, V_{DDD} = 5.5 V, process worst (FF), CM0+ and CM4 executing Dhrystone from flash with cache enabled.
SID49CA	$I_{DD1\_CM04\_8A}$	LP Active mode (CM4 and CM0+ at 8 MHz, all peripherals are enabled)	–	5	51	mA	CM0+ and CM4 clocked at 8 MHz with IMO. All peripherals are enabled. No I/O toggling. M-DMA transferring data from code + work flash, P-DMA chains with maximum trigger activity. TYP: T_A = 25°C, V_{DDD} = 5.0 V, process typ (TT), CM0+ and CM4 executing Dhrystone from flash with cache enabled MAX: T_A = 125°C, V_{DDD} = 5.5 V, process worst (FF), CM0+ and CM4 executing max_power.c from flash with cache enabled.

(table continues...)

Table 30 (continued) DC specifications, CPU current, and transition time specifications

All specifications are valid for –40°C ≤ T_A ≤ 125°C and for 2.7 V to 5.5 V except where noted.

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID49E1	$I_{DD1\_F160\_1M}$	Active mode (CM4 at 160 MHz, CM0+ at 80 MHz, all peripherals are enabled)	–	39	102	mA	PLL enabled at 160 MHz with ECO reference. All peripherals are enabled. No I/O toggling. M-DMA transferring data from code + work flash, P-DMA chains with maximum trigger activity. TYP: T_A = 25°C, V_{DDD} = 5.0 V, process typ (TT), CM4 and CM0+ executing Dhrystone from flash with cache enabled. MAX: T_A = 125°C, V_{DDD} = 5.5 V, process worst (FF), CM4 and CM0+ executing max_power.c from flash with cache enabled
SID53A1	$I_{DD2\_8\_1}$	All CPUs in Sleep mode	–	3	46	mA	PLL disabled, CM4 and CM0+ are sleeping at 8 MHz with IMO. All peripherals, peripheral clocks, interrupts, CSV, DMA, FLL, ECO are disabled. No I/O toggling. Typ: T_A = 25°C, V_{DDD} = 5.0 V, process typ (TT) Max: T_A = 125°C, V_{DDD} = 5.5 V, process worst (FF)

(table continues...)

Table 30 (continued) DC specifications, CPU current, and transition time specifications

All specifications are valid for –40°C ≤ T_A ≤ 125°C and for 2.7 V to 5.5 V except where noted.

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID56A	I_{DD_CWU2}	Average current for cyclic wake-up operation. This is the average current for the specified LP Active mode and DeepSleep mode (RTC, WDT and Event generator operating).	–	46	136	µA	V_{DDD} = 5.5 V, T_A = 25°C, 64-KB SRAM retention, ILO0 operation in DeepSleep, Smart I/O operations with ILO0, CM0+, CM4: Retain Typ: process typ (TT) Max: process worst (FF) This average current is achieved under the following conditions. 1. MCU repetitively goes from DeepSleep to LP Active with a period of 32 ms. 2. One of the I/Os is toggled using Smart I/O to activate an external sensor connected to an analog input of A/D in DeepSleep 3. After 200 µs delay, the CM4 wakes up by event generator trigger to LP Active mode with IMO and A/D conversion is triggered by software. 4. Group A/D conversion is performed on 5 channels with the sampling time of 1 µs each. 5. Once the group A/D conversion is finished, and the results fit in the window of the range comparator, the I/O is toggled back by software to de-activate the sensor and the CM4 goes back to DeepSleep.
SID59A	I_{DD_DS64B}	64-KB SRAM retention, ILO0 operation in DeepSleep mode	–	35	130	µA	DeepSleep Mode (RTC, WDT, and event generator operating, all other peripherals are off except for retention registers), T_A = 25°C, CM0+, CM4: Retained Typ: V_{DDD} = 5.0 V, process typ (TT) Max: V_{DDD} = 5.5 V, process worst (FF)

(table continues...)

Table 30 (continued) DC specifications, CPU current, and transition time specifications

All specifications are valid for –40°C ≤ T_A ≤ 125°C and for 2.7 V to 5.5 V except where noted.

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID61A	I_{DD_DS64D}	64-KB SRAM retention, ILO0 operation in DeepSleep mode	–	0.9	3.5	mA	DeepSleep Mode steady state at T_A = 125°C (RTC, WDT, and event generator operating, all other peripherals are off except for retention registers),CM0+, CM4: Retained Typ: V_{DDD} = 5.0 V, process typ (TT) Max: V_{DDD} = 5.5 V, process worst (FF)
Hibernate mode							
SID62	I_{DD_HIB1}	Hibernate Mode	–	5	–	µA	ILO0/WDT operating. All other peripherals, and all CPUs are off. T_A = 25°C, V_{DDD} = 5.5 V, process typ (TT)
SID62A	I_{DD_HIB2}	Hibernate Mode	–	–	130	µA	ILO0/WDT operating. All other peripherals, and all CPUs are off. T_A = 125°C, V_{DDD} = 5.5 V, process worst (FF)
Power mode transition times							
SID65	t_{ACT_DS}	Power down time from Active to DeepSleep	–	–	2.5	µs	When the IMO is already running and all HFCLK roots are at least 8 MHz. HFCLK roots that are slower than this will require additional time to turn off.
SID63	t_{DS_ACT}	DeepSleep to Active transition time (IMO clock, SRAM execution)	–	–	10[40]	µs	When using the 8-MHz IMO. Measured from wakeup interrupt during DeepSleep until wakeup.
SID63C	t_{DS_ACT}	DeepSleep to Active transition time (IMO clock, flash execution)	–	–	20[40]	µs	When using the 8-MHz IMO. Measured from wakeup interrupt during DeepSleep until flash execution.
SID63A	$t_{DS\_ACT\_FLL}$	DeepSleep to Active transition time (FLL clock, SRAM execution)	–	–	15[40]	µs	When using the FLL to generate 96 MHz from the 8-MHz IMO. Measured from wakeup interrupt during DeepSleep until the FLL locks.

(table continues...)

40 At cold temperature -5°C to - 40°C, the DeepSleep to Active transition time can be higher than the maximum time indicated by as much as 20 µs.

Table 30 (continued) DC specifications, CPU current, and transition time specifications

All specifications are valid for –40°C ≤ T_A ≤ 125°C and for 2.7 V to 5.5 V except where noted.

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID63D	$t_{DS\_ACT\_FLL1}$	DeepSleep to Active transition time (FLL clock, flash execution)	–	–	21.5[40]	µs	When using the FLL to generate 96 MHz from the 8-MHz IMO. Measured from wakeup interrupt during DeepSleep until flash execution.
SID63B	$t_{DS\_ACT\_PLL}$	DeepSleep to Active transition time (PLL clock, SRAM or flash execution)	–	–	60[40]	µs	When using the PLL to generate 96 MHz from the 8-MHz IMO. Measured from wakeup interrupt during DeepSleep until the PLL locks.
SID68	t_{HVR_ACT}	Release time from HV reset (POR, BOD, OVD, OCD, WDT, Hibernate wakeup, or XRES_L) release until CM0+ begins executing ROM boot	–	–	265	µs	Without boot runtime. Guaranteed by design
SID68A	t_{LVR_ACT}	Release time from LV reset (Fault, Internal system reset, MCWDT, or CSV) during Active/ Sleep until CM0+ begins executing ROM boot	–	–	10	µs	Without boot runtime. Guaranteed by design
SID68B	t_{LVR_DS}	Release time from LV reset (Fault, or MCWDT) during DeepSleep until CM0+ begins executing ROM boot	–	–	15	µs	Without boot runtime. Guaranteed by design
SID80A	t_{RB_N}	ROM boot startup time or wakeup time from hibernate in NORMAL protection state	–	–	1800	µs	Guaranteed by Design, CM0+ clocked at 100 MHz (Flash boot version 3.1.0.556 and later)
SID80B	t_{RB_S}	ROM boot startup time or wakeup time from hibernate in SECURE protection state	–	–	2740	µs	Guaranteed by Design, CM0+ clocked at 100 MHz (Flash boot version 3.1.0.556 and later)

(table continues...)

[40] At cold temperature -5°C to - 40°C, the DeepSleep to Active transition time can be higher than the maximum time indicated by as much as 20 µs.

Table 30 (continued) DC specifications, CPU current, and transition time specifications

All specifications are valid for –40°C ≤ T_A ≤ 125°C and for 2.7 V to 5.5 V except where noted.

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID81A	t_{FB}	Flash boot startup time or wakeup time from hibernate in NORMAL/SECURE protection state	–	–	80	µs	Guaranteed by Design, TOC2_FLAGS = 0x2CF, CM0+ clocked at 100 MHz, Listen window = 0 ms (Flash boot version 3.1.0.556 and later)
SID81B	t_{FB_A}	Flash boot with app authentication time in NORMAL/SECURE protection state	–	–	5000	µs	Guaranteed by Design, TOC2_FLAGS = 0x24F, CM0+ clocked at 100 MHz, Listen window = 0 ms, Public key exponent e = 0x010001, APP size is 64 KB with the last 256 bytes being a digital signature in RSASSA-PKCS1-v1.5. Valid for RSA-2048. (Flash boot version 3.1.0.556 and later)
SID80A_2	$t_{RB\_N\_2}$	ROM boot startup time or wakeup time from hibernate in NORMAL protection state	–	–	2930	µs	Guaranteed by Design, CM0+ clocked at 50 MHz, (Flash boot version 3.1.0.556 and later)
SID80B_2	$t_{RB\_S\_2}$	ROM boot startup time or wakeup time from hibernate in SECURE protection state	–	–	4680	µs	Guaranteed by Design, CM0+ clocked at 50 MHz, (Flash boot version 3.1.0.556 and later)
SID81A_2	t_{FB_2}	Flash boot startup time or wakeup time from hibernate in NORMAL/SECURE protection state	–	–	200	µs	Guaranteed by Design, TOC2_FLAGS = 0x2CF, CM0+ clocked at 50 MHz, Listen window = 0 ms (Flash boot version 3.1.0.556 and later)

(table continues...)

Table 30 (continued) DC specifications, CPU current, and transition time specifications

All specifications are valid for –40°C ≤ T_A ≤ 125°C and for 2.7 V to 5.5 V except where noted.

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID81B_2	$t_{FB\_A\_2}$	Flash boot with app authentication time in NORMAL/SECURE protection state	–	–	10000	µs	Guaranteed by Design, TOC2_FLAGS = 0x24F, CM0+ clocked at 50 MHz, Listen window = 0 ms, Public key exponent e = 0x010001, APP size is 64 KB with the last 256 bytes being a digital signature in RSASSA-PKCS1-v1.5. Valid for RSA-2048. (Flash boot version 3.1.0.556 and later)
Regulator specifications							
SID600	V_{CCD}	Core supply voltage	1.05	1.1	1.15	V	
SID601	I_{DD_ACT}	Regulator operating current in Active/Sleep mode	–	80	150	µA	Guaranteed by design
SID602	I_{DD_DPSLP}	Regulator operating current in DeepSleep mode	–	1.5	20	µA	Guaranteed by design
SID604	I_{OUT}	Available regulator output current for operation	–	–	150	mA	Without triggering OVD
SID603	I_{RUSH}	In-rush current	–	–	375	mA	Average V_{DDD} current until C_{s1} (connected to V_{CCD} pin) is charged after Active regulator is turned on

26.4 Reset specifications

All specifications are valid for –40°C ≤ T_A ≤ 125°C and for 2.7 V to 5.5 V except where noted.

Table 31 XRES_L reset

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
XRES_L DC specifications							
SID73	I_{DD_XRES}	I_{DD} when XRES_L asserted	–	–	0.9	mA	TA = 125°C, V_{DDD} = 5.5 V, process worst (FF)
SID74	V_{IH}	Input voltage HIGH threshold	0.7 × V_{DDD}	–	–	V	CMOS input

(table continues...)

Table 31 (continued) XRES_L reset

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID75	V_{IL}	Input voltage LOW threshold	–	–	0.3 × V_{DDD}	V	CMOS input
SID76	R_{PULLUP}	Pull-up resistor	7	–	20	kΩ	
SID77	C_{IN}	Input capacitance	–	–	5	pF	
SID78	$V_{HYSXRES}$	Input voltage hysteresis	0.05 × V_{DDD}	–	–	V	
XRES_L AC specifications							
SID70	t_{XRES_ACT}	XRES_L release to Active transition time	–	–	265	µs	Without boot runtime. Guaranteed by design
SID71	t_{XRES_PW}	XRES_L pulse width	5	–	–	µs	
SID72	t_{XRES_FT}	Pulse suppression width	100	–	–	ns	

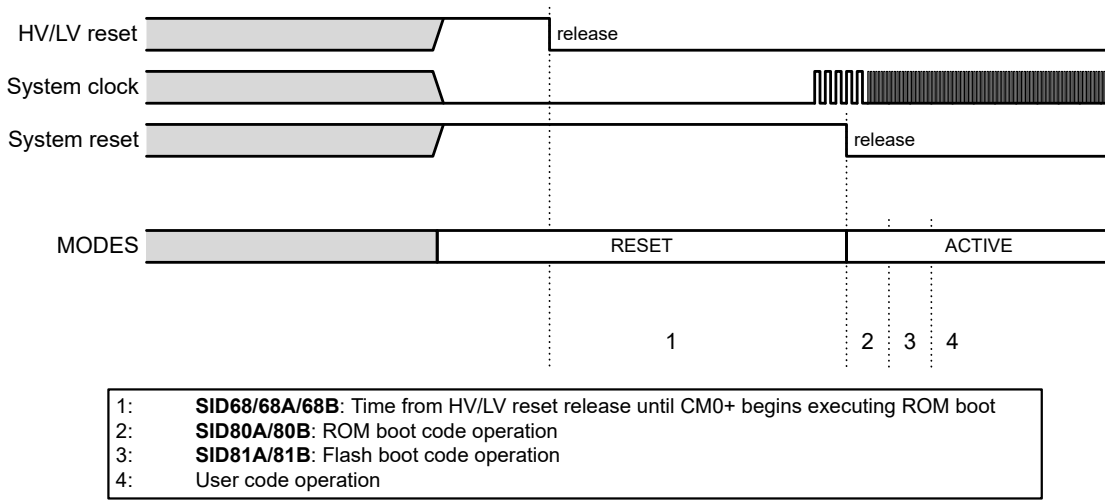


Figure 13 Reset sequence

26.5 I/O

All specifications are valid for –40°C ≤ T_A ≤ 125°C and for 2.7 V to 5.5 V except where noted.

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
GPIO_STD specifications for ports P1 through P23							
SID650	$V_{OL1\_GPIO\_STD}$	Output voltage LOW level	–	–	0.6	V	I_{OL} = 6 mA drive_sel<1:0> = 0b0X, 4.5 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} ≤ 5.5 V

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID650C	$V_{OL1C\_GPIO\_STD}$	Output voltage LOW level	–	–	0.4	V	I_{OL} = 5 mA drive_sel<1:0> = 0b0X, 4.5 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} ≤ 5.5 V
SID651	$V_{OL2\_GPIO\_STD}$	Output voltage LOW level	–	–	0.4	V	I_{OL} = 2 mA drive_sel<1:0> = 0b0X, 2.7 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} < 4.5 V
SID652	$V_{OL3\_GPIO\_STD}$	Output voltage LOW level	–	–	0.4	V	I_{OL} = 1 mA drive_sel<1:0> = 0b10, 2.7 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} < 4.5 V
SID652C	$V_{OL3C\_GPIO\_STD}$	Output voltage LOW level	–	–	0.4	V	I_{OL} = 2 mA drive_sel<1:0> = 0b10, 4.5 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} ≤ 5.5 V
SID653	$V_{OL4\_GPIO\_STD}$	Output voltage LOW level	–	–	0.4	V	I_{OL} = 0.5 mA drive_sel<1:0> = 0b11, 2.7 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} < 4.5 V
SID653C	$V_{OL4C\_GPIO\_STD}$	Output voltage LOW level	–	–	0.4	V	I_{OL} = 1 mA drive_sel<1:0> = 0b11, 4.5 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} ≤ 5.5 V
SID654	$V_{OH1\_GPIO\_STD}$	Output voltage HIGH level	(V_{DDD} or V_{DDIO_1} or V_{DDIO_2}) – 0.5	–	–	V	I_{OH} = –2 mA drive_sel<1:0> = 0b0X, 2.7 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} < 4.5 V
SID655	$V_{OH2\_GPIO\_STD}$	Output voltage HIGH level	(V_{DDD} or V_{DDIO_1} or V_{DDIO_2}) – 0.5	–	–	V	I_{OH} = –5 mA drive_sel<1:0> = 0b0X, 4.5 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} ≤ 5.5 V
SID656	$V_{OH3\_GPIO\_STD}$	Output voltage HIGH level	(V_{DDD} or V_{DDIO_1} or V_{DDIO_2}) – 0.5	–	–	V	I_{OH} = –1 mA drive_sel<1:0> = 0b10, 2.7 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} < 4.5 V
SID656C	$V_{OH3C\_GPIO\_STD}$	Output voltage HIGH level	(V_{DDD} or V_{DDIO_1} or V_{DDIO_2}) – 0.5	–	–	V	I_{OH} = –2 mA drive_sel<1:0> = 0b10, 4.5 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} ≤ 5.5 V

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID657	$V_{OH4\_GPIO\_STD}$	Output voltage HIGH level	(V_{DDD} or V_{DDIO_1} or V_{DDIO_2}) – 0.5	–	–	V	I_{OH} = –0.5 mA drive_sel<1:0> = 0b11, 2.7 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} < 4.5 V
SID657C	$V_{OH4C\_GPIO\_STD}$	Output voltage HIGH level	(V_{DDD} or V_{DDIO_1} or V_{DDIO_2}) – 0.5	–	–	V	I_{OH} = –1 mA drive_sel<1:0> = 0b11, 4.5 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} ≤ 5.5 V
SID658	$R_{PD\_GPIO\_STD}$	Pull-down resistance	25	50	100	kΩ	
SID659	$R_{PU\_GPIO\_STD}$	Pull-up resistance	25	50	100	kΩ	
SID660	$V_{IH\_CMOS\_GPIO\_STD}$	Input voltage HIGH threshold in CMOS mode	0.7 × (V_{DDD} or V_{DDIO_1} or V_{DDIO_2})	–	–	V	
SID661	$V_{IH\_TTL\_GPIO\_STD}$	Input voltage HIGH threshold in TTL mode	2.0	–	–	V	
SID662	$V_{IH\_AUTO\_GPIO\_STD}$	Input voltage HIGH threshold in AUTO mode	0.8 × (V_{DDD} or V_{DDIO_1} or V_{DDIO_2})	–	–	V	
SID663	$V_{IL\_CMOS\_GPIO\_STD}$	Input voltage LOW threshold in CMOS mode	–	–	0.3 × (V_{DDD} or V_{DDIO_1} or V_{DDIO_2})	V	
SID664	$V_{IL\_TTL\_GPIO\_STD}$	Input voltage LOW threshold in TTL mode	–	–	0.8	V	
SID665	$V_{IL\_AUTO\_GPIO\_STD}$	Input voltage LOW threshold in AUTO mode	–	–	0.5 × (V_{DDD} or V_{DDIO_1} or V_{DDIO_2})	V	
SID666	$V_{HYST\_CMOS\_GPIO\_STD}$	Hysteresis in CMOS mode	0.05 × (V_{DDD} or V_{DDIO_1} or V_{DDIO_2})	–	–	V	
SID668	$V_{HYST\_AUTO\_GPIO\_STD}$	Hysteresis in AUTO mode	0.05 × (V_{DDD} or V_{DDIO_1} or V_{DDIO_2})	–	–	V	
SID669	$C_{in\_GPIO\_STD}$	Input pin capacitance	–	–	5	pF	For 10 MHz and 100 MHz

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID670	$I_{IL\_GPIO\_STD}$	Input leakage current	–250	0.02	250	nA	For GPIO_STD except P21.0, P21.1, P21.2, P21.3, P23.3, P23.4. $V_{DDIO\_1} = V_{DDIO\_2} = V_{DDD} = V_{DDA} = 5.5$ V, $V_{SSD} < V_I < V_{DDD}, V_{DDIO\_1}, V_{DDIO\_2}$ –40°C ≤ TA ≤ 125°C TYP: $T_A = 25$°C, $V_{DDIO\_1} = V_{DDIO\_2} = V_{DDD} = V_{DDA} = 5.0$ V
SID670C	$I_{IL\_GPIO\_STD\_B}$	Input leakage current	–700	0.02	700	nA	Only for P21.0, P21.1, P21.2, P21.3, P23.3, P23.4. $V_{DDIO\_1} = V_{DDIO\_2} = V_{DDD} = V_{DDA} = 5.5$ V, $V_{SSD} < V_I < V_{DDD}, V_{DDIO\_1}, V_{DDIO\_2}$ –40°C ≤ TA ≤ 125°C TYP: $T_A = 25$°C, $V_{DDIO\_1} = V_{DDIO\_2} = V_{DDD} = V_{DDA} = 5.0$ V
SID671	t_R or t_F (fast)$_{\_20\_0\_GPIO\_STD}$	Rise time or fall time (10% to 90% of V_{DDIO})	1	–	10	ns	20-pF load, drive_sel<1:0> = 0b00
SID672	t_R or t_F (fast)$_{\_50\_0\_GPIO\_STD}$	Rise time or fall time (10% to 90% of V_{DDIO})	1	–	20	ns	50-pF load, drive_sel<1:0> = 0b00
SID673	t_R or t_F (fast)$_{\_20\_1\_GPIO\_STD}$	Rise time or fall time (10% to 90% of V_{DDIO})	1	–	20	ns	20-pF load, drive_sel<1:0> = 0b01, guaranteed by design
SID674	t_R or t_F (fast)$_{\_10\_2\_GPIO\_STD}$	Rise time or fall time (10% to 90% of V_{DDIO})	1	–	20	ns	10-pF load, drive_sel<1:0> = 0b10, guaranteed by design
SID675	t_R or t_F (fast)$_{\_6\_3\_GPIO\_STD}$	Rise time or fall time (10% to 90% of V_{DDIO})	1	–	20	ns	6-pF load, drive_sel<1:0> = 0b11, guaranteed by design
SID676	t_F (fast)$_{\_100\_GPIO\_STD}$	Fall time (30% to 70% of V_{DDIO})	0.35	–	250	ns	10-pF to 400-pF load, RPU = 767 Ω, drive_sel<1:0>= 0b00, Freq = 100 kHz
SID677	t_F (fast)$_{\_400\_GPIO\_STD}$	Fall time (30% to 70% of V_{DDIO})	0.35	–	250	ns	10-pF to 400-pF load, RPU = 350 Ω, drive_sel<1:0>= 0b00, Freq = 400 kHz
SID678	$f_{IN\_GPIO\_STD}$	Input frequency	–	–	100	MHz	

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID679	$f_{OUT\_GPIO\_STD0H}$	Output frequency	–	–	50	MHz	20-pF load, drive_sel<1:0>= 00, 4.5 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} ≤ 5.5 V
SID680	$f_{OUT\_GPIO\_STD0L}$	Output frequency	–	–	32	MHz	20-pF load, drive_sel<1:0>= 00, 2.7 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} < 4.5 V
SID681	$f_{OUT\_GPIO\_STD1H}$	Output frequency	–	–	25	MHz	20-pF load, drive_sel<1:0>= 01, 4.5 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} ≤ 5.5 V
SID682	$f_{OUT\_GPIO\_STD1L}$	Output frequency	–	–	15	MHz	20-pF load, drive_sel<1:0>= 01, 2.7 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} < 4.5 V
SID683	$f_{OUT\_GPIO\_STD2H}$	Output frequency	–	–	25	MHz	10-pF load, drive_sel<1:0>= 10, 4.5 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} ≤ 5.5 V
SID684	$f_{OUT\_GPIO\_STD2L}$	Output frequency	–	–	15	MHz	10-pF load, drive_sel<1:0>= 10, 2.7 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} < 4.5 V
SID685	$f_{OUT\_GPIO\_STD3H}$	Output frequency	–	–	15	MHz	6-pF load, drive_sel<1:0>= 11, 4.5 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} ≤ 5.5 V
SID686	$f_{OUT\_GPIO\_STD3L}$	Output frequency	–	–	10	MHz	6-pF load, drive_sel<1:0>= 11, 2.7 V ≤ V_{DDD} or V_{DDIO_1} or V_{DDIO_2} < 4.5 V
GPIO_ENH specifications only for P0							
SID650A	$V_{OL1\_GPIO\_ENH}$	Output voltage LOW level	–	–	0.6	V	I_{OL} = 6 mA drive_sel<1:0> = 0b00/01, 2.7 V ≤ V_{DDD} ≤ 5.5 V
SID650D	$V_{OL1D\_GPIO\_ENH}$	Output voltage LOW level	–	–	0.4	V	I_{OL} = 5 mA drive_sel<1:0> = 0b00/01, 4.5 V ≤ V_{DDD} ≤ 5.5 V

26 Electrical specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID651A	$V_{OL2\_GPIO\_ENH}$	Output voltage LOW level	–	–	0.4	V	I_{OL} = 2 mA drive_sel<1:0> = 0b00/01, 2.7 V ≤ V_{DDD} < 4.5 V
SID652A	$V_{OL3\_GPIO\_ENH}$	Output voltage LOW level	–	–	0.4	V	I_{OL} = 1 mA drive_sel<1:0> = 0b10, 2.7 V ≤ V_{DDD} < 4.5 V
SID652D	$V_{OL3D\_GPIO\_ENH}$	Output voltage LOW level	–	–	0.4	V	I_{OL} = 2 mA drive_sel<1:0> = 0b10, 4.5 V ≤ V_{DDD} ≤ 5.5 V
SID653A	$V_{OL4\_GPIO\_ENH}$	Output voltage LOW level	–	–	0.4	V	I_{OL} = 0.5 mA drive_sel<1:0> = 0b11, 2.7 V ≤ V_{DDD} < 4.5 V
SID653D	$V_{OL4D\_GPIO\_ENH}$	Output voltage LOW level	–	–	0.4	V	I_{OL} = 1 mA drive_sel<1:0> = 0b11, 4.5 V ≤ V_{DDD} ≤ 5.5 V
SID654A	$V_{OH1\_GPIO\_ENH}$	Output voltage HIGH level	V_{DDD} – 0.5	–	–	V	I_{OL} = –2 mA drive_sel<1:0> = 0b00/01, 2.7 V ≤ V_{DDD} < 4.5 V
SID655A	$V_{OH2\_GPIO\_ENH}$	Output voltage HIGH level	V_{DDD} – 0.5	–	–	V	I_{OL} = –5 mA drive_sel<1:0> = 0b00/01, 4.5 V ≤ V_{DDD} ≤ 5.5 V
SID656A	$V_{OH3\_GPIO\_ENH}$	Output voltage HIGH level	V_{DDD} – 0.5	–	–	V	I_{OL} = –1 mA drive_sel<1:0> = 0b10, 2.7 V ≤ V_{DDD} < 4.5 V
SID656D	$V_{OH3D\_GPIO\_ENH}$	Output voltage HIGH level	V_{DDD} – 0.5	–	–	V	I_{OL} = –2 mA drive_sel<1:0> = 0b10, 4.5 V ≤ V_{DDD} ≤ 5.5 V
SID657A	$V_{OH4\_GPIO\_ENH}$	Output voltage HIGH level	V_{DDD} – 0.5	–	–	V	I_{OL} = –0.5 mA drive_sel<1:0> = 0b11, 2.7 V ≤ V_{DDD} < 4.5 V
SID657D	$V_{OH4D\_GPIO\_ENH}$	Output voltage HIGH level	V_{DDD} – 0.5	–	–	V	I_{OL} = –1 mA drive_sel<1:0> = 0b11, 4.5 V ≤ V_{DDD} ≤ 5.5 V
SID658A	$R_{PD\_GPIO\_ENH}$	Pull-down resistance	25	50	100	kΩ	
SID659A	$R_{PU\_GPIO\_ENH}$	Pull-up resistance	25	50	100	kΩ	

