



Welcome to [E-XFL.COM](https://www.e-xfl.com)

What is "[Embedded - Microcontrollers](#)"?

"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "[Embedded - Microcontrollers](#)"

Details

Product Status	Active
Core Processor	ARM® Cortex®-M0+
Core Size	32-Bit Single-Core
Speed	48MHz
Connectivity	I ² C, LINbus, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, I ² S, POR, PWM, WDT
Number of I/O	38
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	1.62V ~ 3.6V
Data Converters	A/D 14x12b; D/A 1x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	48-VFQFN Exposed Pad
Supplier Device Package	48-QFN (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsamd21g17a-mft

13. DSU - Device Service Unit

13.1 Overview

The Device Service Unit (DSU) provides a means to detect debugger probes. This enables the ARM Debug Access Port (DAP) to have control over multiplexed debug pads and CPU reset. The DSU also provides system-level services to debug adapters in an ARM debug system. It implements a CoreSight Debug ROM that provides device identification as well as identification of other debug components within the system. Hence, it complies with the ARM Peripheral Identification specification. The DSU also provides system services to applications that need memory testing, as required for IEC60730 Class B compliance, for example. The DSU can be accessed simultaneously by a debugger and the CPU, as it is connected on the High-Speed Bus Matrix. For security reasons, some of the DSU features will be limited or unavailable when the device is protected by the NVMCTRL security bit.

Related Links

[NVMCTRL – Non-Volatile Memory Controller Security Bit](#)

13.2 Features

- CPU reset extension
- Debugger probe detection (Cold- and Hot-Plugging)
- Chip-Erase command and status
- 32-bit cyclic redundancy check (CRC32) of any memory accessible through the bus matrix
- ARM® CoreSight™ compliant device identification
- Two debug communications channels
- Debug access port security filter
- Onboard memory built-in self-test (MBIST)

The presence of a debugger probe is detected when either Hot-Plugging or Cold-Plugging is detected. Once detected, the Debugger Present bit of the Status B register (STATUSB.DBGPRES) is set. For security reasons, Hot-Plugging is not available when the device is protected by the NVMCTRL security bit.

This detection requires that pads are correctly powered. Thus, at cold startup, this detection cannot be done until POR is released. If the device is protected, Cold-Plugging is the only way to detect a debugger probe, and so the external reset timing must be longer than the POR timing. If external reset is deasserted before POR release, the user must retry the procedure above until it gets connected to the device.

Related Links

[NVMCTRL – Non-Volatile Memory Controller Security Bit](#)

13.7 Chip Erase

Chip-Erase consists of removing all sensitive information stored in the chip and clearing the NVMCTRL security bit. Therefore, all volatile memories and the Flash memory (including the EEPROM emulation area) will be erased. The Flash auxiliary rows, including the user row, will not be erased.

When the device is protected, the debugger must reset the device in order to be detected. This ensures that internal registers are reset after the protected state is removed. The Chip-Erase operation is triggered by writing a '1' to the Chip-Erase bit in the Control register (CTRL.CE). This command will be discarded if the DSU is protected by the Peripheral Access Controller (PAC). Once issued, the module clears volatile memories prior to erasing the Flash array. To ensure that the Chip-Erase operation is completed, check the Done bit of the Status A register (STATUSA.DONE).

The Chip-Erase operation depends on clocks and power management features that can be altered by the CPU. For that reason, it is recommended to issue a Chip-Erase after a Cold-Plugging procedure to ensure that the device is in a known and safe state.

The recommended sequence is as follows:

1. Issue the Cold-Plugging procedure (refer to [Cold Plugging](#)). The device then:
 - 1.1. Detects the debugger probe.
 - 1.2. Holds the CPU in reset.
2. Issue the Chip-Erase command by writing a '1' to CTRL.CE. The device then:
 - 2.1. Clears the system volatile memories.
 - 2.2. Erases the whole Flash array (including the EEPROM emulation area, not including auxiliary rows).
 - 2.3. Erases the lock row, removing the NVMCTRL security bit protection.
3. Check for completion by polling STATUSA.DONE (read as one when completed).
4. Reset the device to let the NVMCTRL update fuses.