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID660A	$V_{IH\_CMOS\_GPIO\_ENH}$	Input voltage HIGH threshold in CMOS mode	0.70 × V_{DDD}	–	–	V	
SID661A	$V_{IH\_TTL\_GPIO\_ENH}$	Input voltage HIGH threshold in TTL mode	2	–	–	V	
SID662A	$V_{IH\_AUTO\_GPIO\_ENH}$	Input voltage HIGH threshold in AUTO mode	0.8 × V_{DDD}	–	–	V	
SID663A	$V_{IL\_CMOS\_GPIO\_ENH}$	Input voltage LOW threshold in CMOS mode	–	–	0.3 × V_{DDD}	V	
SID664A	$V_{IL\_TTL\_GPIO\_ENH}$	Input voltage LOW threshold in TTL mode	–	–	0.8	V	
SID665A	$V_{IL\_AUTO\_GPIO\_ENH}$	Input voltage LOW threshold in AUTO mode	–	–	0.5 × V_{DDD}	V	
SID666A	$V_{HYST\_CMOS\_GPIO\_ENH}$	Hysteresis in CMOS mode	0.05 × V_{DDD}	–	–	V	
SID668A	$V_{HYST\_AUTO\_GPIO\_ENH}$	Hysteresis in AUTO mode	0.05 × V_{DDD}	–	–	V	
SID669A	$C_{in\_GPIO\_ENH}$	Input pin capacitance	–	–	5	pF	For 10 MHz and 100 MHz
SID670A	$I_{IL\_GPIO\_ENH}$	Input leakage current	–350	0.055	350	nA	$V_{DDD} = V_{DDA} = 5.5$ V, $V_{SSD} < V_I < V_{DDD}$, $-40°C \le T_A \le 125°C$ TYP: $T_A = 25°C$, $V_{DDD} = V_{DDA} = 5.0$ V
SID671A	t_R or t_F $(fast)_{\_20\_0\_GPIO\_ENH}$	Rise time or fall time (10% to 90% of V_{DDIO})	1	–	10	ns	20-pF load, drive_sel<1:0> = 0b00, slow = 0
SID672A	t_R or t_F $(fast)_{\_50\_0\_GPIO\_ENH}$	Rise time or fall time (10% to 90% of V_{DDIO})	1	–	20	ns	50-pF load, drive_sel<1:0> = 0b00, slow = 0
SID673A	t_R or t_F $(fast)_{\_20\_1\_GPIO\_ENH}$	Rise time or fall time (10% to 90% of V_{DDIO})	1	–	20	ns	20-pF load, drive_sel<1:0> = 0b01, slow = 0, guaranteed by design
SID674A	t_R or t_F $(fast)_{\_10\_2\_GPIO\_ENH}$	Rise time or fall time (10% to 90% of V_{DDIO})	1	–	20	ns	10-pF load, drive_sel<1:0> = 0b10, slow = 0, guaranteed by design

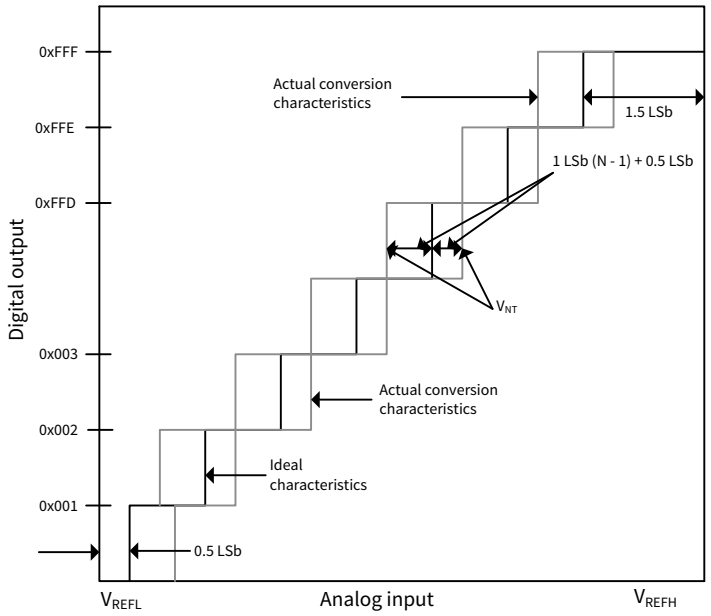
Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID675A	t_R or t_F (fast)$_{\_6\_3\_GPIO\_ENH}$	Rise time or fall time (10% to 90% of V_{DDIO})	1	–	20	ns	6-pF load, drive_sel<1:0> = 0b11, slow = 0, guaranteed by design
SID676A	t_{F_I2C} (slow)$_{\_GPIO\_ENH}$	Fall time (30% to 70% of V_{DDIO})	20 × (V_{DDD} / 5.5)	–	250	ns	10-pF to 400-pF load, drive_sel<1:0> = 0b00, slow = 1, minimum R_{PU} = 400 Ω
SID677A	t_R or t_F (slow)$_{\_20\_GPIO\_ENH}$	Rise time or fall time (10% to 90% of V_{DDIO})	20 × (V_{DDD} / 5.5)	–	160	ns	20-pF load, drive_sel<1:0> = 0b00, slow = 1, output frequency = 1 MHz
SID678A	t_R or t_F (slow)$_{\_400\_GPIO\_ENH}$	Rise time or fall time (10% to 90% of V_{DDIO})	20 × (V_{DDD} / 5.5)	–	250	ns	400-pF load, drive_sel<1:0> = 0b00, slow = 1, output frequency = 400 kHz
SID679A	$f_{IN\_GPIO\_ENH}$	Input frequency	–	–	100	MHz	
SID680A	$f_{OUT\_GPIO\_ENH0H}$	Output frequency	–	–	50	MHz	20-pF load, drive_sel<1:0>= 0b00, 4.5 V ≤ V_{DDD} ≤ 5.5 V
SID681A	$f_{OUT\_GPIO\_ENH0L}$	Output frequency	–	–	32	MHz	20-pF load, drive_sel<1:0>= 0b00, 2.7 V ≤ V_{DDD} < 4.5 V
SID682A	$f_{OUT\_GPIO\_ENH1H}$	Output frequency	–	–	25	MHz	20-pF load, drive_sel<1:0>= 0b01, 4.5 V ≤ V_{DDD} ≤ 5.5 V
SID683A	$f_{OUT\_GPIO\_ENH1L}$	Output frequency	–	–	15	MHz	20-pF load, drive_sel<1:0>= 0b01, 2.7 V ≤ V_{DDD} < 4.5 V
SID684A	$f_{OUT\_GPIO\_ENH2H}$	Output frequency	–	–	25	MHz	10-pF load, drive_sel<1:0>= 0b10, 4.5 V ≤ V_{DDD} ≤ 5.5 V
SID685A	$f_{OUT\_GPIO\_ENH2L}$	Output frequency	–	–	15	MHz	10-pF load, drive_sel<1:0>= 0b10, 2.7 V ≤ V_{DDD} < 4.5 V
SID686A	$f_{OUT\_GPIO\_ENH3H}$	Output frequency	–	–	15	MHz	6-pF load, drive_sel<1:0>= 0b11, 4.5 V ≤ V_{DDD} ≤ 5.5 V

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID687A	$f_{OUT\_GPIO\_ENH3L}$	Output frequency	–	–	10	MHz	6-pF load, drive_sel<1:0>= 0b11, 2.7 V ≤ V_{DDD} < 4.5 V
GPIO input specifications							
SID98	t_{FT}	Analog glitch filter (pulse suppression width)	–	–	50[41]	ns	One filter per port group
SID99	t_{INT}	Minimum pulse width for GPIO interrupt	160	–	–	ns	

26.6 Analog peripherals

All specifications are valid for –40°C ≤ T_A ≤ 125°C and for 2.7 V to 5.5 V except where noted.

26.6.1 SAR ADC



Total error of digital output N = (V_{NT} {1 LSb × (N - 1) + 0.5 LSb}) / 1 LSb [LSb]

1 LSb (Ideal value) = (V_{REFH} – V_{REFL}) / 4096 [V]

N: A/D converter digital output value
V_{ZT} (Ideal value): V_{REFL} + 0.5 LSb [V]
V_{FST} (Ideal value): V_{REFH} - 1.5 LSb [V]
V_{NT}: Voltage at which the digital output changes from N–1 to N

Figure 14 ADC characteristics and error definitions

41 If longer pulse suppression width is required, use Smart I/O.

Table 32 12-Bit SAR ADC DC specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID100	A_RES	SAR ADC resolution	–	–	12	bits	
SID101	A_V_{INS}	Input voltage range	V_{REFL}	–	V_{REFH}	V	
SID102	A_V_{REFH}	V_{REFH} voltage range	2.7	–	V_{DDA}	V	ADC performance degrades when high reference is higher than supply
SID102A	A_V_{DDA}[42]	V_{DDA} voltage range	2.7	–	5.5	V	
SID103	A_V_{REFL}	V_{REFL} voltage range	V_{SSA}	–	V_{SSA}	V	ADC performance degrades when low reference is lower than ground
SID103A	V_{band_gap}	Internal band gap reference voltage	0.882	0.9	0.918	V	
SID19A	CLAMP_COUPLING_RATIO_POS	Ratio of current collected on a pin to the positive current injected into a neighboring pin	–	–	0.25	%	
SID19B	CLAMP_COUPLING_RATIO_NEG	Ratio of current collected on a pin to the negative current injected into a neighboring pin	–	–	1.2	%	
SID19C	$R_{CLAMP\_INTERNAL}$	Internal pin resistance to current collection point	–	–	50	Ω	

26.6.2 Calculating the impact of neighboring pins

The three ADC specifications based on SID19A, SID19B, and SID19C, can be used to calculate the pin leakage and resulting ADC offset caused by injection current using the below formula:

$I_{LEAK} = I_{INJECTED} \times CLAMP\_COUPLING\_RATIO$

$V_{ERROR} = I_{LEAK} \times (R_{CLAMP\_INTERNAL} + R_{SOURCE})$

Code Error = $V_{ERROR} \times 2^{12} / V_{REF}$

Where:

$I_{INJECTED}$ is the injected current in mA.

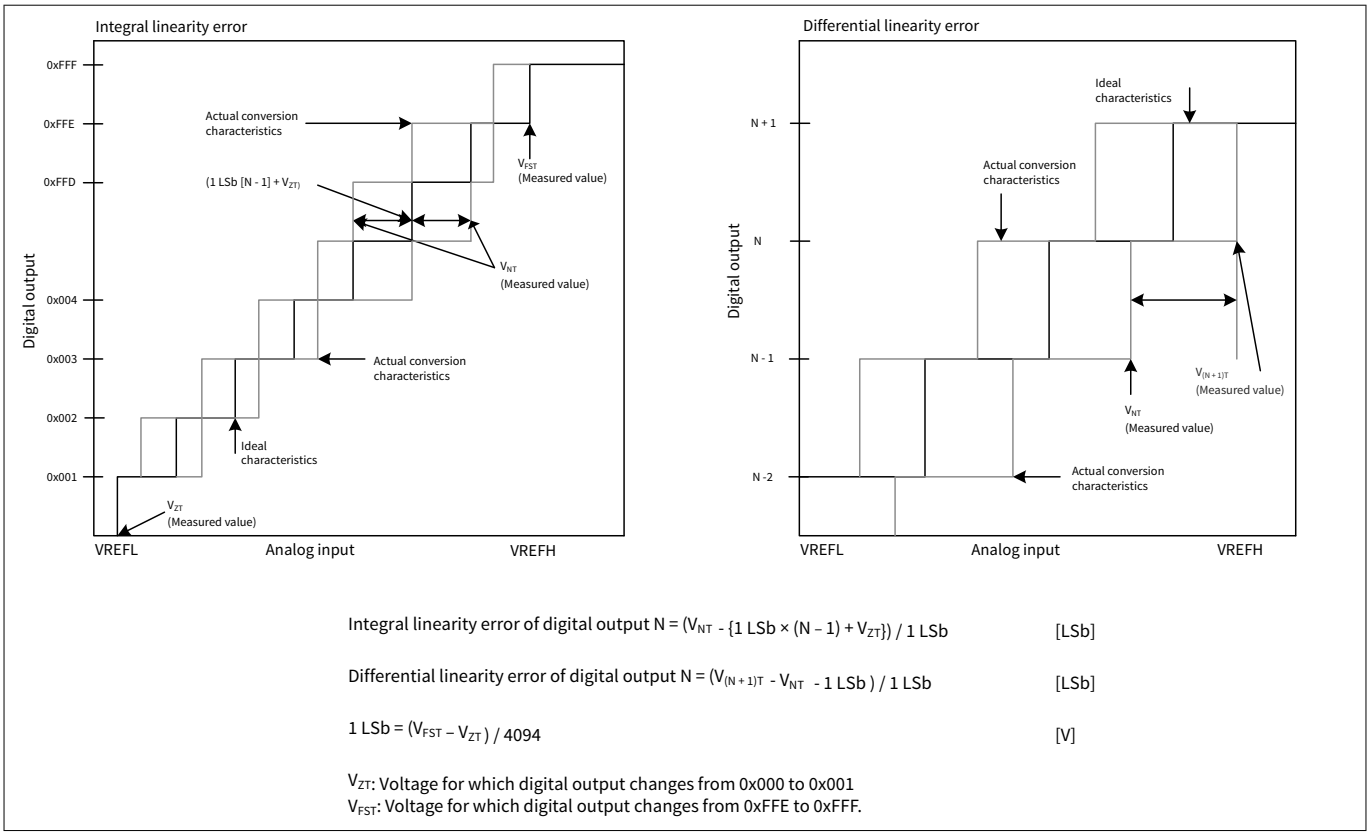
I_{LEAK} is the calculated leakage current in mA.

V_{ERROR} is the voltage error calculated due to leakage currents in V.

V_{REF} is the ADC reference voltage in V.

[42] VDDD must be greater than 0.8 × V_{DDA} when ADC[2] is enabled. VDDIO_1 must be greater than 0.8 × V_{DDA} when ADC[0] is enabled.

26 Electrical specifications



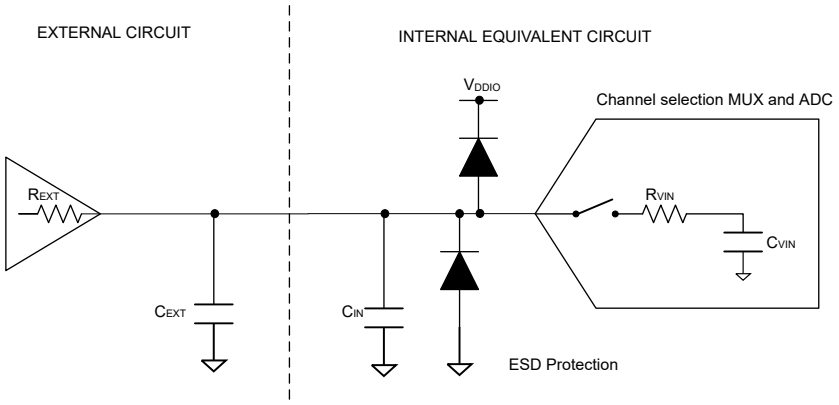
Integral linearity error of digital output N = (V_{NT} - {1 LSb × (N – 1) + V_{ZT}}) / 1 LSb [LSb]

Differential linearity error of digital output N = ($V_{(N+1)T}$ - V_{NT} - 1 LSb) / 1 LSb [LSb]

1 LSb = (V_{FST} – V_{ZT}) / 4094 [V]

V_{ZT}: Voltage for which digital output changes from 0x000 to 0x001
V_{FST}: Voltage for which digital output changes from 0xFFE to 0xFFF.

Figure 15 Integral and differential linearity errors



R_{EXT}: Source impedance
C_{EXT}: On-PCB capacitance
C_{IN}: I/O pad or Input capacitance
R_{VIN}: ADC equivalent input resistance
C_{VIN}: ADC equivalent input capacitance
K: Constant for sampling accuracy, K = ln(abs(4096/LSb_{SAMPLE}))

Sampling Time (t_{SAMPLE}) requirement is shown in the following equation
t_{SAMPLE} > K x { C_{VIN} x (R_{VIN} + R_{EXT}) + (C_{IN} + C_{EXT}) x (R_{EXT}) } [seconds]

K = value of 9.0 is recommended to get ±0.5 LSb sampling accuracy at 12-bit (LSb_{SAMPLE} = ±0.5)

Figure 16 ADC equivalent circuit for analog input

Table 33 SAR ADC AC specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID104	V_{ZT}	Zero transition voltage	–20	–	20	mV	V_{DDA} = 2.7 V to 5.5 V, –40°C ≤ T_A ≤ 125°C before offset adjustment
SID105	V_{FST}	Full-scale transition voltage	–20	–	20	mV	V_{DDA} = 2.7 V to 5.5 V, –40°C ≤ T_A ≤ 125°C before offset adjustment
SID114	f_{ADC_4P5}	ADC operating frequency	2	–	26.67	MHz	4.5 V ≤ V_{DDA} ≤ 5.5 V
SID114A	f_{ADC_2P7}	ADC operating frequency	2	–	13.34	MHz	2.7 V ≤ V_{DDA} < 4.5 V
SID113	t_{S_4P5}	Analog input sample time for channels of own SARMUX	412	–	–	ns	4.5 V ≤ V_{DDA} ≤ 5.5 V Guaranteed by design
SID113A	t_{S_2P7}	Analog input sample time for channels of own SARMUX	600	–	–	ns	2.7 V ≤ V_{DDA} < 4.5 V Guaranteed by design
SID113B	$t_{S\_DR\_4P5}$	Analog input sample time when input is from diagnostic reference	2	–	–	µs	4.5 V ≤ V_{DDA} ≤ 5.5 V Guaranteed by design
SID113C	$t_{S\_DR\_2P7}$	Analog input sample time when input is from diagnostic reference	2.5	–	–	µs	2.7 V ≤ V_{DDA} < 4.5 V Guaranteed by design
SID113D	t_{S_TS}	Analog input sample time for temperature sensor	3	–	–	µs	2.7 V ≤ V_{DDA} ≤ 5.5 V Guaranteed by design
SID113E	$t_{S\_4P5\_A}$	Analog input sample time for channels of another SARMUXn (n=1,2)	824	–	–	ns	4.5 V ≤ V_{DDA} ≤ 5.5 V When ADC0 borrows the SARMUX of another ADC, guaranteed by design
SID113F	$t_{S\_2P7\_A}$	Analog input sample time for channels of another SARMUXn (n=1,2)	1648	–	–	ns	2.7V ≤ V_{DDA} ≤ 4.5 V When ADC0 borrows the SARMUX of another ADC, guaranteed by design
SID106	t_{ST_4P5}	ADC maximum throughput (samples per second) when using the SARMUX of own ADC	–	–	1	Msps	4.5 V ≤ V_{DDA} ≤ 5.5 V, 80 MHz/3 = 26.67 MHz, 11 sampling cycles, 15 conversion cycles
SID106A	t_{ST_2P7}	ADC maximum throughput (samples per second) when using the SARMUX of own ADC	–	–	0.5	Msps	2.7 V ≤ V_{DDA} < 4.5 V 80 MHz/6 = 13.3 MHz, 11 sampling cycles, 15 conversion cycles

(table continues...)

Table 33 (continued) SAR ADC AC specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID106B	$t_{ST\_4P5\_A}$	ADC0 maximum throughput (samples per second) when borrowing the SARMUXn of another ADC (n=1,2)	–	–	0.5	Msps	4.5 V ≤ V_{DDA} ≤ 5.5 V, 80 MHz/6 = 13.3 MHz, 11 sampling cycles, 15 conversion cycles
SID106C	$t_{ST\_2P7\_A}$	ADC0 maximum throughput (samples per second) when borrowing the SARMUXn of another ADC (n=1,2)	–	–	0.25	Msps	2.7V ≤ VDDA < 4.5 V, 80 MHz/12 = 6.67 MHz, 11 sampling cycles, 15 conversion cycles
SID107	C_{VIN}	ADC input sampling capacitance	–	–	4.8	pF	Guaranteed by design
SID108	R_{VIN1}	Input path ON resistance (4.5 V to 5.5 V)	–	–	9.4	kΩ	Guaranteed by design
SID108A	R_{VIN2}	Input path ON resistance (2.7 V to 4.5 V)	–	–	13.9	kΩ	Guaranteed by design
SID108B	R_{DREF1}	Diagnostic path ON resistance (4.5 V to 5.5 V)	–	–	40	kΩ	Guaranteed by design
SID108C	R_{DREF2}	Diagnostic path ON resistance (2.7 V to 4.5 V)	–	–	50	kΩ	Guaranteed by design
SID119	ACC_RLAD	Diagnostic reference resistor ladder accuracy	–4	–	4	%	
SID109	A_TE	Total error	–5	–	5	LSb	V_{DDA} = V_{REFH} = 2.7 V to 5.5 V, V_{REFL} = V_{SSA} –40°C ≤ T_A ≤ 125°C Total error after offset and gain adjustment at 12 bit resolution mode
SID109A	A_TEB	Total error	–12	–	12	LSb	V_{DDA} = V_{REFH} = 2.7 V to 5.5 V, V_{REFL} = V_{SSA} –40°C ≤ T_A ≤ 125°C Total error before offset and gain adjustment at 12 bit resolution mode
SID110	A_INL	Integral nonlinearity	–2.5	–	2.5	LSb	V_{DDA} = 2.7 V to 5.5 V, –40°C ≤ T_A ≤ 125°C
SID111	A_DNL	Differential nonlinearity	–0.99	–	1.9	LSb	V_{DDA} = 2.7 V to 5.5 V, –40°C ≤ T_A ≤ 125°C

(table continues...)

Table 33 (continued) SAR ADC AC specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID112	A_CE	Channel-to-channel variation (for channels connected to same ADC)	–1	–	1	LSb	V_{DDA} = 2.7 V to 5.5 V, –40°C ≤ T_A ≤ 125°C
SID115	I_{AIC}	Analog input leakage current	–350	70	350	nA	When input pad is selected for conversion
SID116	$I_{DIAGREF}$	Diagnostic reference current	–	–	70	µA	
SID117	I_{VDDA}	Analog power supply current while ADC is operating	–	360	550	µA	Per enabled ADC
SID117A	I_{VDDA_DS}	Analog power supply current while ADC is not operating	–	–	21	µA	Per enabled ADC
SID118	I_{VREF}	Analog reference voltage current while ADC is operating	–	360	550	µA	Per enabled ADC
SID118A	I_{VREF_LEAK}	Analog reference voltage current while ADC is not operating	–	1.8	5	µA	Per enabled ADC

26.6.3 Temperature sensor

Table 34 Temperature sensor specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID201	$T_{SENSACC2}$	Temperature sensor accuracy 2	–5	–	5	°C	–40°C ≤ T_J ≤ 150°C This spec is valid when using ADC[0] (V_{DDIO_1}), ADC[1] (V_{DDIO_2}) or ADC[2] (V_{DDD}) with the following conditions: a. 3.0 V ≤ V_{DDD}, V_{DDIO_1} or V_{DDIO_2} = V_{DDA} = V_{REFH} ≤ 3.6 V or b. 4.5 V ≤ V_{DDD}, V_{DDIO_1} or V_{DDIO_2} = V_{DDA} = V_{REFH} ≤ 5.5 V
SID201A	$T_{SENSACC3}$	Temperature sensor accuracy 3	–10	–	10	°C	–40°C ≤ T_J ≤ 150°C This spec is valid when using ADC[0] (V_{DDIO_1}) or ADC[2] (V_{DDD}) with the following condition: 2.7 V ≤ V_{DDD} or V_{DDIO_1} ≤ 5.5 V and 2.7 V ≤ V_{DDA} = V_{REFH} ≤ 5.5 V and 0.8 × V_{DDA} < V_{DDD} or V_{DDIO_1}

26.6.4 Voltage divider accuracy

Table 35 Voltage divider accuracy

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID202	V_{MONDIV}	Uncorrected monitor voltage divider accuracy (measured by ADC), compared to ideal supply/2	–20	2	20	%	Any HV supply pad within 2.7 V - 5.5 V operating range

26.7 AC specifications

Unless otherwise noted, the timings are defined with the guidelines mentioned in the Figure 17.

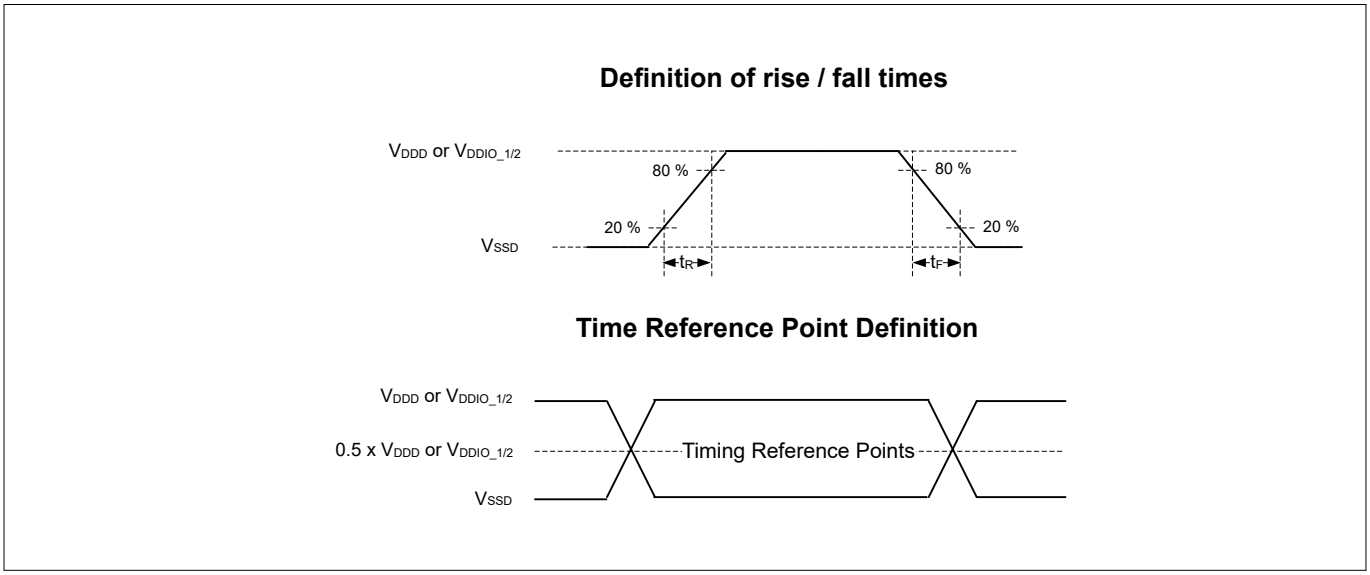


Figure 17 AC timings specifications

26.8 Digital peripherals

All specifications are valid for –40°C ≤ T_A ≤ 125°C and for 2.7 V to 5.5 V except where noted.

Table 36 Timer/counter/PWM (TCPWM) specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID120	f_C	TCPWM operating frequency	–	–	100	MHz	f_C = peripheral clock
SID121	$t_{PWMENEXT}$	Input trigger pulse width for all trigger events	$2/f_C$	–	–	ns	Trigger Events can be Stop, Start, Reload, Count, Capture, or Kill depending on which mode of operation is selected.

(table continues...)

Table 36 (continued) Timer/counter/PWM (TCPWM) specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID122	t_{PWMEXT}	Output trigger pulse widths	$2/f_C$	–	–	ns	Minimum possible width of Overflow, Underflow, and Counter = Compare (CC) value trigger outputs
SID123	t_{CRES}	Resolution of counter	$1/f_C$	–	–	ns	Minimum time between successive counts
SID124	t_{PWMRES}	PWM resolution	$1/f_C$	–	–	ns	Minimum pulse width of PWM output
SID125	t_{QRES}	Quadrature inputs resolution	$2/f_C$	–	–	ns	Minimum pulse width between Quadrature phase inputs.

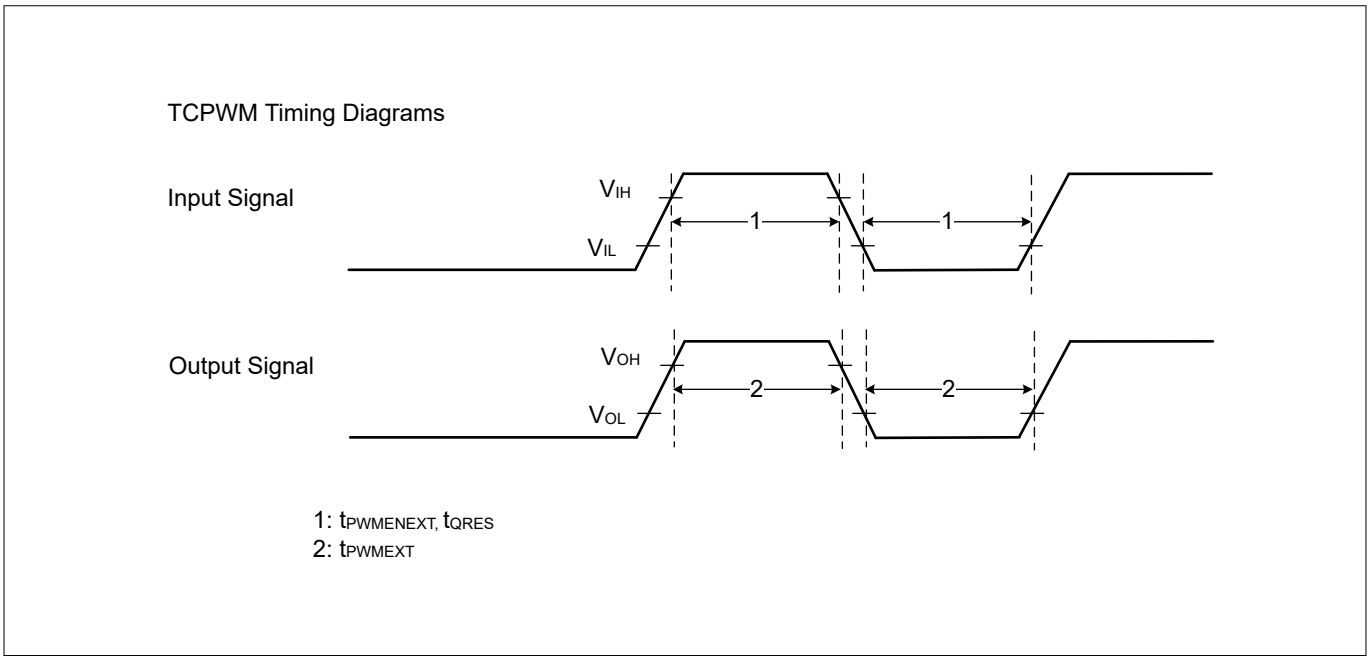


Figure 18 TCPWM timing diagrams

Table 37 Serial communication block (SCB) specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID129	f_{SCB}	SCB operating frequency	–	–	100	MHz	
I2C interface - standard mode							
SID130	f_{SCL}	SCL clock frequency	–	–	100	kHz	
SID131	$t_{HD;STA}$	Hold time, START condition	4000	–	–	ns	
SID132	t_{LOW}	Low period of SCL	4700	–	–	ns	

(table continues...)

Table 37 (continued) Serial communication block (SCB) specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID133	t_{HIGH}	High period of SCL	4000	–	–	ns	
SID134	$t_{SU;STA}$	Setup time for a repeated START	4700	–	–	ns	
SID135	$t_{HD;DAT}$	Data hold time, for receiver	0	–	–	ns	
SID136	$t_{SU;DAT}$	Data setup time	250	–	–	ns	
SID138	t_F	Fall time of SCL and SDA	–	–	300	ns	Input and output
SID139	$t_{SU;STO}$	Setup time for STOP	4000	–	–	ns	
SID140	t_{BUF}	Bus-free time between START and STOP	4700	–	–	ns	
SID141	C_B	Capacitive load for each bus line	–	–	400	pF	
SID142	$t_{VD;DAT}$	Time for data signal from SCL LOW to SDA output	–	–	3450	ns	
SID143	$t_{VD;ACK}$	Data valid acknowledge time	–	–	3450	ns	
SID144	V_{OL}	LOW level output voltage	0	–	0.4	V	Open-drain at 3 mA sink current
SID145	I_{OL}	LOW level output current	3	–	–	mA	V_{OL} = 0.4 V
I2C interface-fast-mode							
SID150	f_{SCL_F}	SCL clock frequency	–	–	400	kHz	
SID151	$t_{HD;STA\_F}$	Hold time, START condition	600	–	–	ns	
SID152	t_{LOW_F}	Low period of SCL	1300	–	–	ns	
SID153	t_{HIGH_F}	High period of SCL	600	–	–	ns	
SID154	$t_{SU;STA\_F}$	Setup time for a repeated START	600	–	–	ns	
SID155	$t_{HD;DAT\_F}$	Data hold time, for receiver	0	–	–	ns	
SID156	$t_{SU;DAT\_F}$	Data setup time	100	–	–	ns	
SID158	t_{F_F}	Fall time of SCL and SDA	20 × (V_{DDD} / 5.5)	–	300	ns	Input and output, GPIO_ENH: slow mode, 400 pF load

(table continues...)

Table 37 (continued) Serial communication block (SCB) specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID158A	t_{FA_F}	Fall time of SCL and SDA	0.35	–	300	ns	Input and output GPIO_STD: drive_sel<1:0>= 0b00 MIN: 10 pF load, RPU = 35.41 kΩ MAX: 400 pF load, RPU = 350 Ω
SID159	$t_{SU;STO\_F}$	Setup time for STOP	600	–	–	ns	Input and output
SID160	t_{BUF_F}	Bus free time between START and STOP	1300	–	–	ns	
SID161	C_{B_F}	Capacitive load for each bus line	–	–	400	pF	
SID162	$t_{VD;DAT\_F}$	Time for data signal from SCL LOW to SDA output	–	–	900	ns	
SID163	$t_{VD;ACK\_F}$	Data valid acknowledge time	–	–	900	ns	
SID164	t_{SP_F}	Pulse width of spikes that must be suppressed by the input filter	–	–	50	ns	
SID165	V_{OL_F}	LOW level output voltage	0	–	0.4	V	Open-drain at 3 mA sink current
SID165	I_{OL_F}	LOW level output current	3	–	–	mA	V_{OL} = 0.4 V
SID167	I_{OL2_F}	LOW level output current	6	–	–	mA	V_{OL} = 0.6 V[43]
I2C interface-fast-plus mode							
SID170	f_{SCL_FP}	SCL clock frequency	–	–	1	MHz	
SID171	$t_{HD;STA\_FP}$	Hold time, START condition	260	–	–	ns	
SID172	t_{LOW_FP}	Low period of SCL	500	–	–	ns	
SID173	t_{HIGH_FP}	High period of SCL	260	–	–	ns	
SID174	$t_{SU;STA\_FP}$	Setup time for a repeated START	260	–	–	ns	
SID175	$t_{HD;DAT\_FP}$	Data hold time, for receiver	0	–	–	ns	
SID176	$t_{SU;DAT\_FP}$	Data setup time	50	–	–	ns	

(table continues...)

[43] To drive full bus load at 400 kHz, 6 mA I_{OL} is required at 0.6 V V_{OL}.

Table 37 (continued) Serial communication block (SCB) specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID178	t_{F_FP}	Fall time of SCL and SDA	20 × (V_{DDD} / 5.5)	–	160	ns	Input and output 20-pF load GPIO_ENH: slow mode
SID179	$t_{SU;STO\_FP}$	Setup time for STOP	260	–	–	ns	Input and output
SID180	t_{BUF_FP}	Bus free time between START and STOP	500	–	–	ns	
SID181	C_{B_FP}	Capacitive load for each bus line	–	–	20	pF	
SID182	$t_{VD;DAT\_FP}$	Time for data signal from SCL LOW to SDA output	–	–	450	ns	
SID183	$t_{VD;ACK\_FP}$	Data valid acknowledge time	–	–	450	ns	
SID184	t_{SP_FP}	Pulse width of spikes that must be suppressed by the input filter	–	–	50	ns	
SID186	V_{OL_FP}	LOW level output voltage	0	–	0.4	V	Open-drain at 3 mA sink current
SID187	I_{OL_FP}	LOW level output current	3	–	–	mA	V_{OL} = 0.4 V[44]
SPI interface master (Full-clock mode: LATE_MISO_SAMPLE = 1) [Conditions: drive_sel<1:0>= 0x]							
SID190	f_{SPI}	SPI operating frequency	–	–	12.5	MHz	Do not use half-clock mode: LATE_MISO_SAMPLE = 0
SID191	t_{DMO}	SPI Master: MOSI valid after SCLK driving edge	–	–	15	ns	
SID192	t_{DSI}	SPI Master: MISO valid before SCLK capturing edge	40	–	–	ns	
SID193	t_{HMO}	SPI Master: Previous MOSI data hold time	0	–	–	ns	
SID194	$t_{W\_SCLK\_H\_L}$	SPI SCLK pulse width HIGH or LOW	–	0.4 × ($1/f_{SPI}$)	–	ns	
SID196	t_{DHI}	SPI Master: MISO hold time after SCLK capturing edge	0	–	–	ns	
SID198	t_{EN_SETUP}	SSEL valid, before the first SCK capturing edge	0.5 × ($1/f_{SPI}$)	–	–	ns	Min is half clock period

(table continues...)

[44] To drive full bus load at 1 MHz, 20 mA I_{OL} is required at 0.4 V V_{OL}. However, this device does not support it.

Table 37 (continued) Serial communication block (SCB) specifications

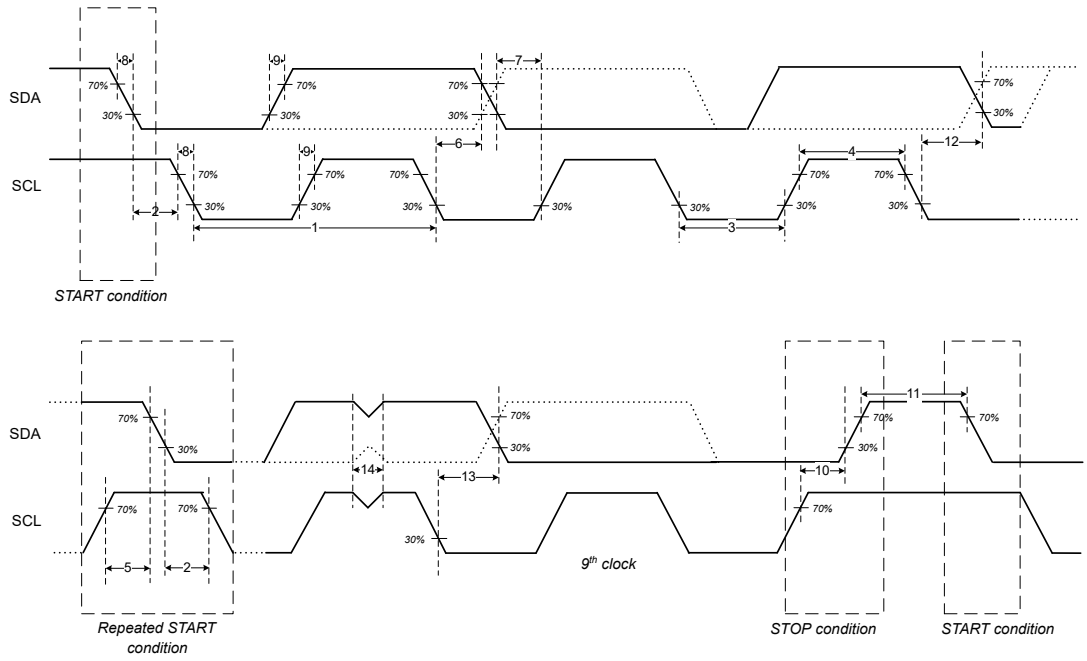
Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID199	t_{EN_SHOLD}	SSEL hold, after the last SCK capturing edge	0.5 × (1/f_{SPI})	–	–	ns	Min is half clock period
SID195	C_{SPIM_MS}	SPI capacitive load	–	–	10	pF	
SPI interface slave (internally clocked) [Conditions: drive_sel<1:0>= 0x]							
SID205	f_{SPI_INT}	SPI operating frequency	–	–	10	MHz	
SID206	t_{DMI_INT}	SPI Slave: MOSI valid before Sclock capturing edge	5	–	–	ns	
SID207	t_{DSO_INT}	SPI Slave: MISO valid after Sclock driving edge, in the internal-clocked mode	–	–	62	ns	
SID208	t_{HSP}	SPI Slave: Previous MISO data hold time	3	–	–	ns	
SID209	$t_{EN\_SETUP\_INT}$	SPI Slave: SSEL valid to first SCK valid edge	33	–	–	ns	
SID210	$t_{EN\_HOLD\_INT}$	SPI Slave Select active (LOW) from last SCLK hold	33	–	–	ns	
SID211	$t_{EN\_SETUP\_PRE}$	SPI Slave: from SSEL valid, to SCK falling edge before the first data bit	20	–	–	ns	
SID212	$t_{EN\_HOLD\_PRE}$	SPI Slave: from SCK falling edge before the first data bit, to SSEL invalid	20	–	–	ns	
SID213	$t_{EN\_SETUP\_CO}$	SPI Slave: from SSEL valid, to SCK falling edge in the first data bit	20	–	–	ns	
SID214	$t_{EN\_HOLD\_CO}$	SPI Slave: from SCK falling edge in the first data bit, to SSEL invalid	20	–	–	ns	
SID215	$t_{W\_DIS\_INT}$	SPI Slave Select inactive time	40	–	–	ns	
SID216	$t_{W\_SCLKH\_INT}$	SPI SCLK pulse width HIGH	20	–	–	ns	
SID217	$t_{W\_SCLKL\_INT}$	SPI SCLK pulse width LOW	20	–	–	ns	
SID218	t_{SIH_INT}	SPI MOSI hold from SCLK	12	–	–	ns	
SID219	C_{SPIS_INT}	SPI Capacitive Load	–	–	10	pF	
SPI interface slave (externally clocked) [Conditions: drive_sel<1:0>= 0x]							
SID220	f_{SPI_EXT}	SPI operating frequency	–	–	12.5	MHz	

(table continues...)

Table 37 (continued) Serial communication block (SCB) specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID221	t_{DMI_EXT}	SPI Slave: MOSI valid before Sclock capturing edge	5	–	–	ns	
SID222	t_{DSO_EXT}	SPI Slave: MISO valid after Sclock driving edge, in the external-clocked mode	–	–	32	ns	
SID223	t_{HSO_EXT}	SPI Slave: Previous MISO data hold time	3	–	–	ns	
SID224	$t_{EN\_SETUP\_EXT}$	SPI Slave: SSEL valid to first SCK valid edge	40	–	–	ns	
SID225	$t_{EN\_HOLD\_EXT}$	SPI Slave Select active (LOW) from last SCLK hold	40	–	–	ns	
SID226	$t_{W\_DIS\_EXT}$	SPI Slave Select inactive time	80	–	–	ns	
SID227	$t_{W\_SCLKH\_EXT}$	SPI SCLK pulse width HIGH	34	–	–	ns	
SID228	$t_{W\_SCLKL\_EXT}$	SPI SCLK pulse width LOW	34	–	–	ns	
SID229	t_{SIH_EXT}	SPI MOSI hold from SCLK	20	–	–	ns	
SID230	C_{SPIS_EXT}	SPI capacitive load	–	–	10	pF	
SID231	t_{VSS_EXT}	SPI Slave: MISO valid after SSEL falling edge (CPHA = 0)	–	–	33	ns	
UART interface							
SID240	f_{BPS}	Data rate	–	–	10	Mbps	

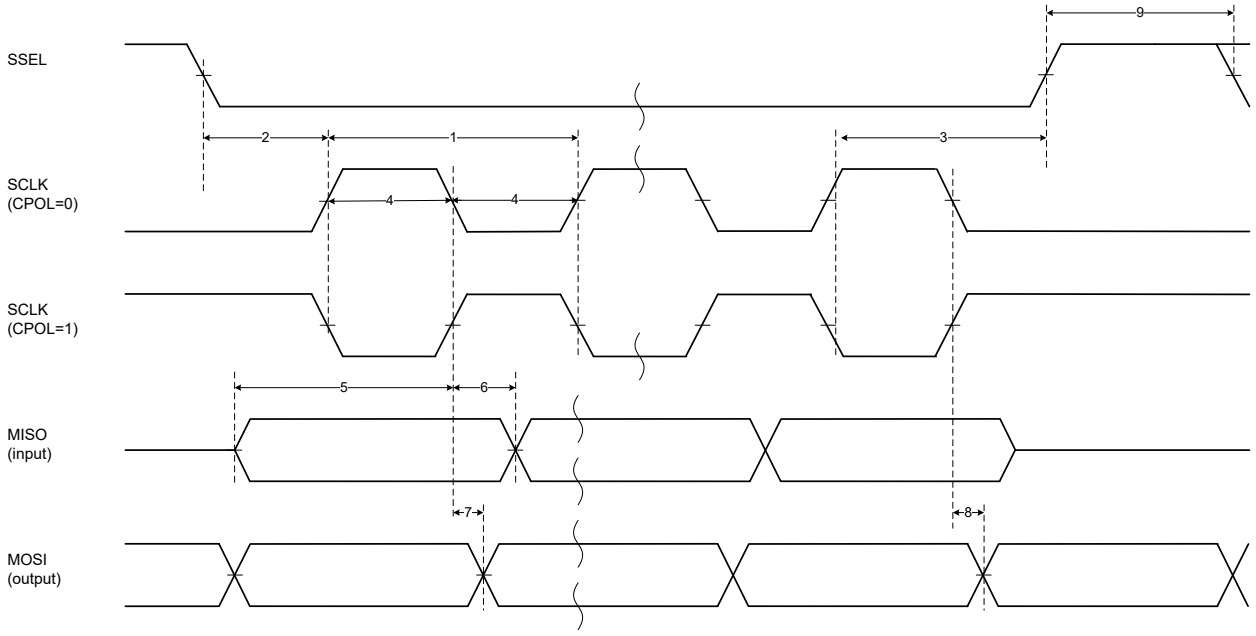
26 Electrical specifications



1: SCL clock period = $1/f_{SCL}$
2: Hold time, START condition = $t_{HD;STA}$
3: LOW period of SCL = t_{LOW}
4: HIGH period of SCL = t_{HIGH}
5: Setup time for a repeated START = $t_{SU;STA}$
6: Data hold time, for receiver = $t_{HD;DAT}$
7: Data setup time = $t_{SU;DAT}$
8: Fall time of SCL and SDA = t_F
9: Rise time of SCL and SDA = t_R
10: Setup time for STOP = $t_{SU;STO}$
11: Bus-free time between START and STOP = t_{BUF}
12: Time for data signal from SCL LOW to SDA output = $t_{VD;DAT}$
13: Data valid acknowledge time = $t_{VD;ACK}$
14: Pulse width of spikes that must be suppressed by the input filter = t_{SP}

Figure 19 I2C timing diagrams

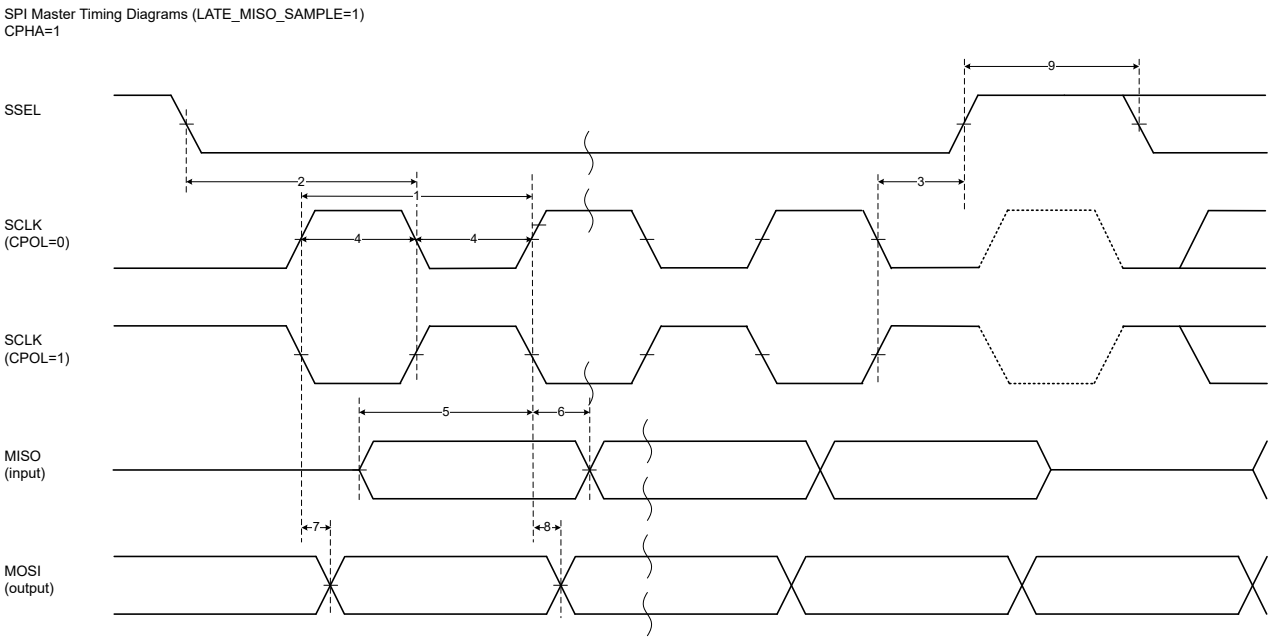
26 Electrical specifications



1: SCLK period = 1 / f_{SPI}
2: Enable lead time (setup) = t_{EN_SETUP} = Depends on SPI_CTRL.SSEL_SETUP_DEL (Refer to the Register TRM)
3: Enable trail time (hold) = t_{EN_HOLD} = Depends on SPI_CTRL.SSEL_HOLD_DEL (Refer to the Register TRM)
4: SCLK high or low time = $t_{W\_SCLK\_H\_L}$
5: Input data setup time = t_{DSI}
6: Input data hold time = t_{DHI}
7: Output data valid after SCLK driving edge = t_{DMO}
8: Output data hold time = t_{HMO}
9: SSEL high pulse width = Depends on SPI_CTRL.SSEL_INTER_FRAME_DEL (Refer to the Register TRM)

Figure 20 SPI master timing diagrams with LOW clock phase

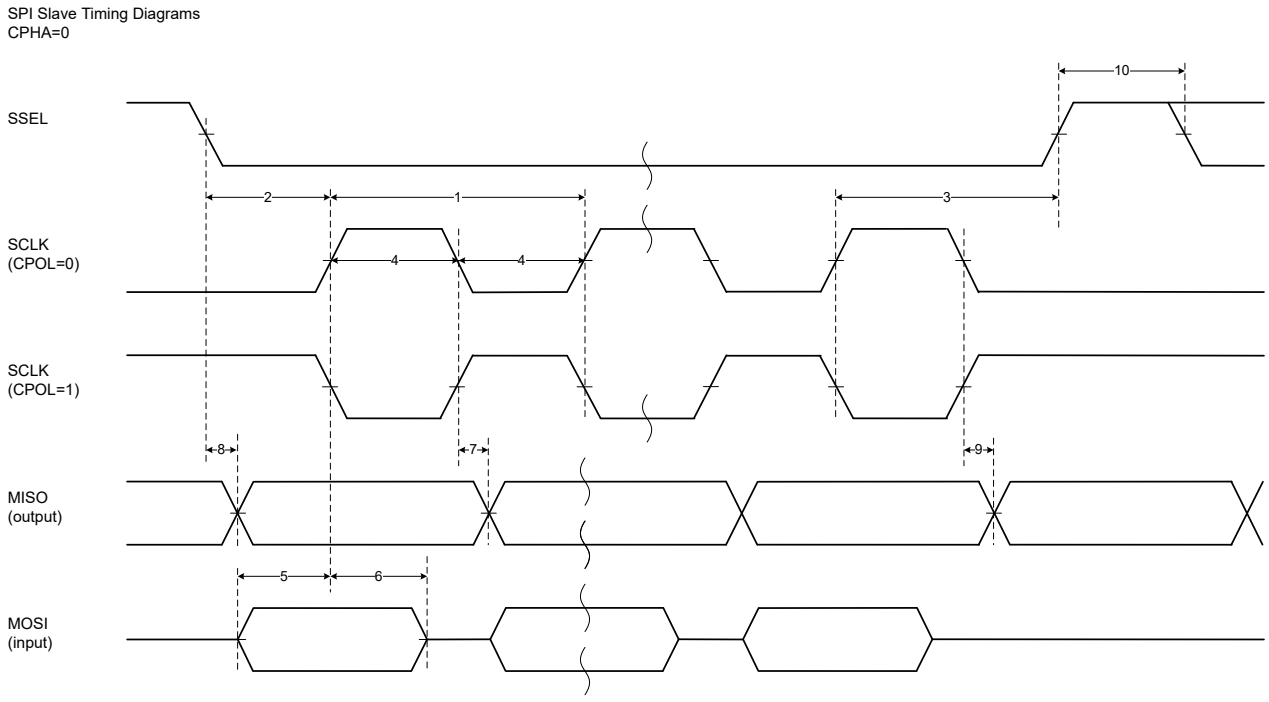
26 Electrical specifications



1: SCLK period = 1 / f_{SPI}
2: Enable lead time (setup) = t_{EN_SETUP} = Depends on SPI_CTRL.SSEL_SETUP_DEL (Refer to the Register TRM)
3: Enable trail time (hold) = t_{EN_HOLD} = Depends on SPI_CTRL.SSEL_HOLD_DEL (Refer to the Register TRM)
4: SCLK high or low time = $t_{W\_SCLK\_H\_L}$
5: Input data setup time = t_{DSI}
6: Input data hold time = t_{HDI}
7: Output data valid after SCLK driving edge = t_{DMO}
8: Output data hold time = t_{HMO}
9: SSEL high pulse width = Depends on SPI_CTRL.SSEL_INTER_FRAME_DEL (Refer to the Register TRM)