13.8 Programming

Programming the Flash or RAM memories is only possible when the device is not protected by the NVMCTRL security bit. The programming procedure is as follows:

1. At power up, $\overline{\text{RESET}}$ is driven low by a debugger. The on-chip regulator holds the system in a POR state until the input supply is above the POR threshold (refer to Power-On Reset (POR))

17.7 Register Summary

Offset	Name	Bit Pos.									
0x00	INTENCLR	7:0	DFLLCKC	DFLLCKF	DFLLOOB	DFLLRDY	OSC8MRDY	OSC32KRDY	XOSC32KRDY	XOSCRDY	
0x01		15:8	DPLLCKR				B33SRDY	BOD33DET	BOD33RDY	DFLLRCS	
0x02		23:16							DPLLTO	DPLLCKF	
0x03		31:24									
0x04	INTENSET	7:0	DFLLCKC	DFLLCKF	DFLLOOB	DFLLRDY	OSC8MRDY	OSC32KRDY	XOSC32KRDY	XOSCRDY	
0x05		15:8	DPLLCKR				B33SRDY	BOD33DET	BOD33RDY	DFLLRCS	
0x06		23:16							DPLLTO	DPLLCKF	
0x07		31:24									
0x08	INTFLAG	7:0	DFLLCKC	DFLLCKF	DFLLOOB	DFLLRDY	OSC8MRDY	OSC32KRDY	XOSC32KRDY	XOSCRDY	
0x09		15:8	DPLLCKR				B33SRDY	BOD33DET	BOD33RDY	DFLLRCS	
0x0A		23:16							DPLLTO	DPLLCKF	
0x0B		31:24									
0x0C	PCLKSR	7:0	DFLLCKC	DFLLCKF	DFLLOOB	DFLLRDY	OSC8MRDY	OSC32KRDY	XOSC32KRDY	XOSCRDY	
0x0D		15:8	DPLLCKR				B33SRDY	BOD33DET	BOD33RDY	DFLLRCS	
0x0E		23:16							DPLLTO	DPLLCKF	
0x0F		31:24									
0x10	XOSC	7:0	ONDEMAND	RUNSTDBY				XTALEN	ENABLE		
0x11		15:8	STARTUP[3:0]					AMPGC	GAIN[2:0]		
0x12 ... 0x13	Reserved										
0x14	XOSC32K	7:0	ONDEMAND	RUNSTDBY	AAMPEN		EN32K	XTALEN	ENABLE		
0x15		15:8				WRTLOCK		STARTUP[2:0]			
0x16 ... 0x17	Reserved										
0x18	OSC32K	7:0	ONDEMAND	RUNSTDBY				EN32K	ENABLE		
0x19		15:8				WRTLOCK		STARTUP[2:0]			
0x1A		23:16		CALIB[6:0]							
0x1B		31:24									
0x1C	OSCULP32K	7:0	WRTLOCK			CALIB[4:0]					
0x1D ... 0x1F	Reserved										
0x20	OSC8M	7:0	ONDEMAND	RUNSTDBY					ENABLE		
0x21		15:8							PRESC[1:0]		
0x22		23:16	CALIB[7:0]								
0x23		31:24	FRANGE[1:0]					CALIB[11:8]			
0x24	DFLLCTRL	7:0	ONDEMAND	RUNSTDBY	USBCRM	LLAW	STABLE	MODE	ENABLE		
0x25		15:8					WAITLOCK	BPLCKC	QLDIS	CCDIS	

32-bit ARM-Based Microcontrollers

Bit 13 – FORCELDO: Force LDO Voltage Regulator

Value	Description
0	The voltage regulator is in low power and low drive configuration in standby sleep mode.
1	The voltage regulator is in low power and high drive configuration in standby sleep mode.

Bit 6 – RUNSTDBY: Run in Standby

Value	Description
0	The voltage regulator is in low power configuration in standby sleep mode.
1	The voltage regulator is in normal configuration in standby sleep mode.

17.8.16 Voltage References System (VREF) Control

Name: VREF

Offset: 0x40

Reset: 0x0XXX0000

Property: Write-Protected

Bit	31	30	29	28	27	26	25	24
						CALIB[10:8]		
Access						R/W	R/W	R/W
Reset						x	x	x
Bit	23	22	21	20	19	18	17	16
	CALIB[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	x	x	x	x	x	x	x	x
Bit	15	14	13	12	11	10	9	8
Access								
Reset								
Bit	7	6	5	4	3	2	1	0
						BGOUTEN	TSEN	
Access						R/W	R/W	
Reset						0	0	

Bits 26:16 – CALIB[10:0]: Bandgap Voltage Generator Calibration

These bits are used to calibrate the output level of the bandgap voltage reference. These bits are loaded from Flash Calibration Row at startup.

Bit 2 – BGOUTEN: Bandgap Output Enable

Value	Description
0	The bandgap output is not available as an ADC input channel.
1	The bandgap output is routed to an ADC input channel.

Bit 1 – TSEN: Temperature Sensor Enable

18.6.5 Synchronization

Due to asynchronicity between the main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read.

When executing an operation that requires synchronization, the Synchronization Busy bit in the Status register (STATUS.SYNCBUSY) will be set immediately, and cleared when synchronization is complete. The Synchronization Ready interrupt can be used to signal when synchronization is complete. This can be accessed via the Synchronization Ready Interrupt Flag in the Interrupt Flag Status and Clear register (INTFLAG.SYNCRDY).

If an operation that requires synchronization is executed while STATUS.SYNCBUSY='1', the bus will be stalled. All operations will complete successfully, but the CPU will be stalled and interrupts will be pending as long as the bus is stalled.

The following registers are synchronized when written:

- Control register (CTRL)
- Clear register (CLEAR)

Required write-synchronization is denoted by the "Write-Synchronized" property in the register description.

Related Links

[Register Synchronization](#)

18.7 Register Summary

Register summary

Offset	Name	Bit Pos.								
0x0	CTRL	7:0	ALWAYSON					WEN	ENABLE	
0x1	CONFIG	7:0	WINDOW[3:0]				PER[3:0]			
0x2	EWCTRL	7:0					EWOFFSET[3:0]			
0x3	Reserved									
0x4	INTENCLR	7:0								EW
0x5	INTENSET	7:0								EW
0x6	INTFLAG	7:0								EW
0x7	STATUS	7:0	SYNCBUSY							
0x8	CLEAR	7:0	CLEAR[7:0]							

18.8 Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers require synchronization when read and/or written. Synchronization is denoted by the "Read-Synchronized" and/or "Write-Synchronized" property in each individual register description.

Some registers are enable-protected, meaning they can only be written when the module is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.

Value	Name	Description
0x2E	CHN	DMA channel 14
0x2F	CHN	DMA channel 15
0x30	CHN	DMA channel 16
0x31	CHN	DMA channel 17
0x32	CHN	DMA channel 18
0x33	CHN	DMA channel 19
0x34	CHN	DMA channel 20
0x35	CHN	DMA channel 21
0x36	CHN	DMA channel 22
0x37	CHN	DMA channel 23
0x38	CHN	DMA channel 24
0x39	CHN	DMA channel 25
0x3A	CHN	DMA channel 26
0x3B	CHN	DMA channel 27
0x3C	CHN	DMA channel 28
0x3D	CHN	DMA channel 29
0x3E	CHN	DMA channel 30
0x3F	CHN	DMA channel 31

Bits 3:2 – CRCPOLY[1:0]: CRC Polynomial Type

These bits define the size of the data transfer for each bus access when the CRC is used with I/O interface, as shown in the table below.

Value	Name	Description
0x0	CRC16	CRC-16 (CRC-CCITT)
0x1	CRC32	CRC32 (IEEE 802.3)
0x2-0x3		Reserved

Bits 1:0 – CRCBEATSIZE[1:0]: CRC Beat Size

These bits define the size of the data transfer for each bus access when the CRC is used with I/O interface.

Value	Name	Description
0x0	BYTE	8-bit bus transfer
0x1	WORD	16-bit bus transfer
0x2	WORD	32-bit bus transfer
0x3		Reserved

20.8.3 CRC Data Input

Name: CRCDATAIN
Offset: 0x04
Reset: 0x00000000
Property: PAC Write-Protection

- Address register (ADDR)

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

PAC write-protection does not apply to accesses through an external debugger.