Figure 21 SPI master timing diagrams with HIGH clock phase

26 Electrical specifications



1: SCLK period = 1 / f_{SPI_EXT}
2: enable lead time (setup) = $t_{EN\_SETUP\_EXT}$
3: enable trail time (hold) = $t_{EN\_HOLD\_EXT}$
4: SCLK high or low time = $t_{w\_SCLKH\_EXT}$ = $t_{w\_SCLKL\_EXT}$
5: input data setup time = t_{DMI_EXT}
6: input data hold time = t_{SIH_EXT}
7: output data valid after SCLK driving edge = t_{DSO_EXT}
8: output data valid after SSEL falling edge (CPHA=0) = t_{VSS_EXT}
9: output data hold time = t_{HSO}
10: SSEL high pulse width = t_{DIS_EXT}

Figure 22 SPI slave timing diagrams with LOW clock phase

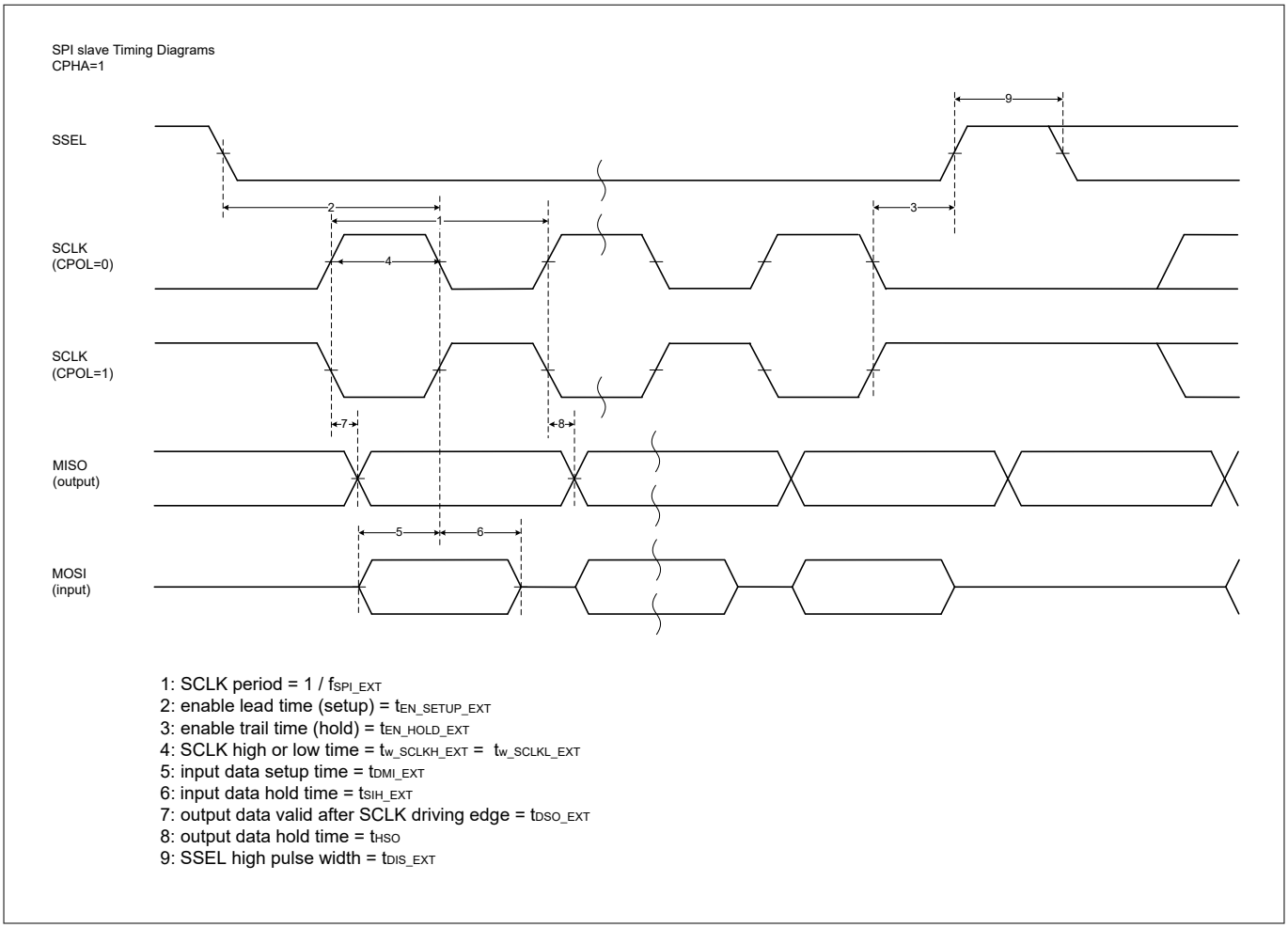


Figure 23 SPI slave timing diagrams with HIGH clock phase

26.8.1 CAN FD specifications

Table 38 CAN FD specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID630	f_{HCLK}	System clock frequency	–	–	100	MHz	$f_{cclk} \le f_{hclk}$, Guaranteed by design
SID631	f_{CCLK}	CAN clock frequency	–	–	100	MHz	$f_{cclk} \le f_{hclk}$, Guaranteed by design

26.9 Memory

All specifications are valid for –40°C ≤ T_A ≤ 125°C and for 2.7 V to 5.5 V except where noted.

Table 39 Flash DC specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID260	V_{PE}	Erase and program voltage	2.7	–	5.5	V	

Table 40 Flash AC specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID257	f_{FO}	Maximum flash memory operation frequency	–	–	100	MHz	Zero wait access to code-flash memory up to 100 MHz Zero wait access with cache hit up to 160 MHz
SID254	t_{ERS_SUS}	Maximum time from erase suspend command till erase is indeed suspend	–	–	37.5	µs	
SID255	$t_{ERS\_RES\_SUS}$	Minimum time allowed from erase resume to erase suspend	250	–	–	µs	Guaranteed by design
SID258	t_{BC_WF}	Blank check time for N-bytes of work-flash	–	–	10 + 0.3 × N	µs	At 100 MHz, N ≥ 4 and multiple of 4, excludes system overhead time
SID259	$t_{SECTORERASE1}$	Sector erase time (Code-flash: 32 KB)	–	45	90	ms	Includes internal preprogramming time
SID259A	$t_{SECTORERASE2}$	Sector erase time (Code-flash: 8 KB)	–	15	30	ms	Includes internal preprogramming time
SID261	$t_{SECTORERASE3}$	Sector erase time (Work-flash, 2 KB)	–	80	160	ms	Includes internal preprogramming time
SID262	$t_{SECTORERASE4}$	Sector erase time (Work-flash, 128 bytes)	–	5	15	ms	Includes internal preprogramming time
SID263	t_{WRITE1}	64-bit write time (Code-flash)	–	30	60	µs	Excludes system overhead time
SID264	t_{WRITE2}	256-bit write time (Code-flash)	–	40	70	µs	Excludes system overhead time
SID265	t_{WRITE3}	4096-bit write time (Code-flash)[45]	–	320	1200	µs	Excludes system overhead time
SID266	t_{WRITE4}	32-bit write time (Work-flash)	–	30	60	µs	Excludes system overhead time
SID267	t_{FRET1}	Code-flash retention. 1000 program/erase cycles	20	–	–	years	T_A (power on and off) ≤ 85°C average
SID268	t_{FRET3}	Work-flash retention. 125,000 program/erase cycles	20	–	–	years	T_A (power on and off) ≤ 85°C average

(table continues...)

45 The code-flash includes a 'Write Buffer' of 4096-bit. If the application software writes this buffer multiple times, to get the overall write time multiply one sector write time with the corresponding factor (say for factor 64, example, 64 x 512 B = 32 KB [one sector].

Table 40 (continued) Flash AC specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID269	t_{FRET4}	Work-flash retention. 250,000 program/erase cycles	10	–	–	years	T_A (power on and off) ≤ 85°C average
SID612	I_{CC_ACT2}	Program operating current (Code or Work-flash)	–	15	48	mA	V_{DDD} = 5 V Guaranteed by design
SID613	I_{CC_ACT3}	Erase operating current (Code or Work-flash)	–	15	48	mA	V_{DDD} = 5 V Guaranteed by design

26.10 System resources

All specifications are valid for –40°C ≤ T_A ≤ 125°C and for 2.7 V to 5.5 V except where noted.

Table 41 System resources

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/ conditions
Power-on-reset specifications							
SID270	V_{POR_R}	V_{DDD} rising voltage to de assert POR	1.5	–	2.35	V	Guaranteed by design
SID276	V_{POR_F}	V_{DDD} falling voltage to assert POR	1.45	–	2.1	V	
SID271	V_{POR_H}	Level detection hysteresis	20	–	300	mV	Guaranteed by design
SID272	t_{DLY_POR}	Delay between V_{DDD} rising through 2.3 V and an internal deassertion of POR	–	–	3	µs	Guaranteed by design
SID273	t_{POFF}	V_{DDD} Power off time	100	–	–	µs	V_{DDD} < 1.45 V
SID274	POR_RR1	V_{DDD} power ramp rate with robust BOD (BOD operation is guaranteed)	–	–	100	mV/µs	This ramp supports robust BOD
SID275	POR_RR2	V_{DDD} power ramp rate without robust BOD	–	–	1000	mV/µs	This ramp does not support robust BOD t_{POFF} must be satisfied
High-voltage BOD (HV BOD) specifications							
SID500	$V_{TR\_2P7\_R}$	HV BOD 2.7 V rising detection point for V_{DDD} and V_{DDA} (default)	2.474	2.55	2.627	V	
SID501	$V_{TR\_2P7\_F}$	HV BOD 2.7 V falling detection point for V_{DDD} and V_{DDA} (default)	2.449	2.525	2.601	V	

(table continues...)

Table 41 (continued) System resources

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/ conditions
SID502	$V_{TR\_3P0\_R}$	HV BOD 3.0 V rising detection point for V_{DDD} and V_{DDA}	2.765	2.85	2.936	V	
SID503	$V_{TR\_3P0\_F}$	HV BOD 3.0 V falling detection point for V_{DDD} and V_{DDA}	2.74	2.825	2.91	V	
SID505	HVBOD_RR_A	Power ramp rate: V_{DDD} and V_{DDA} (Active)	–	–	100	mV/µs	
SID506	HVBOD_RR_DS	Power ramp rate: V_{DDD} and V_{DDA} (DeepSleep)	–	–	10	mV/µs	
SID507	$t_{DLY\_ACT\_HVBOD}$	Active mode delay between V_{DDD} falling/rising through $V_{TR\_2P7\_F/R}$ or $V_{TR\_3P0\_F/R}$ and an internal HV BOD signal transitioning	–	–	0.5	µs	Guaranteed by design
SID507A	$t_{DLY\_ACT\_HVBOD}$	Active mode delay between V_{DDA} falling/rising through $V_{TR\_2P7\_F/R}$ or $V_{TR\_3P0\_F/R}$ and internal HV BOD signal transitioning	–	–	1	µs	Guaranteed by design
SID507B	$t_{DLY\_DS\_HVBOD}$	DeepSleep mode delay between V_{DDD}/V_{DDA} falling/ rising through $V_{TR\_2P7\_F/R}$ or $V_{TR\_3P0\_F/R}$ and an internal HV BOD signal transitioning	–	–	4	µs	Guaranteed by design
SID508	t_{RES_HVBOD}	Response time of HV BOD, V_{DDD}/V_{DDA} supply. (For falling-then-rising supply at maximum ramp rate; threshold is $V_{TR\_2P7\_F}$ or $V_{TR\_3P0\_F}$.)	100	–	–	ns	Guaranteed by design
Low-voltage BOD (LV BOD) specifications							
SID510	$V_{TR\_R\_LVBOD}$	LV BOD rising detection point for V_{CCD}	0.917	0.945	0.973	V	
SID511	$V_{TR\_F\_LVBOD}$	LV BOD falling detection point for V_{CCD}	0.892	0.92	0.948	V	
SID515	$t_{DLY\_ACT\_LVBOD}$	Active delay between V_{CCD} falling/rising through V_{TR_R/F_LVBOD} and an internal LV BOD signal transitioning	–	–	1	µs	Guaranteed by design
SID515A	$t_{DLY\_DS\_LVBOD}$	DeepSleep mode delay between V_{CCD} falling/rising through V_{TR_R/F_LVBOD} and an internal LV BOD signal transitioning	–	–	12	µs	Guaranteed by design

(table continues...)

Table 41 (continued) System resources

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/ conditions
SID516	t_{RES_LVBOD}	Response time of LV BOD. (For falling-then-rising supply at maximum ramp rate; threshold is $V_{TR\_F\_LVBOD}$.)	100	–	–	ns	Guaranteed by design
Low-voltage detector (LVD) DC specifications							
SID520	$V_{TR\_2P8\_F}$	LVD 2.8 V falling detection point for V_{DDD}	Typ – 4%	2800	Typ + 4%	mV	
SID521	$V_{TR\_2P9\_F}$	LVD 2.9 V falling detection point for V_{DDD}	Typ – 4%	2900	Typ + 4%	mV	
SID522	$V_{TR\_3P0\_F}$	LVD 3.0 V falling detection point for V_{DDD}	Typ – 4%	3000	Typ + 4%	mV	
SID523	$V_{TR\_3P1\_F}$	LVD 3.1 V falling detection point for V_{DDD}	Typ – 4%	3100	Typ + 4%	mV	
SID524	$V_{TR\_3P2\_F}$	LVD 3.2 V falling detection point for V_{DDD}	Typ – 4%	3200	Typ + 4%	mV	
SID525	$V_{TR\_3P3\_F}$	LVD 3.3 V falling detection point for V_{DDD}	Typ – 4%	3300	Typ + 4%	mV	
SID526	$V_{TR\_3P4\_F}$	LVD 3.4 V falling detection point for V_{DDD}	Typ – 4%	3400	Typ + 4%	mV	
SID527	$V_{TR\_3P5\_F}$	LVD 3.5 V falling detection point for V_{DDD}	Typ – 4%	3500	Typ + 4%	mV	
SID528	$V_{TR\_3P6\_F}$	LVD 3.6 V falling detection point for V_{DDD}	Typ – 4%	3600	Typ + 4%	mV	
SID529	$V_{TR\_3P7\_F}$	LVD 3.7 V falling detection point for V_{DDD}	Typ – 4%	3700	Typ + 4%	mV	
SID530	$V_{TR\_3P8\_F}$	LVD 3.8 V falling detection point for V_{DDD}	Typ – 4%	3800	Typ + 4%	mV	
SID531	$V_{TR\_3P9\_F}$	LVD 3.9 V falling detection point for V_{DDD}	Typ – 4%	3900	Typ + 4%	mV	
SID532	$V_{TR\_4P0\_F}$	LVD 4.0 V falling detection point for V_{DDD}	Typ – 4%	4000	Typ + 4%	mV	
SID533	$V_{TR\_4P1\_F}$	LVD 4.1 V falling detection point for V_{DDD}	Typ – 4%	4100	Typ + 4%	mV	
SID534	$V_{TR\_4P2\_F}$	LVD 4.2 V falling detection point for V_{DDD}	Typ – 4%	4200	Typ + 4%	mV	
SID535	$V_{TR\_4P3\_F}$	LVD 4.3 V falling detection point for V_{DDD}	Typ – 4%	4300	Typ + 4%	mV	
SID536	$V_{TR\_4P4\_F}$	LVD 4.4 V falling detection point for V_{DDD}	Typ – 4%	4400	Typ + 4%	mV	

(table continues...)

Table 41 (continued) System resources

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/ conditions
SID537	$V_{TR\_4P5\_F}$	LVD 4.5 V falling detection point for V_{DDD}	Typ – 4%	4500	Typ + 4%	mV	
SID538	$V_{TR\_4P6\_F}$	LVD 4.6 V falling detection point for V_{DDD}	Typ – 4%	4600	Typ + 4%	mV	
SID539	$V_{TR\_4P7\_F}$	LVD 4.7 V falling detection point for V_{DDD}	Typ – 4%	4700	Typ + 4%	mV	
SID540	$V_{TR\_4P8\_F}$	LVD 4.8 V falling detection point for V_{DDD}	Typ – 4%	4800	Typ + 4%	mV	
SID541	$V_{TR\_4P9\_F}$	LVD 4.9 V falling detection point for V_{DDD}	Typ – 4%	4900	Typ + 4%	mV	
SID542	$V_{TR\_5P0\_F}$	LVD 5.0 V falling detection point for V_{DDD}	Typ – 4%	5000	Typ + 4%	mV	
SID543	$V_{TR\_5P1\_F}$	LVD 5.1 V falling detection point for V_{DDD}	Typ – 4%	5100	Typ + 4%	mV	
SID544	$V_{TR\_5P2\_F}$	LVD 5.2 V falling detection point for V_{DDD}	Typ – 4%	5200	Typ + 4%	mV	
SID545	$V_{TR\_5P3\_F}$	LVD 5.3 V falling detection point for V_{DDD}	Typ – 4%	5300	Typ + 4%	mV	
SID546	$V_{TR\_2P8\_R}$	LVD 2.8 V rising detection point for V_{DDD}	Typ – 4%	2825	Typ + 4%	mV	Same as $V_{TR\_2P8\_F}$ + 25 mV
SID547	$V_{TR\_2P9\_R}$	LVD 2.9 V rising detection point for V_{DDD}	Typ – 4%	2925	Typ + 4%	mV	Same as $V_{TR\_2P9\_F}$ + 25 mV
SID548	$V_{TR\_3P0\_R}$	LVD 3.0 V rising detection point for V_{DDD}	Typ – 4%	3025	Typ + 4%	mV	Same as $V_{TR\_3P0\_F}$ + 25 mV
SID549	$V_{TR\_3P1\_R}$	LVD 3.1 V rising detection point for V_{DDD}	Typ – 4%	3125	Typ + 4%	mV	Same as $V_{TR\_3P1\_F}$ + 25 mV
SID550	$V_{TR\_3P2\_R}$	LVD 3.2 V rising detection point for V_{DDD}	Typ – 4%	3225	Typ + 4%	mV	Same as $V_{TR\_3P2\_F}$ + 25 mV
SID551	$V_{TR\_3P3\_R}$	LVD 3.3 V rising detection point for V_{DDD}	Typ – 4%	3325	Typ + 4%	mV	Same as $V_{TR\_3P3\_F}$ + 25 mV
SID552	$V_{TR\_3P4\_R}$	LVD 3.4 V rising detection point for V_{DDD}	Typ – 4%	3425	Typ + 4%	mV	Same as $V_{TR\_3P4\_F}$ + 25 mV

(table continues...)

Table 41 (continued) System resources

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/ conditions
SID553	$V_{TR\_3P5\_R}$	LVD 3.5 V rising detection point for V_{DDD}	Typ – 4%	3525	Typ + 4%	mV	Same as $V_{TR\_3P5\_F}$ + 25 mV
SID554	$V_{TR\_3P6\_R}$	LVD 3.6 V rising detection point for V_{DDD}	Typ – 4%	3625	Typ + 4%	mV	Same as $V_{TR\_3P6\_F}$ + 25 mV
SID555	$V_{TR\_3P7\_R}$	LVD 3.7 V rising detection point for V_{DDD}	Typ – 4%	3725	Typ + 4%	mV	Same as $V_{TR\_3P7\_F}$ + 25 mV
SID556	$V_{TR\_3P8\_R}$	LVD 3.8 V rising detection point for V_{DDD}	Typ – 4%	3825	Typ + 4%	mV	Same as $V_{TR\_3P8\_F}$ + 25 mV
SID557	$V_{TR\_3P9\_R}$	LVD 3.9 V rising detection point for V_{DDD}	Typ – 4%	3925	Typ + 4%	mV	Same as $V_{TR\_3P9\_F}$ + 25 mV
SID558	$V_{TR\_4P0\_R}$	LVD 4.0 V rising detection point for V_{DDD}	Typ – 4%	4025	Typ + 4%	mV	Same as $V_{TR\_4P0\_F}$ + 25 mV
SID559	$V_{TR\_4P1\_R}$	LVD 4.1 V rising detection point for V_{DDD}	Typ – 4%	4125	Typ + 4%	mV	Same as $V_{TR\_4P1\_F}$ + 25 mV
SID560	$V_{TR\_4P2\_R}$	LVD 4.2 V rising detection point for V_{DDD}	Typ – 4%	4225	Typ + 4%	mV	Same as $V_{TR\_4P2\_F}$ + 25 mV
SID561	$V_{TR\_4P3\_R}$	LVD 4.3 V rising detection point for V_{DDD}	Typ – 4%	4325	Typ + 4%	mV	Same as $V_{TR\_4P3\_F}$ + 25 mV
SID562	$V_{TR\_4P4\_R}$	LVD 4.4 V rising detection point for V_{DDD}	Typ – 4%	4425	Typ + 4%	mV	Same as $V_{TR\_4P4\_F}$ + 25 mV
SID563	$V_{TR\_4P5\_R}$	LVD 4.5 V rising detection point for V_{DDD}	Typ – 4%	4525	Typ + 4%	mV	Same as $V_{TR\_4P5\_F}$ + 25 mV
SID564	$V_{TR\_4P6\_R}$	LVD 4.6 V rising detection point for V_{DDD}	Typ – 4%	4625	Typ + 4%	mV	Same as $V_{TR\_4P6\_F}$ + 25 mV
SID565	$V_{TR\_4P7\_R}$	LVD 4.7 V rising detection point for V_{DDD}	Typ – 4%	4725	Typ + 4%	mV	Same as $V_{TR\_4P7\_F}$ + 25 mV

(table continues...)

Table 41 (continued) System resources

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/ conditions
SID566	$V_{TR\_4P8\_R}$	LVD 4.8 V rising detection point for V_{DDD}	Typ – 4%	4825	Typ + 4%	mV	Same as $V_{TR\_4P8\_F}$ + 25 mV
SID567	$V_{TR\_4P9\_R}$	LVD 4.9 V rising detection point for V_{DDD}	Typ – 4%	4925	Typ + 4%	mV	Same as $V_{TR\_4P9\_F}$ + 25 mV
SID568	$V_{TR\_5P0\_R}$	LVD 5.0 V rising detection point for V_{DDD}	Typ – 4%	5025	Typ + 4%	mV	Same as $V_{TR\_5P0\_F}$ + 25 mV
SID569	$V_{TR\_5P1\_R}$	LVD 5.1 V rising detection point for V_{DDD}	Typ – 4%	5125	Typ + 4%	mV	Same as $V_{TR\_5P1\_F}$ + 25 mV
SID570	$V_{TR\_5P2\_R}$	LVD 5.2 V rising detection point for V_{DDD}	Typ – 4%	5225	Typ + 4%	mV	Same as $V_{TR\_5P2\_F}$ + 25 mV
SID571	$V_{TR\_5P3\_R}$	LVD 5.3 V rising detection point for V_{DDD}	Typ – 4%	5325	Typ + 4%	mV	Same as $V_{TR\_5P3\_F}$ + 25 mV
SID573	LVD_RR_A	Power ramp rate: V_{DDD} (Active)	–	–	100	mV/µs	
SID574	LVD_RR_DS	Power ramp rate: V_{DDD} (DeepSleep)	–	–	10	mV/µs	
SID575	$t_{DLY\_ACT\_LVD}$	Active mode delay between V_{DDD} falling/rising through LVD rising/falling point and an internal LVD signal transitioning	–	–	1	µs	Guaranteed by design
SID575A	$t_{DLY\_DS\_LVD}$	DeepSleep mode delay between V_{DDD} falling/rising through LVD rising/falling point and an internal LVD signal rising	–	–	4	µs	Guaranteed by design
SID576	t_{RES_LVD}	Response time of LVD, V_{DDD} supply. LVD guaranteed to generate pulse for V_{DDD} pulse width greater than this. (For falling-then-rising supply at maximum ramp rate; pulse width is time below LVD falling point)	100	–	–	ns	Guaranteed by design
High-voltage OVD (HV OVD) specifications							
SID580	$V_{TR\_5P0\_R}$	HV OVD 5.0-V rising detection point for V_{DDD} and V_{DDA}	5.049	5.205	5.361	V	

(table continues...)

Table 41 (continued) System resources

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/ conditions
SID581	$V_{TR\_5P0\_F}$	HV OVD 5.0-V falling detection point for V_{DDD} and V_{DDA}	5.025	5.18	5.335	V	
SID582	$V_{TR\_5P5\_R}$	HV OVD 5.5-V rising detection point for V_{DDD} and V_{DDA} (default)	5.548	5.72	5.892	V	
SID583	$V_{TR\_5P5\_F}$	HV OVD 5.5-V falling detection point for V_{DDD} and V_{DDA} (default)	5.524	5.695	5.866	V	
SID585	HVOVD_RR_A	Power ramp rate: V_{DDD} and V_{DDA} (Active)	–	–	100	mV/µs	
SID586	HVOVD_RR_DS	Power ramp rate: V_{DDD} and V_{DDA} (DeepSleep)	–	–	10	mV/µs	
SID587	$t_{DLY\_ACT\_HVOVD}$	Active mode delay between V_{DDD} falling/rising through $V_{TR\_5P0\_F/R}$ or $V_{TR\_5P5\_F/R}$ and an internal HV OVD signal transitioning	–	–	1	µs	Guaranteed by design
SID587A	$t_{DLY\_ACT\_HVOVD\_A}$	Active mode delay between V_{DDA} falling/rising through $V_{TR\_5P0\_F/R}$ or $V_{TR\_5P5\_F/R}$ and an internal HV OVD signal transitioning	–	–	1.5	µs	Guaranteed by design
SID587B	$t_{DLY\_DS\_HVOVD}$	DeepSleep mode delay between V_{DDD}/V_{DDA} falling/ rising through $V_{TR\_5P0\_F/R}$ or $V_{TR\_5P5\_F/R}$ and an internal HV OVD signal transitioning	–	–	4	µs	Guaranteed by design
SID588	t_{RES_HVOVD}	Response time of HV OVD. (For rising-then-falling supply at maximum ramp rate; threshold is $V_{TR\_5P0\_R}$ or $V_{TR\_5P5\_R}$.)	100	–	–	ns	Guaranteed by design
Low-voltage OVD (LV OVD) specifications							
SID590	$V_{TR\_R\_LVOVD}$	LV OVD rising detection point for V_{CCD}	1.261	1.3	1.339	V	
SID591	$V_{TR\_F\_LVOVD}$	LV OVD falling detection point for V_{CCD}	1.237	1.275	1.313	V	
SID595	$t_{DLY\_ACT\_LVOVD}$	Active mode delay between V_{CCD} falling/rising through V_{TR_F/R_LVOVD} and an internal LV OVD signal transitioning	–	–	1	µs	Guaranteed by design

(table continues...)

Table 41 (continued) System resources

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/ conditions
SID595A	$t_{DLY\_DS\_LVOVD}$	DeepSleep mode delay between V_{CCD} falling/rising through V_{TR_F/R_LVOVD} and an internal LV OVD signal transitioning	–	–	12	µs	Guaranteed by design
SID596	t_{RES_LVOVD}	Response time of LV OVD. (For rising-then-falling supply at maximum ramp rate; threshold is $V_{TR\_R\_LVOVD}$.)	100	–	–	ns	Guaranteed by design
Over current detection (OCD) specifications							
SID598	I_{OCD}	OCD detection range for V_{CCD}	156	–	315	mA	Guaranteed by design
SID599	I_{OCD_DPSLP}	Over current detection range in DeepSleep mode	18	–	72	mA	Guaranteed by design

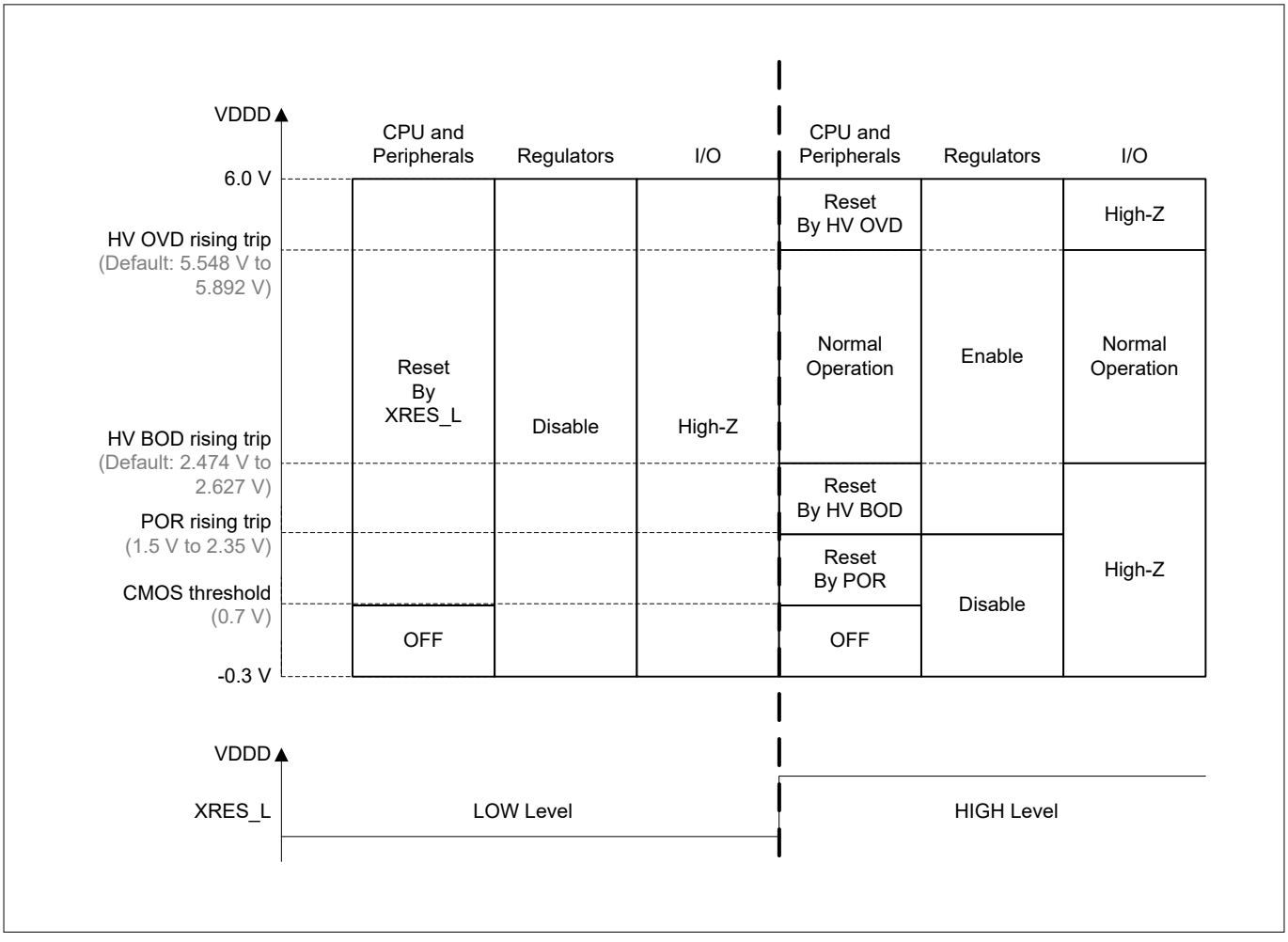


Figure 24 Device operations supply range

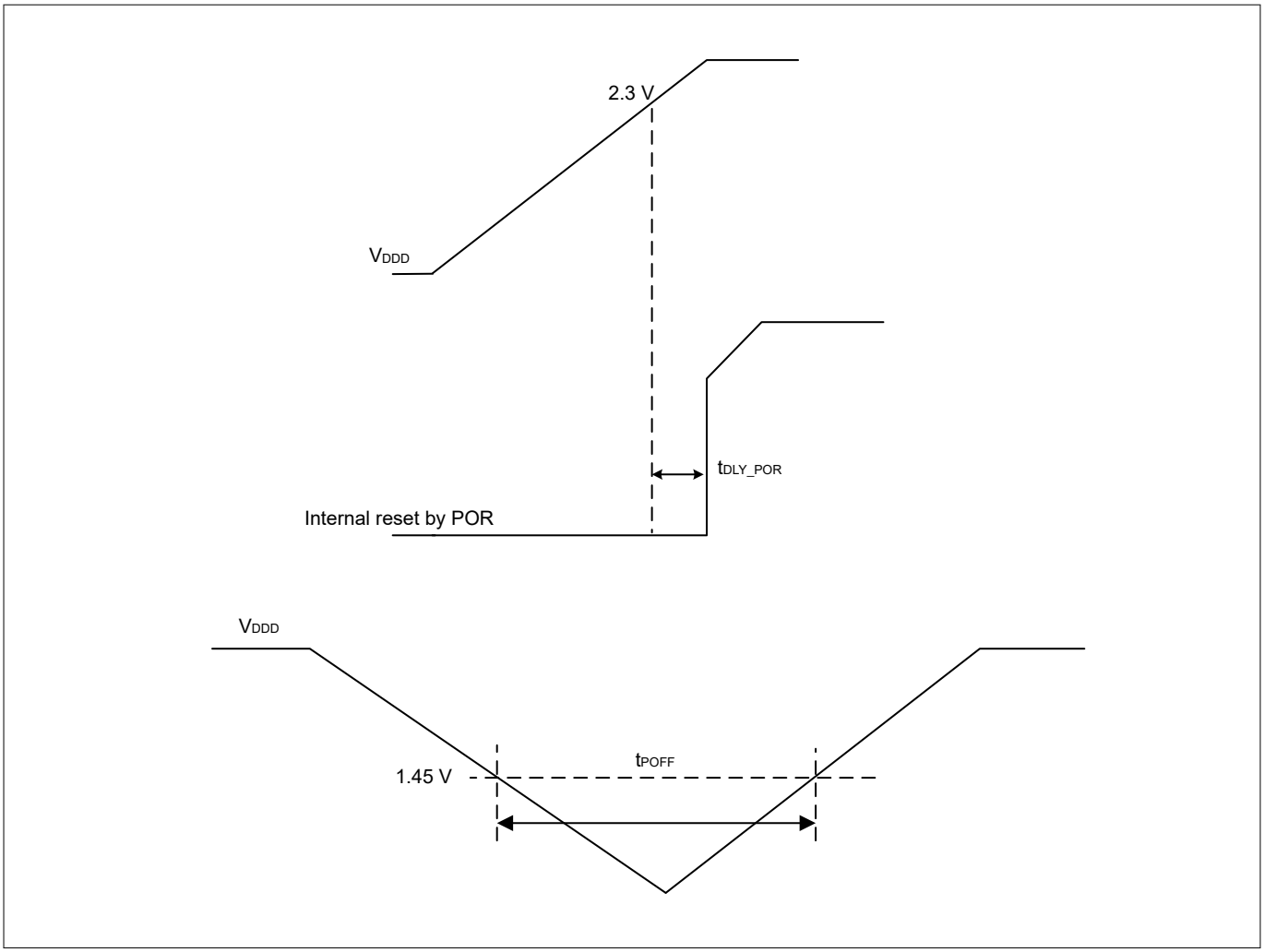


Figure 25 POR specifications

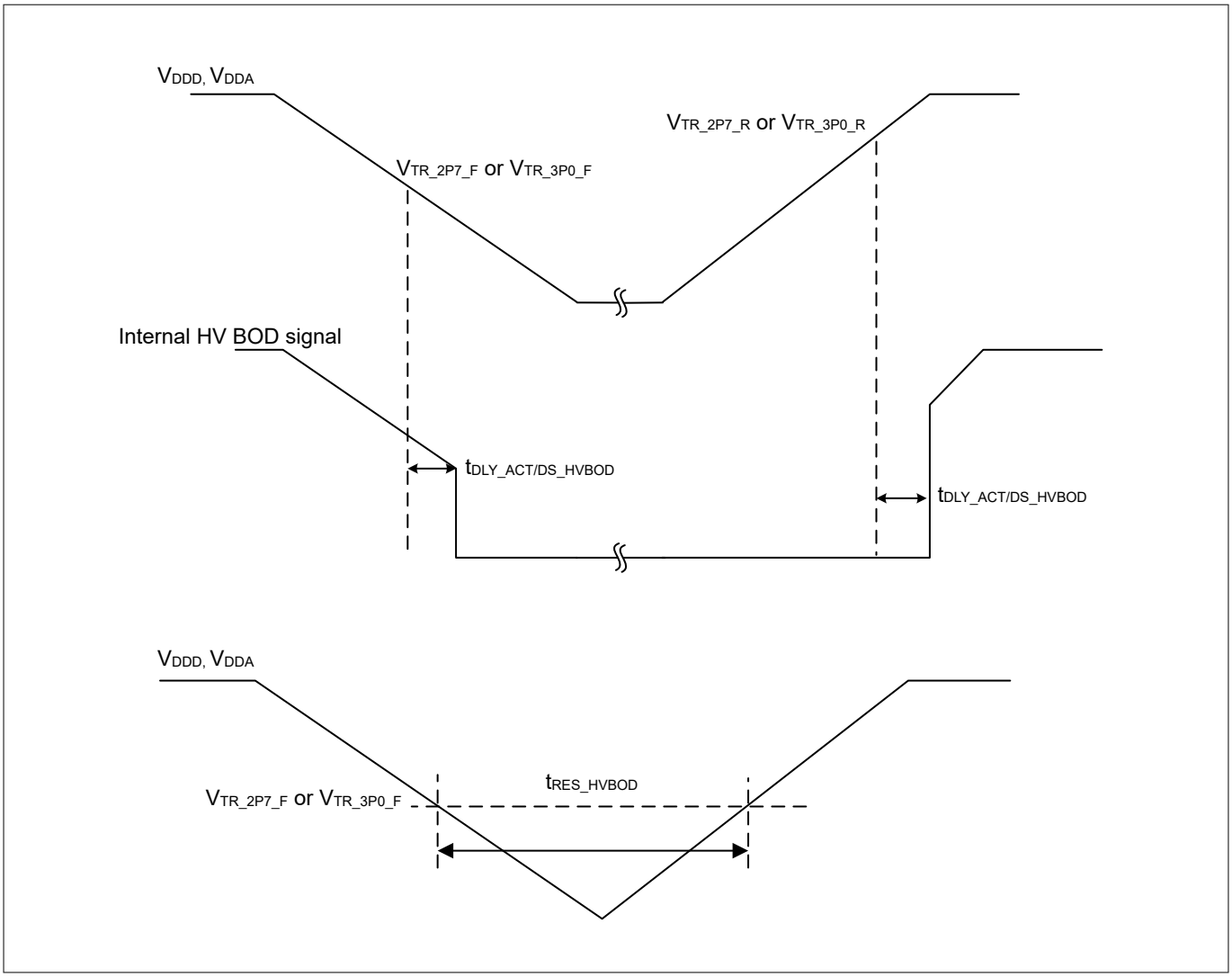


Figure 26 High-voltage BOD specifications

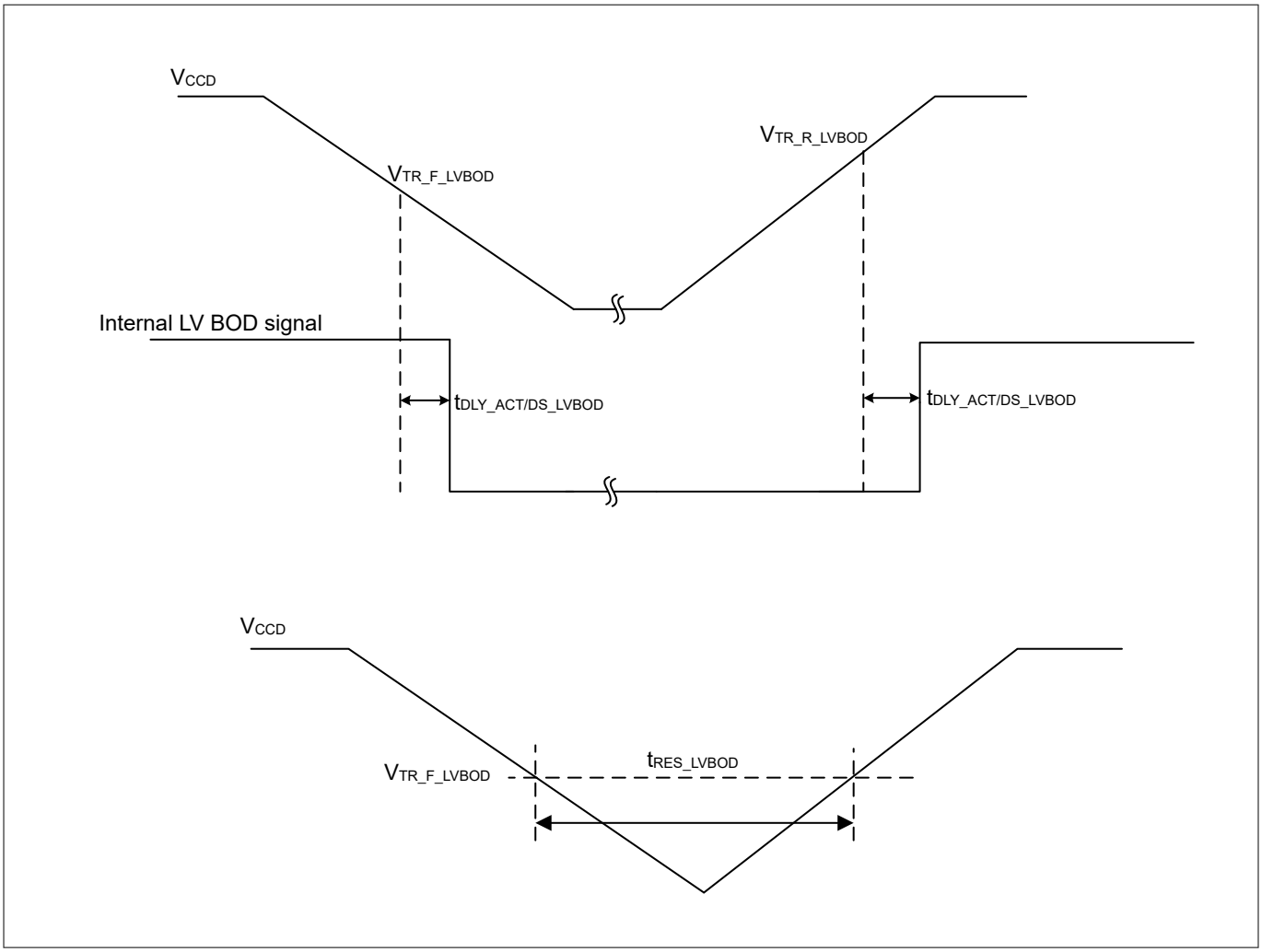


Figure 27 Low-voltage BOD specifications

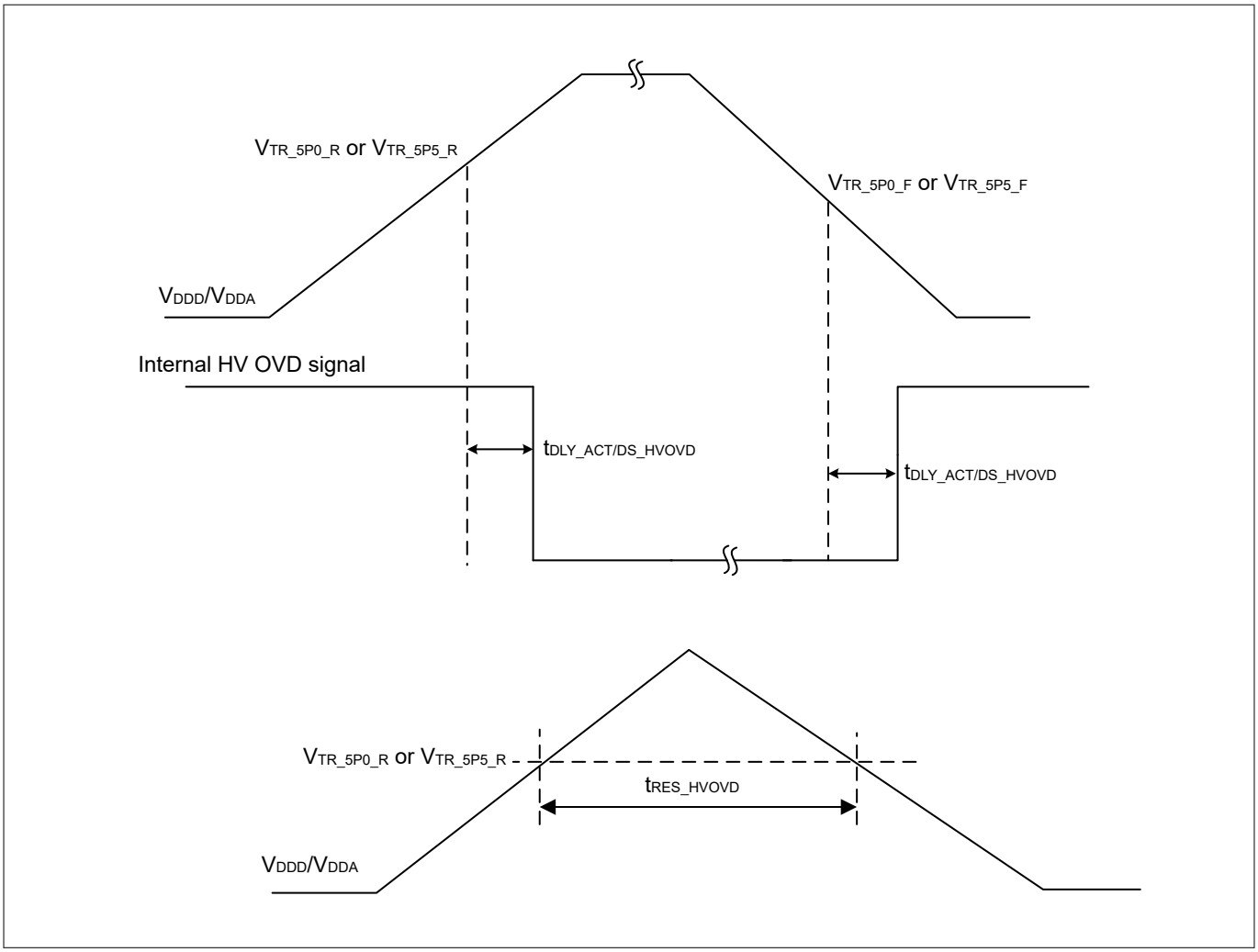


Figure 28 High-voltage OVD specifications

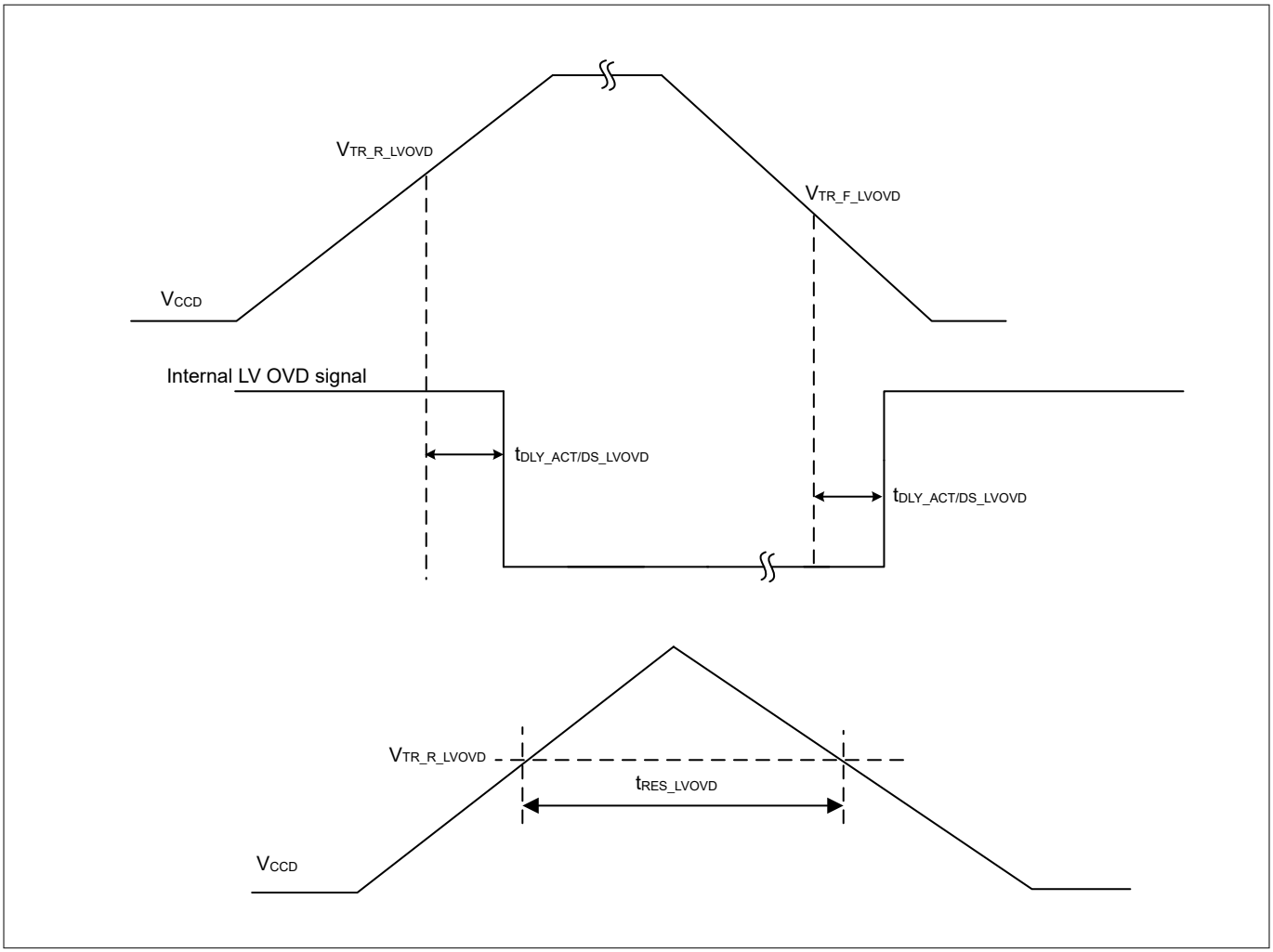


Figure 29 Low-voltage OVD specifications

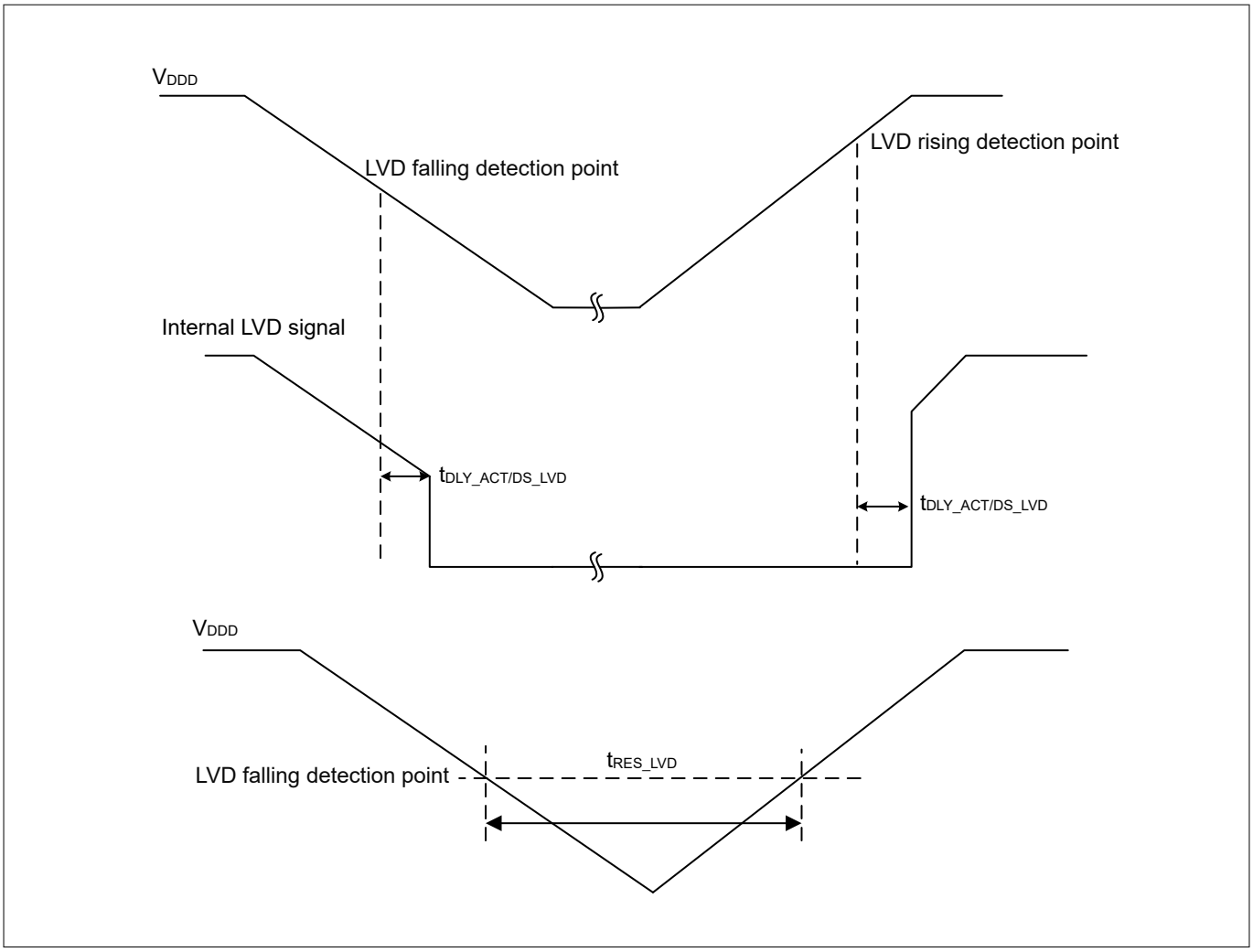


Figure 30 LVD specifications

26.11 Debug

All specifications are valid for –40°C ≤ T_A ≤ 125°C and for 2.7 V to 5.5 V except where noted.

26.11.1 SWD

Table 42 SWD interface specifications [Conditions: drive_sel<1:0> = 00]

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID300	f_{SWDCLK}	SWD clock input frequency	–	–	10	MHz	2.7 V ≤ V_{DDD} ≤ 5.5 V
SID301	t_{SWDI_SETUP}	SWDI setup time	0.25 × T	–	–	ns	T = 1/f_{SWDCLK}
SID302	t_{SWDI_HOLD}	SWDI hold time	0.25 × T	–	–	ns	T = 1/f_{SWDCLK}
SID303	t_{SWDO_VALID}	SWDO valid time	–	–	0.5 × T	ns	T = 1/f_{SWDCLK}
SID304	t_{SWDO_HOLD}	SWDO hold time	1	–	–	ns	T = 1/f_{SWDCLK}

26.11.2 JTAG

Table 43 JTAG AC specifications [Conditions: drive_sel<1:0> = 00]

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID620	t_{JCKH}	TCK HIGH time	30	–	–	ns	30-pF load
SID621	t_{JCKL}	TCK LOW time	30	–	–	ns	30-pF load
SID622	t_{JCP}	TCK clock period	66.7	–	–	ns	30-pF load
SID623	t_{JSU}	TDI/TMS setup time	12	–	–	ns	30-pF load
SID624	t_{JH}	TDI/TMS hold time	12	–	–	ns	30-pF load
SID625	t_{JZX}	TDO High-Z to active	–	–	30	ns	30-pF load
SID626	t_{JXZ}	TDO active to High-Z	–	–	30	ns	30-pF load
SID627	t_{JCO}	TDO clock to output	–	–	30	ns	30-pF load

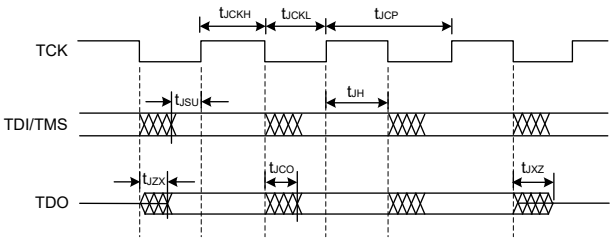


Figure 31 JTAG timing diagram

26.11.3 Trace

Table 44 Trace specifications [Conditions: drive_sel<1:0> = 00]

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID1412A	C_{TRACE}	Trace capacitive load	–	–	30	pF	
SID1412	t_{TRACE_CYC}	Trace clock period	40	–	–	ns	Trace clock cycle time for 25 MHz
SID1413	t_{TRACE_CLKL}	Trace clock LOW pulse width	2	–	–	ns	Clock low pulse width
SID1414	t_{TRACE_CLKH}	Trace clock HIGH pulse width	2	–	–	ns	Clock high pulse width
SID1415A	t_{TRACE_SETUP}	Trace data setup time	3	–	–	ns	Trace data setup time
SID1416A	t_{TRACE_HOLD}	Trace data hold time	2	–	–	ns	Trace data hold time

26.12 Clock specifications

All specifications are valid for –40°C ≤ T_A ≤ 125°C and for 2.7 V to 5.5 V except where noted.

The basic requirement on the clock-frequency dependency of the cores is that the Cortex®-M0+ core should run at an integer divider from the Cortex®-M4F core clock.

Example combinations are listed in Table 45.

Table 45 Clock requirements

Core Cortex®-M4F clock (MHz)	Core Cortex®-CM0+ clock (MHz)
160	80
120	60
100	100
80	80

Table 46 Root and intermediate clocks

Clock	Max frequency (MHz)	Description
CLK_HF0	160	Root clock for CPUSS, PERI
CLK_HF1	100	Event generator (CLK_REF), Clock output on EXT_CLK pins (when used as output)
CLK_HF2	2	CSV
CLK_FAST	160	Generated by dividing CLK_HF0, intermediate clock for CM4
CLK_SLOW	100	Generated by clock gating CLK_PERI, intermediate clock for CM0+, Crypto, P-DMA, M-DMA
CLK_PERI	100	Generated by clock gating CLK_HF0, intermediate clock for SCB, PASS, CAN, TCPWM, IOSS, CPU trace

***Note**: Intermediate clocks that are not listed have the same limitations as that of their parent clock.*

Table 47 IMO AC specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID310	f_{IMOTOL}	IMO operating frequency	7.92	8	8.08	MHz	
SID311	$t_{STARTIMO}$	IMO startup time	–	–	7.5	µs	Startup time to 90% of final frequency
SID312	I_{IMO_ACT}	IMO current	–	13.5	22	µA	Guaranteed by design

Table 48 ILO AC specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID320	$f_{ILOTRIM}$	ILO operating frequency	31.1296	32.768	34.4064	kHz	
SID321	$t_{STARTILO}$	ILO startup time	–	8	12	µs	Startup time to 90% of final frequency
SID323	I_{ILO}	ILO current	–	500	2800	nA	Guaranteed by design

Table 49 ECO specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID330	f_{ECO}	Crystal frequency range	3.988	–	33.34	MHz	
SID332	R_{FDBK}	Feedback resistor value. Min: RTRIM = 3; Max: RTRIM = 0 with 100 kΩ step size on RTRIM	100	–	400	kΩ	Guaranteed by design
SID333	I_{ECO3}	ECO current at T_J = 150 °C	–	–	2000	µA	Maximum operation current with a 33-MHz crystal, maximum 18-pF load
SID334	t_{START_4M}	4-MHz ECO startup time[46]	–	–	10	ms	Time from set CLK_ECO_CONFIG.ECO_EN to 1 until CLK_ECO_STATUS.ECO_READY is set to 1 (See Clock Timing Diagrams)
SID335	t_{START_33M}	33-MHz ECO startup time[46]	–	–	1	ms	Time from set CLK_ECO_CONFIG.ECO_EN to 1 until CLK_ECO_STATUS.ECO_READY is set to 1 (See Clock Timing Diagrams)

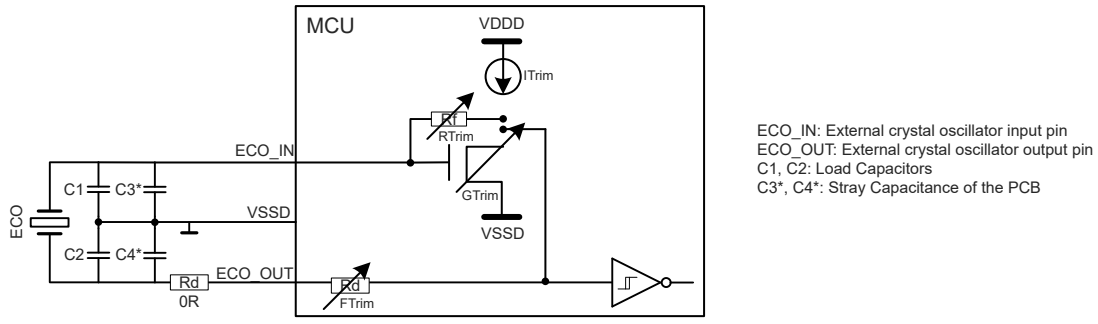


Figure 32 ECO connection scheme

***Note**: See the family-specific Architecture TRM for more information on crystal requirements (XMC5000 MCU architecture technical reference manual).*

[46] Mainly depends on the external crystal.

Table 50 PLL specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID340	PLL_LOCK	Time to achieve PLL lock	–	–	35	µs	
SID341	f_{PLL_OUT}	Output frequency from PLL block	11	–	160	MHz	
SID342	PLL_LJIT1	Long term jitter	–0.25	–	0.25	ns	For 125 ns f_{PLL_VCO}: 320 MHz f_{PLL_OUT}: 40 MHz to 160 MHz f_{PLL_PFD}: 8 MHz f_{PLL_IN}: ECO
SID343	PLL_LJIT2	Long term jitter	–0.5	–	0.5	ns	For 500 ns f_{PLL_VCO}: 320 MHz f_{PLL_OUT}: 40 MHz to 160 MHz f_{PLL_PFD}: 8 MHz f_{PLL_IN}: ECO
SID344	PLL_LJIT3	Long term jitter	–0.5	–	0.5	ns	For 1000 ns f_{PLL_VCO}: 320 MHz f_{PLL_OUT}: 40 MHz to 160 MHz f_{PLL_PFD}: 8 MHz f_{PLL_IN}: ECO
SID345A	PLL_LJIT5	Long term jitter	–0.75	–	0.75	ns	For 10000 ns f_{PLL_VCO}: 320 MHz f_{PLL_OUT}: 40 MHz to 160 MHz f_{PLL_PFD}: 8 MHz f_{PLL_IN}: ECO
SID346	f_{PLL_IN}	PLL input frequency	3.988	–	33.34	MHz	
SID347	I_{PLL_160M1}	PLL operating current (f_{OUT} = 160 MHz)	–	740	1110	µA	f_{IN} = 4 MHz, f_{PFD} = 4 MHz, f_{VCO} = 320 MHz, f_{OUT} = 160 MHz
SID347A	I_{PLL_160M2}	PLL operating current (f_{OUT} = 160 MHz)	–	750	1125	µA	f_{IN} = 8 MHz, f_{PFD} = 8 MHz, f_{VCO} = 320 MHz, f_{OUT} = 160 MHz
SID347B	I_{PLL_160M3}	PLL operating current (f_{OUT} = 160 MHz)	–	750	1125	µA	f_{IN} = 16 MHz, f_{PFD} = 8 MHz, f_{VCO} = 320 MHz, f_{OUT} = 160 MHz

(table continues...)

Table 50 (continued) PLL specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID339	I_{PLL_100M1}	PLL operating current (f_{OUT} = 100 MHz)	–	520	780	µA	f_{IN} = 4 MHz, f_{PFD} = 4 MHz, f_{VCO} = 200 MHz, f_{OUT} = 100 MHz
SID339A	I_{PLL_100M2}	PLL operating current (f_{OUT} = 100 MHz)	–	530	795	µA	f_{IN} = 8 MHz, f_{PFD} = 8 MHz, f_{VCO} = 200 MHz, f_{OUT} = 100 MHz
SID339B	I_{PLL_100M3}	PLL operating current (f_{OUT} = 100 MHz)	–	530	795	µA	f_{IN} = 16 MHz, f_{PFD} = 8 MHz, f_{VCO} = 200 MHz, f_{OUT} = 100 MHz
SID348	I_{PLL_80M1}	PLL operating current (f_{OUT} = 80 MHz)	–	520	780	µA	f_{IN} = 4 MHz, f_{PFD} = 4 MHz, f_{VCO} = 240 MHz, f_{OUT} = 80 MHz
SID348A	I_{PLL_80M2}	PLL operating current (f_{OUT} = 80 MHz)	–	530	795	µA	f_{IN} = 8 MHz, f_{PFD} = 8 MHz, f_{VCO} = 240 MHz, f_{OUT} = 80 MHz
SID348B	I_{PLL_80M3}	PLL operating current (f_{OUT} = 80 MHz)	–	530	795	µA	f_{IN} = 16 MHz, f_{PFD} = 8 MHz, f_{VCO} = 240 MHz, f_{OUT} = 80 MHz
SID348C	f_{PLL_VCO}	VCO frequency	170	–	400	MHz	
SID349C	f_{PLL_PFD}	PFD frequency	3.988	–	8	MHz	

Table 51 FLL specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID350	t_{FLL_WAKE}	FLL wake up time	–	–	5	µs	Wakeup with < 10°C temperature change while in DeepSleep. f_{FLL_IN} = 8 MHz, f_{FLL_OUT} = 100 MHz, Time from stable reference clock until FLL frequency is within 5% of final value

(table continues...)

Table 51 (continued) FLL specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID351	f_{FLL_OUT}	Output frequency from FLL block	24	–	100	MHz	Output range of FLL divided-by-2 output
SID352	FLL_CJIT	FLL frequency accuracy	–1	–	1	%	This is added to the error of the source
SID353	f_{FLL_IN}	Input frequency	0.25	–	80	MHz	
SID354	I_{FLL}	FLL operating current	–	250	360	µA	Reference clock: IMO, CCO frequency: 200 MHz, FLL frequency: 100 MHz, guaranteed by design

Table 52 WCO specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID360	f_{WCO}	Watch Crystal frequency	–	32.768	–	kHz	Maximum drive level: 0.5 µW
SID361	WCO_DC	WCO duty cycle	10	–	90	%	
SID362E	t_{START_WCOE}	WCO start-up time[47]	–	–	1400	ms	Time from set CTL.WCO_EN to 1 until STATUS.WCO_OK is set to 1. (See Clock Timing Diagrams)
SID363	I_{WCO}	WCO current	–	1.4	–	µA	

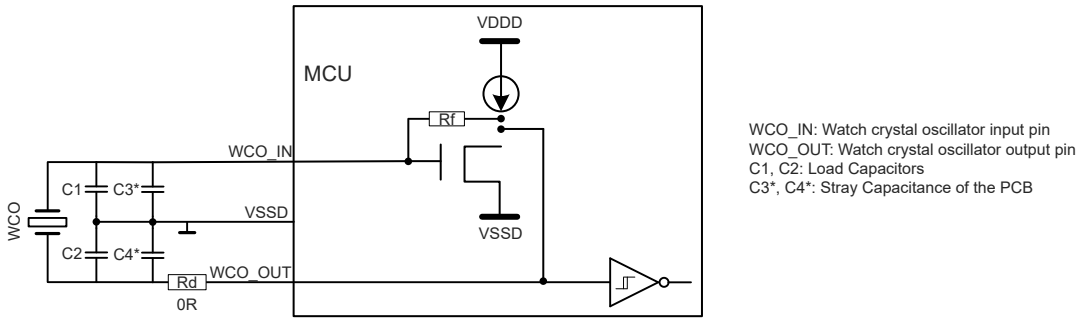


Figure 33 WCO connection scheme

***Note**: See the family specific Architecture TRM for more information on crystal requirements (XMC5000 MCU architecture technical reference manual).*

47 Mainly depends on the external crystal.

Table 53 External clock input specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID366	f_{EXT}	External clock input frequency	0.25	–	80	MHz	For EXT_CLK pin (all input level settings: CMOS, TTL)
SID367	EXT_DC	External clock duty cycle	45	–	55	%	

26.12.1 Clock timing diagrams

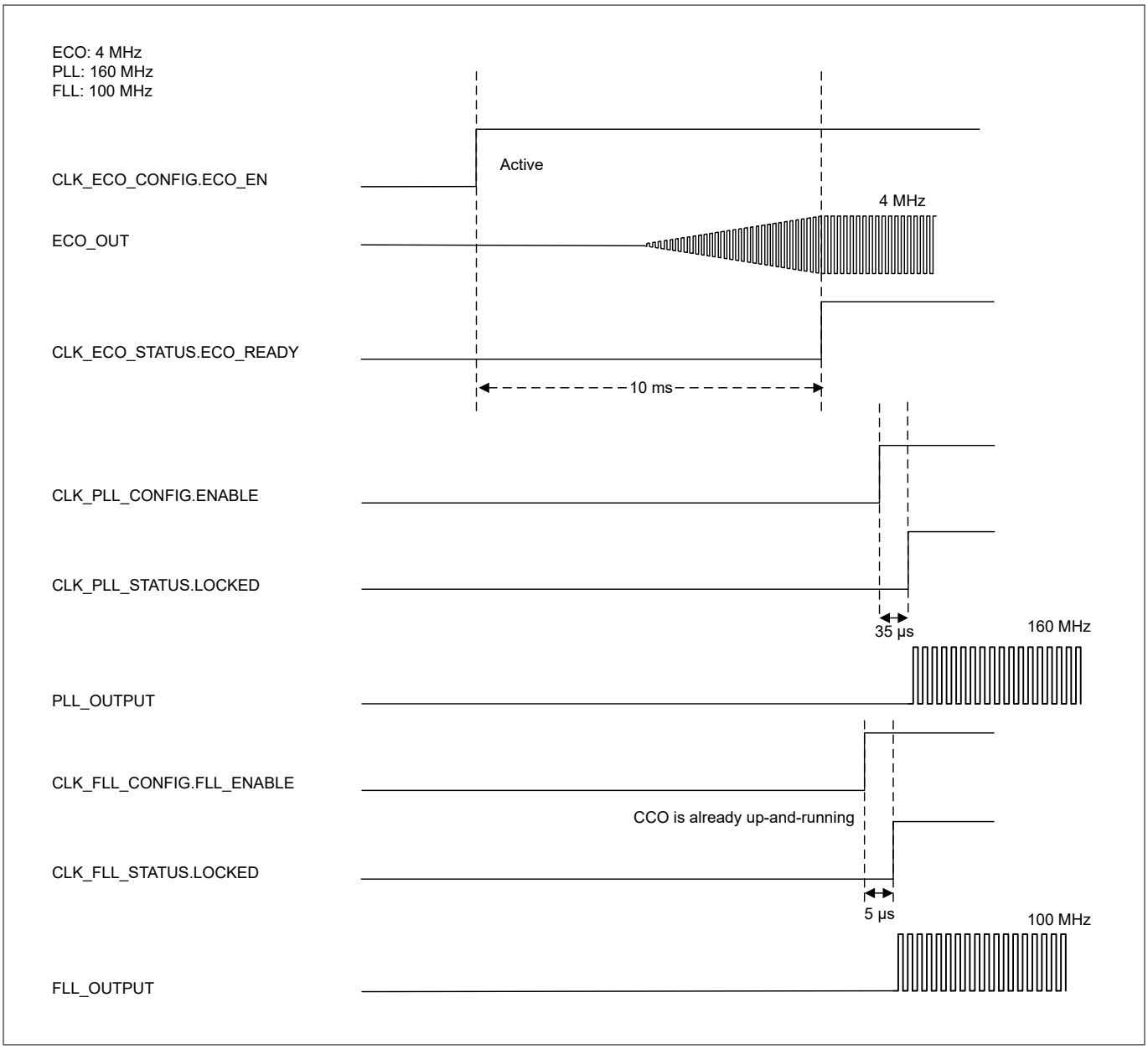


Figure 34 ECO to PLL or FLL diagram

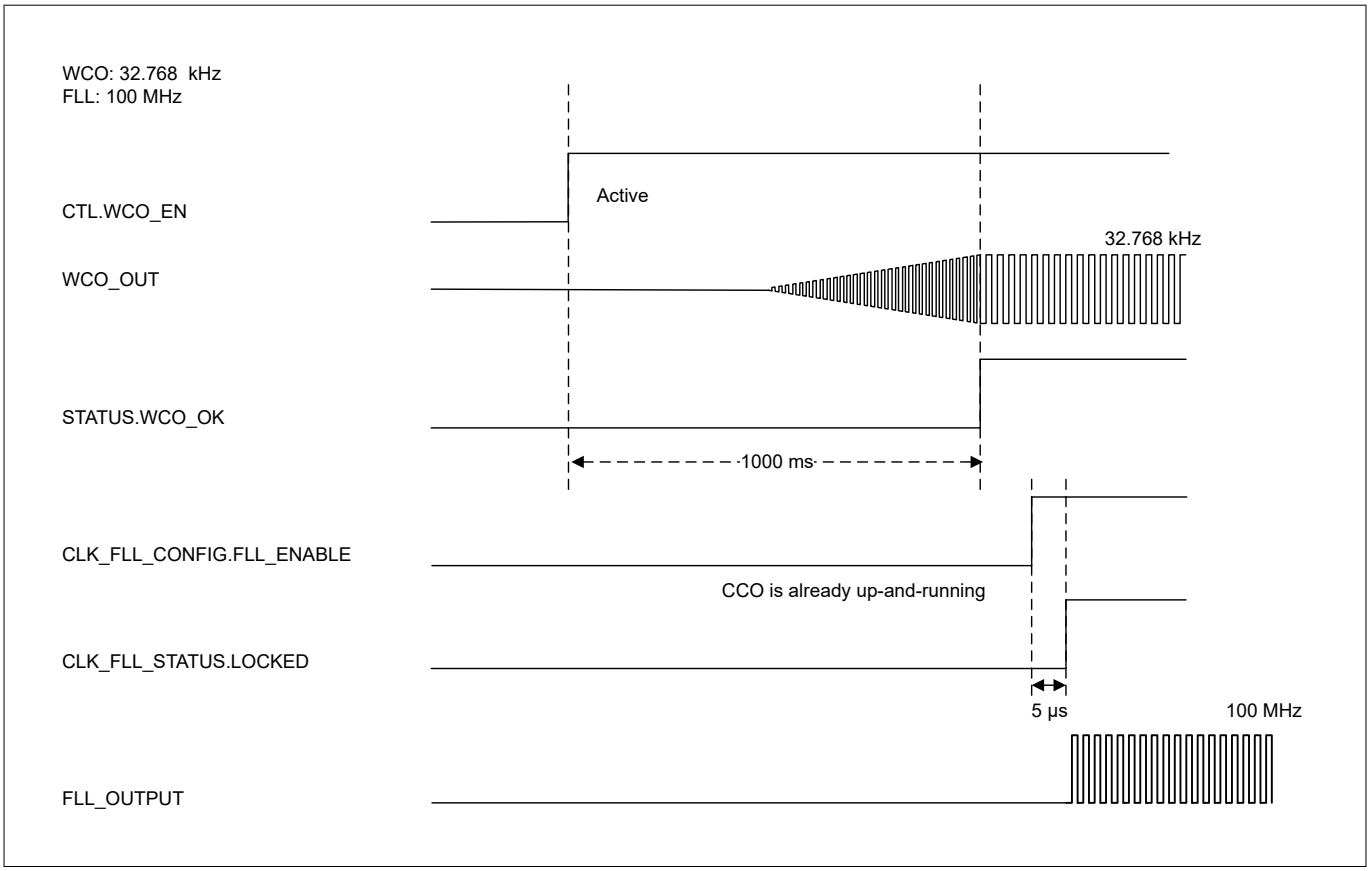


Figure 35 WCO to FLL diagram

Table 54 MCWDT timeout specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID410	t_{MCWDT1}	Minimum MCWDT timeout	58.12	–	–	µs	When using the ILO (32.768 kHz + 5%) and 16-bit MCWDT counter Guaranteed by design
SID411	t_{MCWDT2}	Maximum MCWDT timeout	–	–	2.11	s	When using the ILO (32.768 kHz – 5%) and 16-bit MCWDT counter Guaranteed by design

Table 55 WDT timeout specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID412	t_{WDT1}	Minimum WDT timeout	58.12	–	–	µs	When using the ILO (32.768 kHz + 5%) and 32-bit WDT counter Guaranteed by design

(table continues...)

Table 55 (continued) WDT timeout specifications

Spec ID	Parameter	Description	Min	Typ	Max	Units	Details/conditions
SID413	t_{WDT2}	Maximum WDT timeout	–	–	38.33	h	When using the ILO (32.768 kHz – 5%) and 32-bit WDT counter Guaranteed by design
SID414	t_{WDT3}	Default WDT timeout	–	1000	–	ms	When using the ILO and 32-bit WDT counter at 0x8000 (default value), guaranteed by design

27 Ordering information

The XMC5200 microcontroller part numbers and features are listed in Table 56. The Arm® TAP JTAG ID is 0x6BA0 0477.

Table 56 XMC5200 ordering information

Device code	Ordering code	Package	Code flash(KB)	Work flash(KB)	RAM (KB)	ADC channels	SCB channels	CANFD channels	Temperature grade	JTAG ID CODE
XMC5200-F64	XMC5200-F64K1088AA	64-LQFP	1088[48]	96[49]	128	27	7	5	–40°C to 125°C	0x2E3CB069[50]
XMC5200-F100	XMC5200-F100K1088AA	100-LQFP	1088	96	128	39	8	6	–40°C to 125°C	0x2E3DB069
XMC5200-F144	XMC5200-F144K1088AA	144-LQFP	1088	96	128	54	8	6	–40°C to 125°C	0x2E3E3069

27.1 Part number nomenclature

Table 57 Ordering code nomenclature

Description	Values	Meaning	Comment
XMC™ prefix	XMC™	XMC™ prefix- industrial microcontroller	Fixed
Series name	5100	XMC5000 device family	Arm® Cortex®-M4F based
	5200		
	5300		
Package option	F	LQFP	Available package options
PKG pin count	64	64-pin	PKG pin count options
	100	100-pin	
	144	144-pin	
Code flash/Work flash/RAM density	576	576 KB/64 KB/64 KB	–
	1088	1088 KB/96 KB/128 KB	
	2112	2112 KB/128 KB/256 KB	

48 Code-flash size 1088 KB = 32 KB × 30 (Large Sectors) + 8 KB × 16 (Small Sectors).
49 Work-flash size 96 KB = 2 KB × 36 (Large Sectors) + 128 B × 192 (Small Sectors).
50 JTAG ID CODE bits 12 through 27, represents the Silicon ID of the device.

28 Packaging

XMC5200 is offered in the packages listed in Table 58.

Table 58 Package information

Package	Dimensions	Contact/lead pitch	Coefficient of Thermal Expansion[51]	I/O Pins
144-LQFP	20 × 20 × 1.7 mm (max)	0.5 mm	a1[52] = 8.5 ppm/°C, a2[53] = 33.7 ppm/°C	122
100-LQFP	14 × 14 × 1.7 mm (max)	0.5 mm	a1[52] = 8.5 ppm/°C, a2[53] = 33.6 ppm/°C	78
64-LQFP	10 × 10 × 1.7 mm (max)	0.5 mm	a1[52] = 8.5 ppm/°C, a2[53] = 33.2 ppm/°C	49

Table 59 Package characteristics

Parameter	Description	Conditions	Min	Typ	Max	Units
T_A	Operating ambient temperature	–	–40	–	125	°C
T_J	Operating junction temperature	–	–	–	150	°C
$R_{\theta JA}$	Package thermal resistance, junction to ambient θ_{JA}[54] [55]	64 LQFP	–	–	37.6	°C/Watt
		100 LQFP	–	–	29.8	°C/Watt
		144 LQFP	–	–	26.2	°C/Watt
$R_{\theta JB}$	Package thermal resistance, junction to board θ_{JB}	64 LQFP	–	–	32.0	°C/Watt
		100 LQFP	–	–	21.3	°C/Watt
		144 LQFP	–	–	20.9	°C/Watt
$R_{\theta JC}$	Package thermal resistance, junction to case θ_{JC}	64 LQFP	–	–	7.8	°C/Watt
		100 LQFP	–	–	5.6	°C/Watt
		144 LQFP	–	–	4.2	°C/Watt

Table 60 Solder reflow peak temperature, package moisture sensitivity level (MSL), IPC/JEDEC J-STD-2

Package	Maximum peak temperature (°C)	Maximum time at peak temperature (seconds)	MSL
144-LQFP	260	30	3
100-LQFP	260	30	3
64-LQFP	260	30	3

51 The numbers are estimated values based simulation only and are based on a single bill of material combination per package type.
52 a1 = CTE (Coefficient of Thermal Expansion) value below T_g (ppm/°C) (T_g is glass transition temperature which is 131°C).
53 a2 = CTE value above Tg (ppm/°C).
54 Board condition complies to JESD51-7 (4 Layers).
55 Maximum value °C/Watt shown is for T_A = 125°C.

28 Packaging

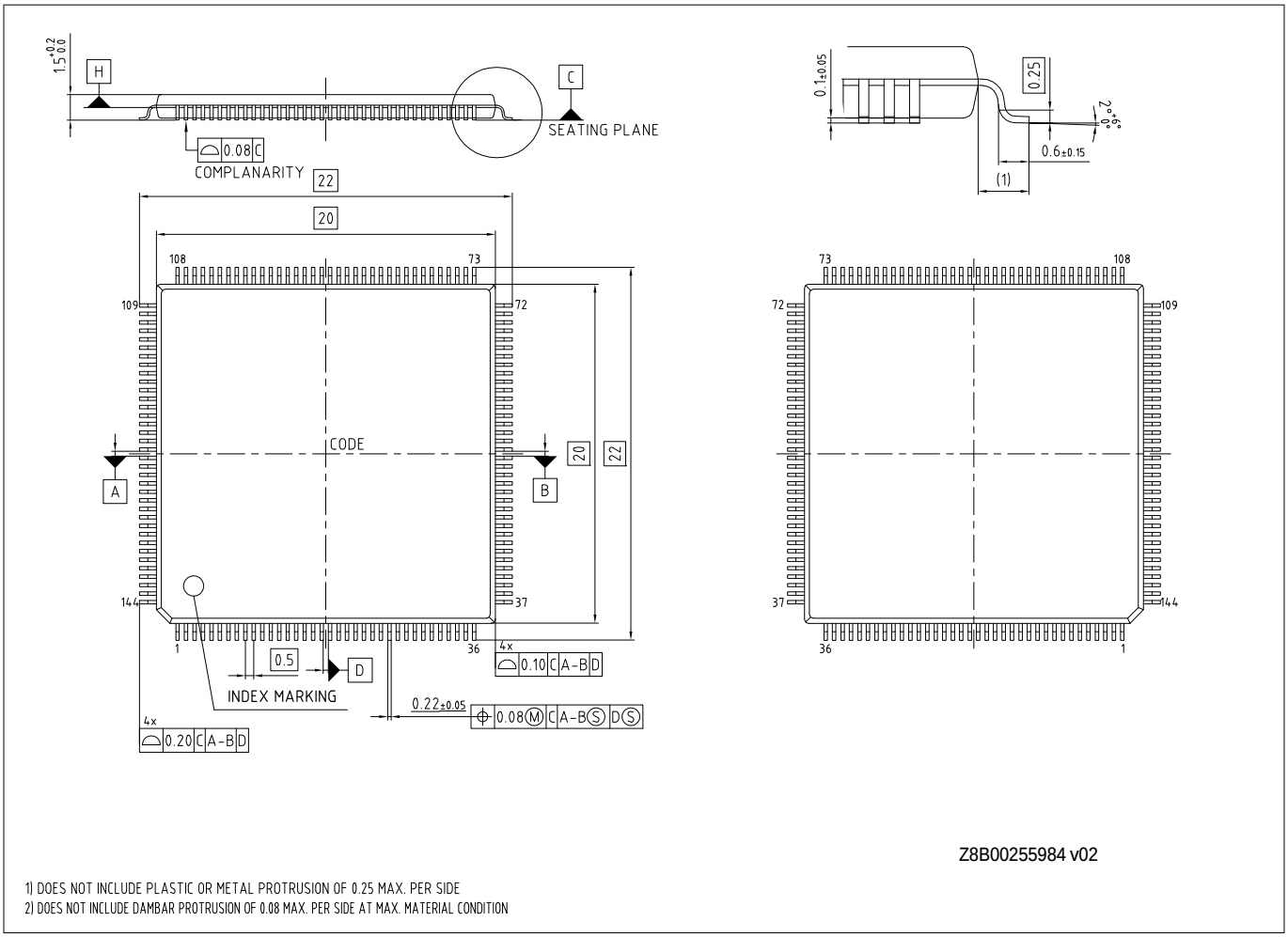


Figure 36 144-pin LQFP (20.0 × 20.0 × 1.7 mm) LQS144 (PG-LQFP-144)

28 Packaging

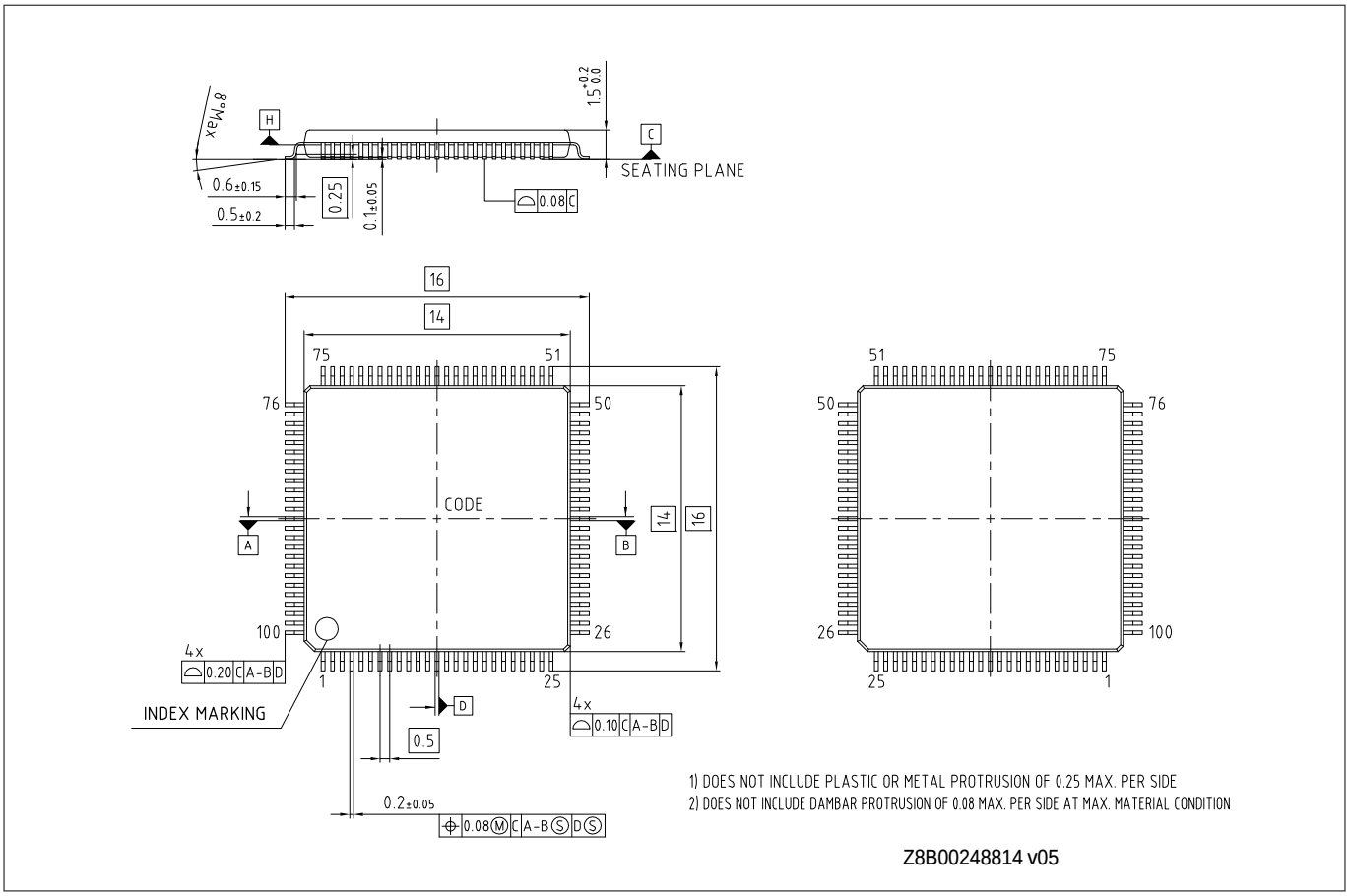


Figure 37 100-pin LQFP (14.0 × 14.0 × 1.7 mm) LQI100 (PG-LQFP-100)

28 Packaging

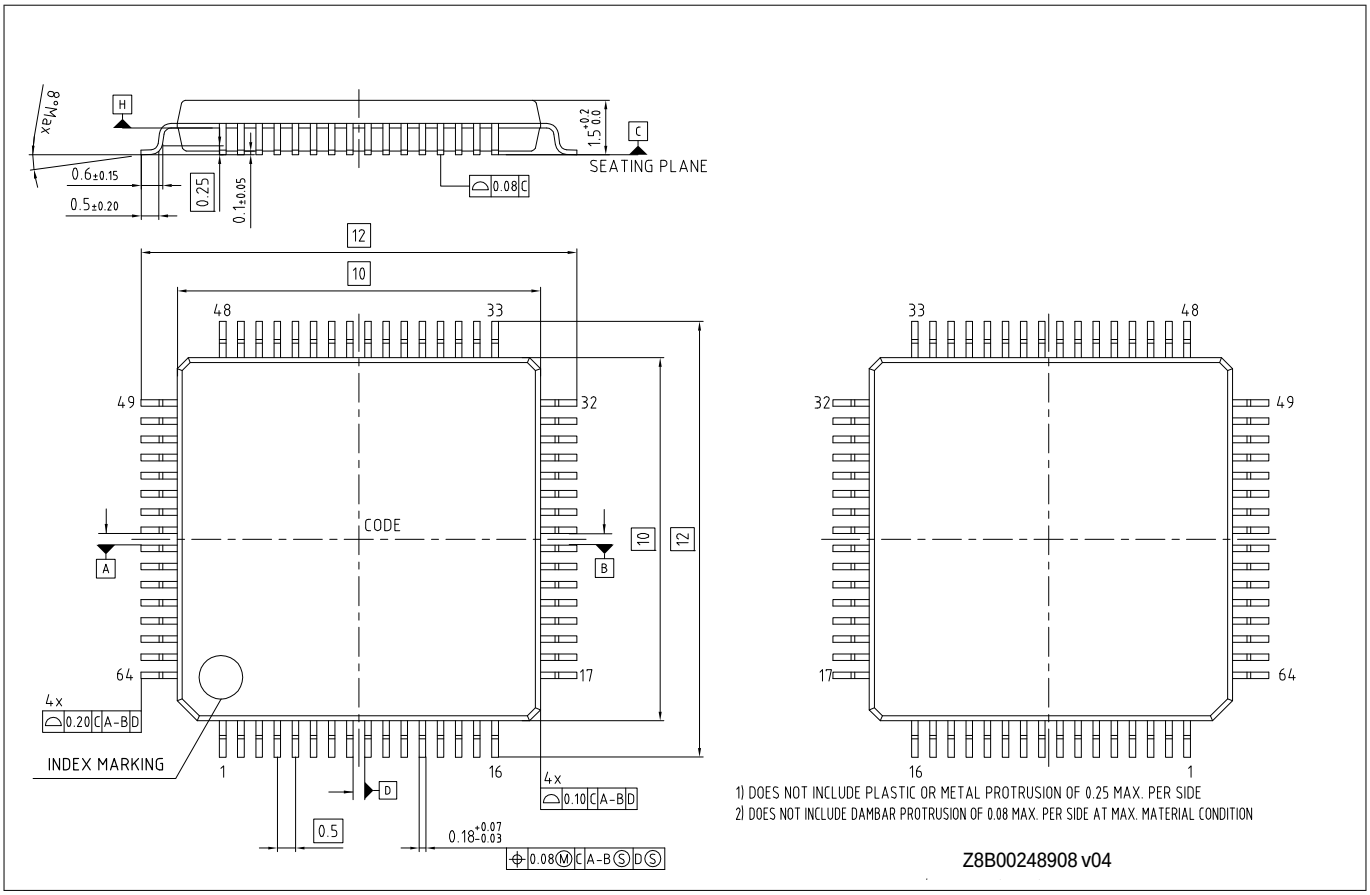


Figure 38 64-pin LQFP (10.0 × 10.0 × 1.7 mm) LQD064 (PG-LQFP-64)

29 Glossary

Table 61 Glossary

Acronym	Description
A/D	Analog-to-digital
ABS	Absolute
ADC	Analog-to-digital converter
AES	Advanced encryption standard
AHB	AMBA (advanced microcontroller bus architecture) high-performance bus, Arm® data transfer bus
Arm®	Advanced RISC machine, a CPU architecture
SIL	Safety integrity level
BOD	Brown-out detection
CAN FD	Controller Area Network with Flexible Data rate
CMOS	Complementary metal-oxide-semiconductor
CPU	Central Processing Unit
CRC	Cyclic redundancy check, an error-checking protocol
CSV	Clock supervisor
CTI	Cross trigger interface
DFT	Design-for-test
DW	Datawire same as P-DMA
ECC	Error correcting code/Elliptical curve cryptography
ECO	External crystal oscillator
ETM	Embedded Trace Macrocell
EVTGEN	Event generator
FLL	Frequency-locked loop
FPU	Floating point unit
GPIO	General purpose input/output
HSM	Hardware security module
I/O	Input/output
I2C	Inter-Integrated Circuit, a communications protocol
ILO	Internal low-speed oscillator
IMO	Internal main oscillator
IOSS	Input/output sub-system
IPC	Inter-processor communication
IrDA	Infrared interface

(table continues...)

29 Glossary

Table 61 (continued) Glossary

Acronym	Description
IRQ	Interrupt request
JTAG	Joint test action group
LDO	Low drop out regulators
LVD	Low voltage detection
OTA	Over-the-air programming
OTP	One-time programmable
OVD	Overvoltage detection
P-DMA	Peripheral-Direct Memory Access same as DW
PLL	Phase Locked Loop
POR	Power-on reset
PPU	Peripheral protection unit
PRNG	Pseudorandom number generator
PWM	Pulse-width modulation
MCU	Microcontroller Unit
MCWDT	Multi-counter watchdog timer
M-DMA	Memory-Direct Memory Access
MISO	SPI Master-in slave-out
MMIO	Memory mapped I/O
MOSI	SPI Master-out slave-in
MPU	Memory protection unit
MTB	Micro trace buffer
MUL	Multiplier
MUX	Multiplexer
NVIC	Nested vectored interrupt controller
RAM	Random access memory
RISC	Reduced-instruction-set computing
ROM	Read only memory
RTC	Real-time clock
SAR	Successive approximation register
SCB	Serial communication block
SCL	I^2C serial clock
SDA	I^2C serial data
TCPWM	Timer/Counter Pulse-width modulator

(table continues...)

Table 61 (continued) Glossary

Acronym	Description
TTL	Transistor-transistor logic
TRNG	True random number generator
UART	Universal Asynchronous Transmitter Receiver
WCO	Watch crystal oscillator
WDT	Watchdog timer reset
XRES_L	External reset I/O pin

29.1 Bootloading or end-of-line programming

- Triggered at device startup if a trigger condition is applied
- CAN communication is utilized
- The bootloader polls for communication on the CAN interface at specific time intervals until a total timeout of 300 seconds is reached
- If a bootloader command is received on any communication interface, the polling stops, and the bootloader switches to using that interface

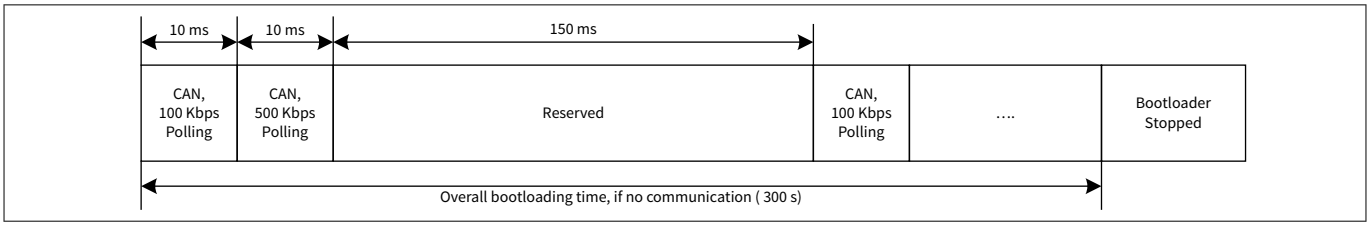


Figure 39 Bootloading sequence

Table 62 CAN interface details

Sl. No.	CAN interface	Configuration
1	CAN mode	Classic CAN
2	CAN instance	CAN0, Channel#1
3	CAN TX	P0.2/CAN0_1_TX
4	CAN RX	P0.3/CAN0_1_RX
5	CAN Transceiver NSTB/EN (Low)	P23.3 (optional)
6	CAN Transceiver EN/EN (High)	P2.1 (optional)
7	CAN RX message ID	0x1A1
8	CAN TX message ID	0x1B1
9	Baud	100 or 500 kbps alternating

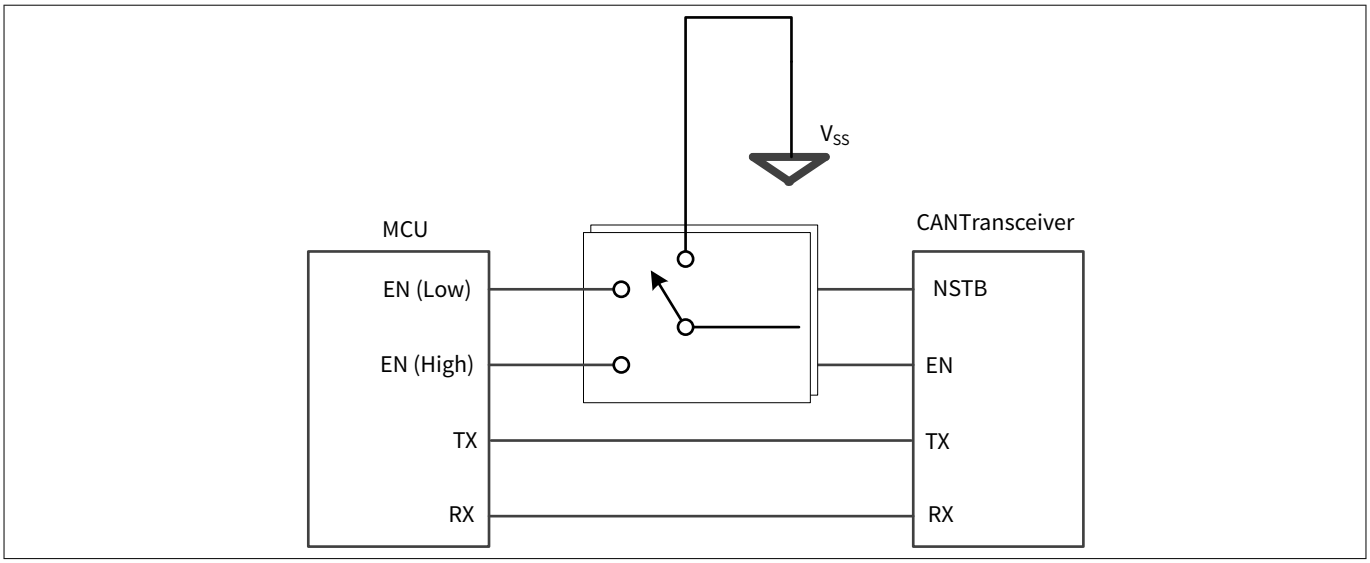


Figure 40 MCU to CAN transceiver connections

29.2 External IP revisions

Table 63 IP revisions

Module	IP	Revision	Vendor
CAN FD	mxttcanfd	M_TTCAN IP revision: Rev.3.2.3	Bosch
Arm® Cortex®-M0+	armcm0p	Cortex®-M0+-r0p1	Arm®
Arm® Cortex®-M4F	armcm4	Cortex®-M4-r0p1	Arm®
Arm® Coresight	armcoresighttk	CoreSight-SoC-TM100-r3p2	Arm®

30 Errata

This section outlines the errata for the XMC5200 product family. It includes details such as trigger conditions, scope of impact, available workarounds, and silicon revision applicability. For further questions, contact your local Infineon sales representative.

Part numbers affected

All XMC5200 parts.

XMC5200 qualification status

Production samples

XMC5200 errata summary

Table 64 defines the errata applicability to available XMC5200 family devices.

Table 64 Errata applicability

Items	Errata ID	XMC5200	Silicon rev.	Fix status
Crypto LSL1, LSR1, LSL1_WITH_CARRY, & LSR1_WITH_CARRY instructions may work incorrectly in certain scenarios	53	All parts	D	No silicon fix planned. Use workaround.
Crypto MEM_BUF may be corrupted	42	All parts	D	No silicon fix planned. Use workaround.
ConfigureFmInterrupt API assumes a parameter with 8-byte boundary but the actual boundary is 4 bytes	67	All parts	D	No silicon fix planned. Use workaround.
SMPU/MPU/PPU protection region size is limited to 2 GB	68	All parts	D	No silicon fix planned. Use workaround.
DirectExecute API may return error if called with arguments placed in SRAM memory	69	All parts	D	No silicon fix planned. Use workaround.
CAN FD RX FIFO top pointer feature does not function as expected	96	All parts	D	No silicon fix planned. Use workaround.
CAN FD debug message handling state machine is not reset to Idle state when CANFD_CH_CCCR.INIT is set	97	All parts	D	No silicon fix planned. Use workaround.
TPIU Peripheral ID mismatch	98	All parts	D	No silicon fix planned.
Limitation of the memory hole in SCB register space	124	All parts	D	No silicon fix planned.
WDT service can be missed	129	All parts	D	No silicon fix planned. Use workaround.

(table continues...)

Table 64 (continued) Errata applicability

Items	Errata ID	XMC5200	Silicon rev.	Fix status
CAN FD controller message order inversion when transmitting from dedicated Tx buffers configured with same Message ID	147	All parts	D	No silicon fix planned. Use workaround.
CAN FD incomplete description of dedicated Tx buffers and Tx queue related to transmission from multiple buffers configured with the same Message ID	167	All parts	D	No silicon fix planned. Use workaround. TRM was updated.
Misleading status is returned for Flash and eFuse system calls, if there are pending NC ECC faults in SRAM controller #0	175	All parts	D	No silicon fix planned. Use workaround. TRM was updated.
WDT reset causes loss of SRAM retention	176	All parts	D	No silicon fix planned. Use workaround. TRM was updated.
Crypto ECC errors may be set after boot with application authentication	185	All parts	D	No silicon fix planned. TRM was updated.
Incomplete erase of Code Flash cells could happen Erase Suspend/Erase Resume is used along with Erase Sector operation in Non-Blocking mode	198	All parts	D	Fixed to update the Flash settings from date code 304xxxxx.
Limitation for keeping the port state from peripheral IP after wakeup from DeepSleep	199	All parts	D	No silicon fix planned. TRM was updated.
Limitation of clock configuration before entering DeepSleep mode.	202	All parts	D	No silicon fix planned. TRM was updated.
SCBx_INTR_TX.UNDERFLOW bit may be set unintentionally.	204	All parts	D	No silicon fix planned. TRM was updated.
Hardfault may occur when calling the SROM APIs listed below while executing EraseSector or ProgramRow in non-blocking mode	206	All parts	D	No silicon fix planned. TRM will be updated.
CAN FD sporadic data corruption (payload) in case acceptance filtering does not finish before reception of data R3 (DB7..DB4) is complete	209	All parts	D	No silicon fix planned. Use workaround.

Crypto LSL1, LSR1, LSL1_WITH_CARRY, & LSR1_WITH_CARRY instructions may work incorrectly in certain scenarios	
Problem definition	The instructions `LSL1`, `LSR1`, `LSL1_WITH_CARRY`, and `LSR1_WITH_CARRY` are designed to ignore the value in `IW[3:0]`, as the "shift by 1" operation does not utilize these fields. However, due to a hardware issue, the shift operation fails when the register data field pointed to by `IW[3:0]` is `0`. In this case, the destination data remains the same as the source data.
Parameters affected	NA
Trigger condition(s)	Using LSL1, LSR1, LSL1_WITH_CARRY, and LSR1_WITH_CARRY instructions
Scope of impact	The shift does not happen (destination data is same as source data).

Workaround	`IW[3:0]` should point to a dummy register where the data field of the register contains a non-zero value (e.g., `rsrc0->data[12:0]`). Since the stack pointer (`r15`) inherently points to a non-zero value (as using the `LSL1` instruction requires that at least one register has been allocated, ensuring the stack pointer is not zero), it is safe to use `r15` as `rsrc0`.
	`static __forceinline void LSL1 (int rdst, int rsrc1)`
	`{`
	`AHB_WRITE_W (MMIO_CRYPTO_INSTR _FF_WR, (CRYPTO_VU_LSL_OPC << 24) | (rdst << 12) | (rsrc1 << 4) | 15);`
	`}`
	This workaround is also applicable to other instructions, including `LSR1`, `LSL1_WITH_CARRY`, and `LSR1_WITH_CARRY`.

Fix status	No silicon fix planned. Use workaround.

Crypto MEM_BUF may be corrupted	
Problem definition	The SRAM in the Crypto block has a size of 8 KB. However, the address decoding is configured to create four 8-KB images of the SRAM within a 32-KB address space. As a result, any writes to the memory space beyond the initial 8-KB image will corrupt the contents of the SRAM.
Parameters affected	NA
Trigger condition(s)	Any write to addresses in the range `0x40108000` to `0x4010FFFF` may result in unintended behavior.
Scope of impact	The `CRYPTO MEM_BUF` may become corrupted.
Workaround	The software must ensure that no access occurs beyond the 8-KB `MEM_BUF` address range, either through MMIO writes or address overflows, during the execution of Crypto operations.
Fix status	No silicon fix planned. Use workaround.

ConfigureFmInterrupt API assumes a parameter with 8-byte boundary but the actual boundary is 4 bytes	
Problem definition	The `STATUS_ADDR_PROTECTED` error will be returned if the `ConfigureFmInterrupt` API is called with arguments stored in SRAM that are aligned to a 4-byte boundary (either within available SRAM or protected boundary SRAM).

Parameters affected	NA
Trigger condition(s)	Call the `ConfigureFmInterrupt` API with arguments stored in SRAM that are aligned to a 4-byte boundary, either within the available SRAM or the protected boundary of SRAM.
Scope of impact	The `ConfigureFmInterrupt` API will fail and return the `STATUS_ADDR_PROTECTED` error status if it is called with an argument aligned to a 4-byte boundary within the available SRAM or the protected boundary of SRAM.
Workaround	Allow a 4-byte margin (assuming the API parameter size is 8 bytes) when storing the arguments for the `ConfigureFmInterrupt` API.
Fix status	No silicon fix planned. Use workaround.

SMPU/MPU/PPU protection region size is limited to 2 GB	
Problem definition	If the SMPU/MPU/PPU protection block size is configured to 4 GB (`PROT_SMPU_SMPU_STRUCT_ATT0.REGION.SIZE = 31`), an overflow will occur in the internal `uint32` variable used during the protection check in SROM. Since 4 GB equals `0x1_0000_0000`, which exceeds the range of a 32-bit unsigned integer, the overflow causes SROM to interpret the protection size as zero. Consequently, no protection will be applied.
Parameters affected	NA
Trigger condition(s)	To configure the SMPU/MPU/PPU for a protection region size of 4 GB (or a region size with the value `31u`.
Scope of impact	If SMPU/MPU/PPU is configured to protect region size of 4 GB, then SROM software does not apply any protection per the request.
Workaround	Use two protection blocks, each with a region size of 2 GB, to achieve the equivalent of 4-GB region size protection.
Fix status	No silicon fix planned. Use workaround.

DirectExecute API may return error if called with arguments placed in SRAM memory	
Problem definition	If DirectExecute API is called in the master PC (other than PC0 or PC1) with arguments in SRAM_SCRATCH_ADDR, then the API will return STATUS_ADDR_PROTECTED status.
Parameters affected	NA
Trigger condition(s)	Call DirectExecute API with arguments in SRAM_SCRATCH_ADDR and master PC configured > 1.
Scope of impact	DirectExecute API, if called with master PC configured > 1 and arguments in SRAM_SCRATCH_ADDR, will return STATUS_ADDR_PROTECTED.
Workaround	Call DirectExecute API with master PC0 or PC1 if the arguments are stored in SRAM memory.
Fix status	No silicon fix planned. Use workaround.

CAN FD RX FIFO top pointer feature does not function as expected

30 Errata

Problem definition	RX FIFO top pointer function calculates the address for received messages in Message RAM by hardware. This address should restart from the start address after reading all messages of RX FIFO n size (n: 0 or 1). However, the address does not restart from the start address when RX FIFO n size is set to 1(CANFD_CH_RXFnC.FnS = 0x01). This results in CPU/DMA reading messages from the wrong address in Message RAM.
Parameters affected	NA
Trigger condition(s)	The RX FIFO top pointer function is used when RX FIFO n size is set to 1 element (CANFD_CH_RXFnC.FnS = 0x01).
Scope of impact	Received message cannot be correctly read by using the RX FIFO top pointer function, when RX FIFO n size is set to 1 element.
Workaround	Any of the following can be used as a workaround: 1) Set RX FIFO n size to 2 or more when using RX FIFO top pointer function. 2) Do not use the RX FIFO top pointer function when RX FIFO n size is set to 1 element. Instead of RX FIFO top pointer, read received messages from the Message RAM directly.
Fix status	No silicon fix planned. Use workaround.

CAN FD debug message handling state machine is not reset to Idle state when CANFD_CH_CCCR.INIT is set

Problem definition	If either of the CANFD_CH_CCCR.INIT bits is set by the Host or when the M_TTCAN module enters BusOff state, the debug message handling state machine stays in its current state instead of being reset to Idle state. Configuring the bit CANFD_CH_CCCR.CCE does not change CANFD_CH_RXF1S.DMS.
Parameters affected	NA
Trigger condition(s)	Either of the CANFD_CH_CCCR.INIT bits is set by the Host or when the M_TTCAN module enters BusOff state.
Scope of impact	The errata is limited to the use case when the debug on CAN functionality is active. Normal operation of the CAN module is not affected, in which case the debug message handling state machine always remains in Idle state. In the described use case, the debug message handling state machine is stopped and remains in the current state signaled by the CANFD_CH_RXF1S.DMS bit. In case CANFD_CH_RXF1S.DMS is set to 0b11, the DMA request remains active.
Workaround	In case the debug message handling state machine has stopped while CANFD_CH_RXF1S.DMS is 0b01 or 0b10, it can be reset to Idle state by hardware reset or by reception of debug messages after CANFD_CH_CCCR.INIT is reset to zero.
Fix status	No silicon fix planned. Use workaround.

TPIU Peripheral ID mismatch

Problem definition	TPIU peripheral ID indicates that it is M3-TPIU instead of M4-TPIU.
Parameters affected	NA
Trigger condition(s)	When debugger reads PID registers for component identification.
Scope of impact	The only impact is that the debugger reads the TPIU as M3-TPIU.
Workaround	No specific workaround required. Debugger can use trace features.

Fix status	No silicon fix planned.

Limitation of the memory hole in SCB register space

Problem definition	The memory hole [offset address: 0x1000 to 0xFFFF] inside the SCB register space is not aligned to the below defined spec. Since the offset address bits [15:12] are ignored and treated as 4'b0000, write/read access to the offset address [0x1000 to 0xFFFF] will actually happen to [0x0000 to 0x0FFF]. - Access to address gaps in mapped memory space: writes are ignored and any read returns a zero.
Parameters affected	NA
Trigger condition(s)	Access to the memory hole [offset address: 0x1000 to 0xFFFF] in the SCB register space
Scope of impact	The memory hole [offset address: 0x1000 to 0xFFFF] in the SCB register space is not aligned to other IP registers.
Workaround	Do not access to the memory hole [offset address: 0x1000 to 0xFFFF] in SCB register space.
Fix status	No silicon fix planned.

WDT service can be missed

Problem definition	If WDT service happens within 4 ILO clock cycles before DeepSleep entry, it clears the counter but does not fully complete an internal handshake. A service after DeepSleep wakeup may then be missed if it occurs less than 2 ILO clock cycles after the processor resumes clocking. After this time, the internal handshake is complete and servicing works normally.
Parameters affected	NA
Trigger condition(s)	Service WDT within four ILO clock cycles before DeepSleep entry and within two ILO clock cycles of processor clock resuming
Scope of impact	WDT service after DeepSleep wakeup may be ignored and WDT continues counting. This can cause unintended WARN_ACTION or UPPER_ACTION, including interrupt, fault, and/or reset.
Workaround	Wait 130 µs or more after DeepSleep wakeup. (For example, to measure 130 µs, software can read the WDT_CNT register at wake up and make sure that WDT_CNT was incremented of 4 units before servicing WDT). Afterwards, write '1' to WDT service (WDT_SERVICE.SERVICE) after waiting until WDT service (WDT_SERVICE.SERVICE) reads '0'.
Fix status	No silicon fix planned. Use workaround.

CAN FD controller message order inversion when transmitting from dedicated Tx buffers configured with same Message ID

Problem definition	Configuration: Several Tx buffers are configured with same Message ID. Transmission of these Tx buffers is requested sequentially with a delay between the individual Tx requests. Expected behavior: When multiple Tx buffers that are configured with the same Message ID have pending Tx requests, they shall be transmitted in ascending order of their Tx buffer numbers. The Tx Buffer with lowest buffer number and pending Tx request is transmitted first. Observed behavior: It may happen, depending on the delay between the individual Tx requests, that in the case where multiple Tx buffers are configured with the same Message ID the Tx buffers are not transmitted in order of the Tx buffer number (lowest number first).
Parameters affected	NA
Trigger condition(s)	When multiple Tx buffers that are configured with the same Message ID have pending Tx requests.
Scope of impact	In the case described it may happen, that Tx buffers configured with the same Message ID and pending Tx request are not transmitted with lowest Tx buffer number first (message order inversion).
Workaround	Any of the following: 1) First write the group of Tx message with the same Message ID to the Message RAM and then afterwards request transmission of all these messages concurrently by a single write access to CANFDx_CHy_TXBAR. Before requesting a group of Tx messages with this Message ID ensure that no message with this Message ID has a pending Tx request. 2) Use the Tx FIFO instead of dedicated Tx buffers for the transmission of several messages with the same Message ID in a specific order. Applications not able to use workaround #1 or #2 can implement a counter within the data section of their messages sent with same ID in order to allow the recipients to determine the correct sending sequence.
Fix status	No silicon fix planned. Use workaround.

CAN FD incomplete description of dedicated Tx buffers and Tx queue related to transmission from multiple buffers configured with the same Message ID

Problem definition	The following are the updated description in Sections "Dedicated Tx Buffers" and "Tx Queue" of the Architecture TRM related to the transmission from multiple buffers configured with the same Message ID. Dedicated Tx buffers - TRM Statement: If multiple Tx buffers are configured with the same Message ID, the Tx buffer with the lowest buffer number is transmitted first. - Enhancement: These Tx buffers shall be requested in ascending order with lowest buffer number first. Alternatively all Tx buffers configured with the same Message ID can be requested simultaneously by a single write access to CANFDx_CHy_TXBAR. Tx Queue - TRM Statement: If multiple queue buffers are configured with the same Message ID, the queue buffer with the lowest buffer number is transmitted first. - Replacement: In case that multiple Tx Queue buffers are configured with the same Message ID, the transmission order depends on numbers of the buffers where the messages were stored for transmission. As these buffer numbers depend on the then current states of the PUT Index, a prediction of the transmission order is not possible. - TRM Statement: An Add Request cyclically increments the Put Index to the next free Tx buffer. - Replacement: The PUT Index always points to the free buffer of the Tx Queue with the lowest number.
Parameters affected	NA
Trigger condition(s)	Using multiple dedicated Tx buffers or Tx Queue buffers configured with the same Message ID.
Scope of impact	In the case the dedicated Tx buffers with the same Message ID are not requested in ascending order or at the same time or in case of multiple Tx Queue buffers with the same Message ID, it cannot be guaranteed, that these messages are transmitted in ascending order with lowest buffer number first.
Workaround	In case a defined order of transmission is required the Tx FIFO shall be used for transmission of messages with the same Message ID. Alternatively dedicated Tx buffers with the same Message ID shall be requested in ascending order with lowest buffer number first or by a single write access to CANFDx_CHy_TXBAR. Alternatively a single Tx buffer can be used to transmit those messages one after the other.
Fix status	No silicon fix planned. Use workaround. TRM was updated.

Misleading status is returned for Flash and eFuse system calls, if there are pending NC ECC faults in SRAM controller #0	
Problem definition	Flash and eFuse system calls will return misleading status of 0xF0000005 (“Page is write protected”) even for non-protected row, or 0xF0000002 (“Invalid eFuse address”) for valid eFuse address in case of pending NC ECC faults in SRAM controller #0.
Parameters affected	Return status of Flash and eFuse system calls.
Trigger condition(s)	NC ECC fault(s) pending in SRAM controller #0 and SWPUs are populated in the design.
Scope of impact	Flash and eFuse system calls will not work until the NC ECC fault(s) pending in SRAM controller #0 is/are properly handled.

Workaround	If the NC ECC fault(s) are not due to hardware malfunction (i.e. if the faults are due to usage of non-initialized SRAM or improper SRAM initialization), then clearing of these pending faults will resolve the issue.
Fix status	No silicon fix planned. TRM was updated.

WDT reset causes loss of SRAM retention	
Problem Definition	Architecture TRM table on “Reset Cause Distribution” shows that the WDT reset can retain SRAM if there is an orderly shutdown of the SRAM only during a warning interrupt. However, this is wrong. WDT reset causes loss of SRAM retention.
Parameters Affected	NA
Trigger Condition(s)	WDT reset
Scope of Impact	WDT reset causes loss of SRAM retention.
Workaround	None
Fix Status	No silicon fix planned. TRM was updated.

Crypto ECC errors may be set after boot with application authentication	
Problem Definition	Due to the improper initialization of the Crypto memory buffer, Crypto ECC errors may be set after boot with application authentication.
Parameters Affected	N/A
Trigger Condition(s)	Boot device with application authentication.
Scope of Impact	Crypto ECC errors may be set after boot with application authentication.
Workaround	Clear or ignore Crypto ECC errors which generated during boot with application authentication.
Fix Status	No silicon fix planned. TRM was updated.

Incomplete erase of Code Flash cells could happen Erase Suspend/Erase Resume is used along with Erase Sector operation in Non-Blocking mode	
Problem Definition	Code Flash memory can be erased in “Non-Blocking” mode; a Non-Blocking mode supported option allows users to suspend an ongoing erase sector operation. When an ongoing erase operation is interrupted using “Erase Suspend” and “Erase Resume”, Flash cells may not have been erased completely, even after the erase operation complete is indicated by FLASHC_STATUS register. Only Code Flash is impacted by this issue, Work Flash and Supervisory Flash (SFlash) are not impacted.
Parameters Affected	N/A
Trigger Condition(s)	Using EraseSector System Call in Non-Blocking mode for CM0+ to erase Code Flash and the ongoing erase operation is interrupted using EraseSuspend and EraseResume System calls.
Scope of Impact	When Code Flash sectors are erased in Non-Blocking mode and the ongoing erase operation is interrupted by Erase Suspend/Erase Resume, it cannot be guaranteed that the Code Flash cells are fully erased. Any read on the Code Flash area after the erase is complete or read on the programmed data after ProgramRow is complete can trigger ECC errors.

Workaround	Use any of the following: 1) User can use Non-Blocking mode for EraseSector, but must not interrupt the erase operation using Erase Suspend/Erase Resume. 2) If a Code Flash sector erase operation is interrupted using Erase Suspend/Erase Resume, then erase the same sector again without Erase Suspend/Erase Resume before reading the sector or programming the sector.
Fix Status	Fixed to update the Flash settings from date code 304xxxxx.

Limitation for keeping the port state from peripheral IP after wakeup from DeepSleep	
Problem Definition	The port state is not retained when the port selects peripheral IP (except CAN FD) and MCU wakes up from DeepSleep.
Parameters Affected	N/A
Trigger Condition(s)	The port selects peripherals (except CAN FD) and MCU wakes up from DeepSleep.
Scope of Impact	Unexpected port output change might affect user system.
Workaround	If the port selects peripherals (except for CAN FD), and the port output value needs to be maintained after wakeup from DeepSleep, set HSIOM_PRTx_PORT_SEL.IOy_SEL = 0 (GPIO) before DeepSleep and set the required output value in GPIO configuration registers. After wakeup, change HSIOM_PRTx_PORT_SEL.IOy_SEL back to the peripheral module as needed.
Fix Status	No silicon fix planned. TRM was updated.

Limitation of clock configuration before entering DeepSleep mode.	
Problem Definition	DeepSleep should not be entered while any FLL/PLL is enabled and uses ECO as its reference clock. Since the unstable ECO clock after wakeup is outside the allowed reference clock limits for FLL/PLL, it is possible for DeepSleep wakeup to fail.
Parameters Affected	N/A
Trigger Condition(s)	DeepSleep transition while any FLL/PLL is enabled and uses ECO as its reference clock.
Scope of Impact	Possibility of DeepSleep wakeup failure.
Workaround	If any FLL/PLL operates with the ECO as its reference clock, change the clock to either ECO direct or IMO direct or IMO with FLL/PLL before entering DeepSleep.
Fix Status	No silicon fix planned. TRM was updated.

SCBx_INTR_TX.UNDERFLOW bit may be set unintentionally.	
Problem Definition	There is possibility of setting the SCBx_INTR_TX.UNDERFLOW bit even if the FIFO is not empty.
Parameters Affected	N/A
Trigger Condition(s)	Using the TX FIFO for SCB when the AHB-Lite interface clock (CLK_GR6) frequency of the AHB bus is greater than 3x the SCB functionality clock (PCLK_SCBx_CLOCK).
Scope of Impact	SCBx_INTR_TX.UNDERFLOW bit may be set unintentionally.
Workaround	Ignore the SCBx_INTR_TX.UNDERFLOW bit if the FIFO is not empty.
Fix Status	No silicon fix planned. TRM was updated.

Hardfault may occur when calling the SROM APIs listed below while executing EraseSector or ProgramRow in non-blocking mode	
Problem Definition	The following SROM APIs read data in SFlash from bank#0 (or bank#1 if dual bank mode with mapping B is used). While doing that, they check if active non-blocking erase or program of bank#0 (or bank#1 if dual bank mode with mapping B is used) is not performed. Therefore, reading bank#0 (or bank#1 if dual bank mode with mapping B is used) while there is an active erase/program operation triggers a bus error. This results in a hardfault based on the FLASHC_FLASH_CTL register settings. Affected SROM APIs: • ReadSWPU • WriteSWPU • GenerateHash • Checksum* • ComputeBasicHash* • CheckFactoryHash • ProgramWorkFlash** *: Do not call it to calculate on the bank where programming/erasing is in progress. **: Do not use it during non-blocking operation.
Parameters Affected	N/A
Trigger Condition(s)	Calling the affected SROM APIs while executing EraseSector or ProgramRow in non-blocking mode on bank#0 (or bank#1 if dual bank mode with mapping B is used).
Scope of Impact	The affected SROM APIs cannot be used while executing EraseSector or ProgramRow in non-blocking mode on bank#0 (or bank#1 if dual bank mode with mapping B is used).
Workaround	Do not use the affected SROM APIs while executing EraseSector or ProgramRow in non-blocking mode on bank#0 (or bank#1 if dual bank mode with mapping B is used).
Fix Status	No silicon fix planned. TRM will be updated.

CAN FD sporadic data corruption (payload) in case acceptance filtering does not finish before reception of data R3 (DB7..DB4) is complete

Problem Definition

During frame reception the Rx Handler accesses the external Message RAM for acceptance filtering (read accesses) and for storing of the accepted messages (write accesses).

The time needed for acceptance filtering and for storing of a received message depends on

- The Host clock frequency
- The worst-case latency of the read and write accesses to the external Message RAM
- The number of configured filter elements
- The workload of the transmit message (Tx) handler in parallel to the receive message (Rx) handler

Received data bytes (DB0..DBm) from the CAN Core are buffered in the cache of the Rx Handler before they are written to the Message RAM (in words of 4 byte). Data words inside the Message RAM are numbered from R2 to Rn (n ≤ 17).

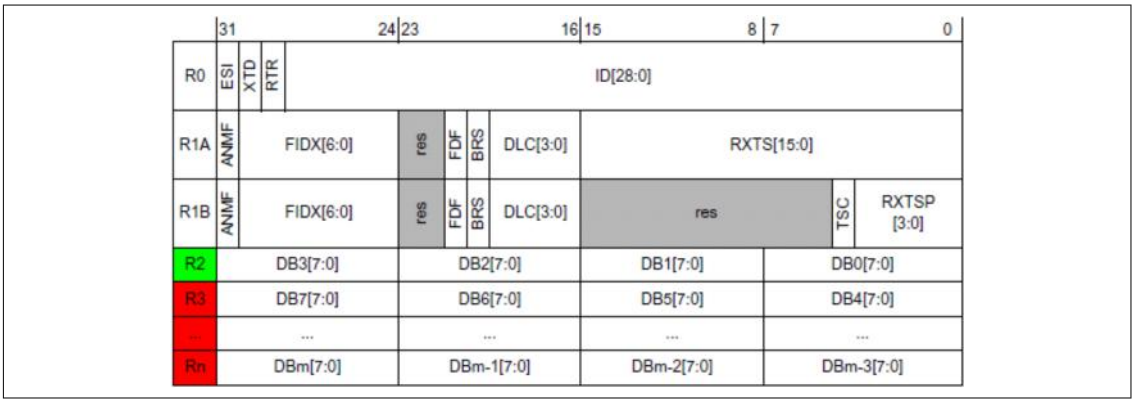


Figure 41 Rx buffer and FIFO element

Under the following conditions, a received message has corrupted data while the received message is signaled as valid to the host.

1. The data length code (DLC) of the received Message is greater than 4 (DLC > 4)
2. The storage of Ri of a received message into the Message RAM (after acceptance filtering is done) has not completed before R(i+1) is transferred from the CAN Core into the cache of the Rx Handler (where 2 ≤ i ≤ 5)
3. While condition 1) and 2) apply, a concurrent read of data word Ri from the cache and write of data word R(i+1) into the cache of the Rx handler happens

The data will be corrupted in a way, that in the Message RAM R(i+1) has the same content as Ri.

Despite the corrupted data, the M_TTCAN signals the storage of a valid frame in the Message RAM:

- Rx FIFO: FIFO put index RXFnS.FnPI is updated
- Dedicated Rx Buffer: New Data flag NDATn.NDxx is set
- Interrupt flag IR.MRAF is not set

The issue may occur in the FD Frame Format as well as in the Classic Frame Format.

Figure 42 shows how the available time for acceptance filtering and storage is reduced.

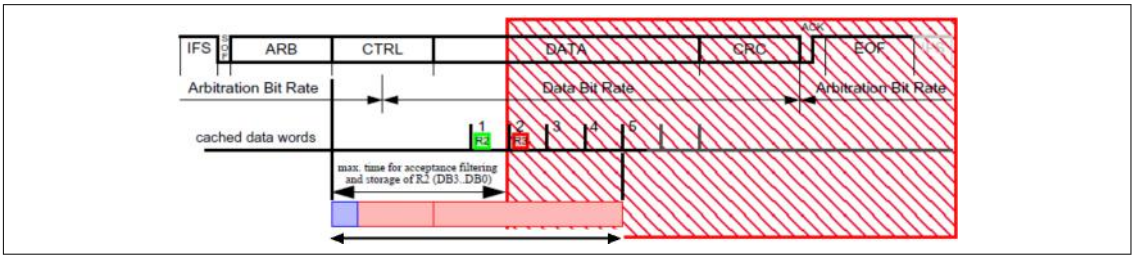


Figure 42 CAN Frame with DLC>4

Table 65 XMC5200: Minimum host clock frequency for CAN FD when DLC = 5

Number of configured active filter element 11-bit IDs/29-bit IDs[56) 57)]	Number of active CAN channels in an instance	Arbitration bit rate = 0.5 Mbps				Arbitration bit rate = 1 Mbps			
		Data bit rate = 0.5 Mbps	Data bit rate = 1 Mbps	Data bit rate = 2 Mbps	Data bit rate = 4 Mbps	Data bit rate = 1 Mbps	Data bit rate = 2 Mbps	Data bit rate = 4 Mbps	Data bit rate = 5 Mbps
32/16	2	3.9 MHz	7.1 MHz	13.1 MHz	22.8 MHz	7.7 MHz	14.1 MHz	26.1 MHz	31.5 MHz
	3	5.4 MHz	9.9 MHz	18.3 MHz	31.8 MHz	10.7 MHz	19.7 MHz	36.5 MHz	44.0 MHz
	4	6. 9 MHz	12.7 MHz	23.5 MHz	40.8 MHz	13.8 MHz	25.3 MHz	46.9 MHz	56.5 MHz
64/32	2	7.4 MHz	13.5 MHz	24.9 MHz	43.4 MHz	14.7 MHz	26.9 MHz	49.8 MHz	60.0 MHz
	3	10.3 MHz	18.8 MHz	34.9 MHz	60.7 MHz	20.5 MHz	37.6 MHz	69.7 MHz	84.0 MHz
	4	13.2 MHz	24.2 MHz	44.8 MHz	78.0 MHz	26.3 MHz	48.4 MHz	89.5 MHz	107.9 MHz[58)]
96/49	2	10. 8 MHz	19.9 MHz	36.8 MHz	64.0 MHz	21.6 MHz	39.7 MHz	73.5 MHz	88.6 MHz
	3	15.1 MHz	27.8 MHz	51.5 MHz	89.6 MHz	30.2 MHz	55.6 MHz	102.9[58)] MHz	124.0 MHz[58)]
	4	19.4 MHz	35.7 MHz	66.1 MHz	115.1 MHz[58)]	38.8 MHz	71.4 MHz	132.2 MHz[58)]	159.3 MHz[58)]
128/64	2	14.3 MHz	26.3 MHz	48.6 MHz	84.7 MHz	28.4 MHz	52.5 MHz	97.2 MHz	117.2 MHz[58)]

(table continues...)

56 M_TTCAN always starts at filter element #0 and proceeds through the filter list to find a matching element. Acceptance filtering stops at the first matching element and the following filter elements are not evaluated for this message. Therefore, the sequence of configured filter elements has a significant impact on the performance of the filtering process.

57 Acceptance filtering search for 11-bit IDs and 29-bit IDs filter element runs separately; only one configured filter setting should be considered. Searching for one 29-bit filter element requires approximately double cycles for one 11-bit filter element.

58 Frequency is not reachable since the maximum host clock frequency for M_TTCAN in XMC5200 is 100 MHz.

Table 65 (continued) XMC5200: Minimum host clock frequency for CAN FD when DLC = 5

Number of configured active filter element 11-bit IDs/29-bit IDs[56) 57)]	Number of active CAN channels in an instance	Arbitration bit rate = 0.5 Mbps				Arbitration bit rate = 1 Mbps			
		Data bit rate = 0.5 Mbps	Data bit rate = 1 Mbps	Data bit rate = 2 Mbps	Data bit rate = 4 Mbps	Data bit rate = 1 Mbps	Data bit rate = 2 Mbps	Data bit rate = 4 Mbps	Data bit rate = 5 Mbps
	3	20.0 MHz	36.8 MHz	68.0 MHz	118.5 MHz[58)]	40.0 MHz	73.5 MHz	136.0 MHz[58)]	164.0 MHz[58)]
	4	25.7 MHz	47.2 MHz	87.5 MHz	152.3 MHz[58)]	51.4 MHz	94.4 MHz	174.9 MHz[58)]	210.8 MHz[58)]

Parameters Affected	N/A
Trigger Condition(s)	Under the following conditions a received message has corrupted data while the received message is signaled as valid to the host: **1.** The data length code (DLC) of the received message is greater than 4 (DLC > 4) **2.** The storage of Ri of a received message into the Message RAM (after acceptance filtering is done) has not completed before R(i+1) is transferred from the CAN Core into the cache of the Rx Handler (where 2 ≤ i ≤ 5) **3.** While condition 1) and 2) apply, a concurrent read of data word Ri from the cache and write of data word R(i+1) into the cache of the Rx handler happens
Scope of Impact	The erratum is limited to the case when the Host clock frequency used in the actual device is below the limit shown in Table 1 Corrupted data is written to the Rx FIFO element from the respective dedicated Rx Buffer. The received frame is nevertheless signaled as valid.

56 M_TTCAN always starts at filter element #0 and proceeds through the filter list to find a matching element. Acceptance filtering stops at the first matching element and the following filter elements are not evaluated for this message. Therefore, the sequence of configured filter elements has a significant impact on the performance of the filtering process.

57 Acceptance filtering search for 11-bit IDs and 29-bit IDs filter element runs separately; only one configured filter setting should be considered. Searching for one 29-bit filter element requires approximately double cycles for one 11-bit filter element.

58 Frequency is not reachable since the maximum host clock frequency for M_TTCAN in XMC5200 is 100 MHz.

Workaround	Check whether the minimum Host clock frequency (shown in Table 1) is below the Host clock frequency used in the actual device. If yes, there is no problem with the selected configuration If no, use one of the following two workarounds. **1.** Try a different configuration by changing the following parameters until the actual Host clock frequency (CLK_GR5) is above the minimum host frequency shown in Table 1: **a.** Increase the CLK_GR5 frequency in the actual device **b.** Reduce the CAN-FD data bit rate **c.** Reduce the number of configured filter elements **d.** Reduce the number of active CAN channels in an instance Also, use DLC ≥ 8 instead of DLCs 5, 6, and 7 in the CAN environment/system, as they place higher demands on the minimum Host clock frequency (the worst case is DLC = 5) or restrict your CAN environment/system to DLC 4 ***Note**: While changing the actual host clock frequency, CLK_GR5 must always be equal to or higher than PCLK_- CANFD[x]_CLOCK_CAN[y] for all configurations.* **2.** Due to condition 3) listed in “Trigger Conditions”, the issue occurs only sporadically. Use an end-to-end (E2E) protection (for example, checksum or CRC covering the data field) and add it to all messages in the CAN system, to detect data corruption in the received frames.
Fix Status	No silicon fix planned. Use workaround.
Impact on Infineon software	Impact: Limitation Related modules: CAN, MCU Comment: The user must evaluate the impact of the erratum for each CAN instance separately. A CAN instance is the entirety of CanControllers with the same CanControllerInstance value. **1.** For the number of active CAN nodes: Use the maximum number of CanController configurations of a CAN instance that can be active (Autosar controller state STARTED or SLEEP) at a time **2.** For the host clock frequency: In McuPeriGroupSettings, locate the setting with McuPeriGroup = MCU_PERI_GROUP5_MMIO5 and take the value from McuPeriGroupClockFrequency **3.** **4.** For the number of configured active filter element 11-bit IDs/29-bit IDs: Use the corresponding values from the "Message RAM (…) linking table" in the generated `Can_PBcfg.h` file. Note that each CanController has its separate table. Take the maximum values **5.** For the arbitration bit rate: Use the maximum CanControllerBaudRate value of all the CanControllers **6.** For the data bit rate: Use the maximum CanControllerFdBaudRate value of all the CanControllers if configured. Otherwise use CanControllerBaudRate

Revision history

Document version	Date of release	Description of changes
**	2025-07-14	Initial release
*A	2025-08-21	Publish to web

Trademarks

All referenced product or service names and trademarks are the property of their respective owners.

Edition 2025-08-21

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
IFX-tdt1744261598375

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.

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.