Related Links

[PAC - Peripheral Access Controller](#)

25.5.9 Analog Connections

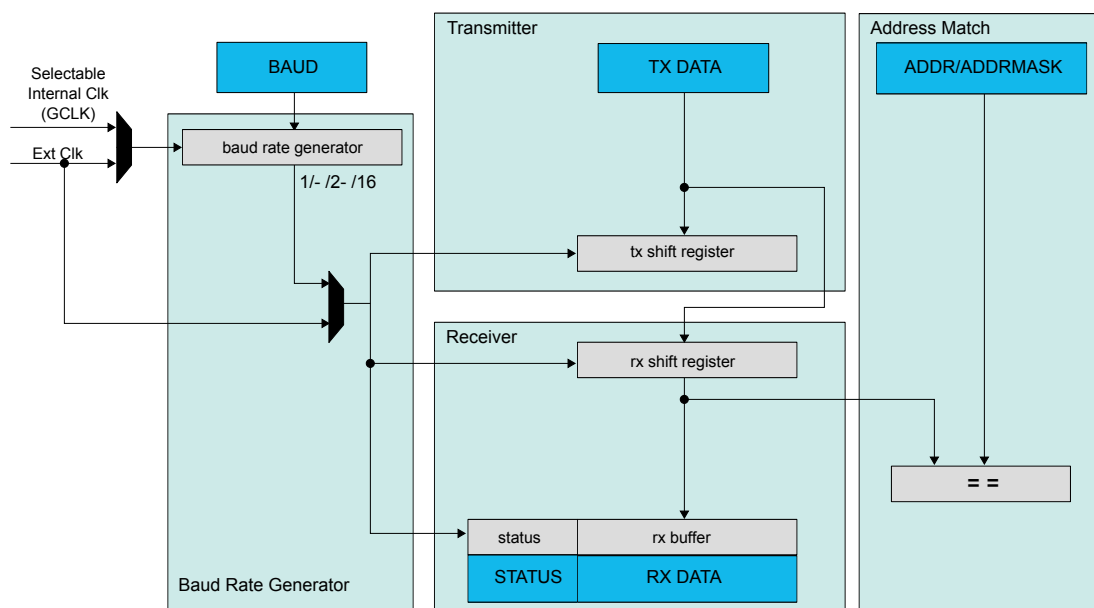
Not applicable.

25.6 Functional Description

25.6.1 Principle of Operation

The basic structure of the SERCOM serial engine is shown in [Figure 25-2](#). Labels in capital letters are synchronous to the system clock and accessible by the CPU; labels in lowercase letters can be configured to run on the GCLK_SERCOMx_CORE clock or an external clock.

Figure 25-2. SERCOM Serial Engine



The transmitter consists of a single write buffer and a shift register.

The receiver consists of a two-level receive buffer and a shift register.

The baud-rate generator is capable of running on the GCLK_SERCOMx_CORE clock or an external clock.

Address matching logic is included for SPI and I²C operation.

25.6.2 Basic Operation

25.6.2.1 Initialization

The SERCOM must be configured to the desired mode by writing the Operating Mode bits in the Control A register (CTRLA.MODE). Refer to table SERCOM Modes for details.

- D represent the data bits per frame
- S represent the sum of start and first stop bits, if present.

Table 25-3 shows the BAUD register value versus baud frequency f_{BAUD} at a serial engine frequency of 48MHz. This assumes a D value of 8 bits and an S value of 2 bits (10 bits, including start and stop bits).

Table 25-3. BAUD Register Value vs. Baud Frequency

BAUD Register Value	Serial Engine CPF	f_{BAUD} at 48MHz Serial Engine Frequency (f_{REF})
0 – 406	160	3MHz
407 – 808	161	2.981MHz
809 – 1205	162	2.963MHz
...
65206	31775	15.11kHz
65207	31871	15.06kHz
65208	31969	15.01kHz

25.6.3 Additional Features

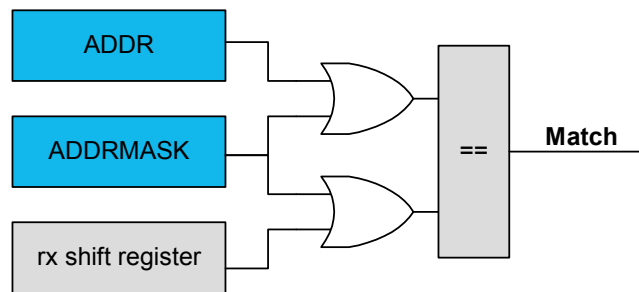
25.6.3.1 Address Match and Mask

The SERCOM address match and mask feature is capable of matching either one address, two unique addresses, or a range of addresses with a mask, based on the mode selected. The match uses seven or eight bits, depending on the mode.

Address With Mask

An address written to the Address bits in the Address register (ADDR.ADDR), and a mask written to the Address Mask bits in the Address register (ADDR.ADDRMASK) will yield an address match. All bits that are masked are not included in the match. Note that writing the ADDR.ADDRMASK to 'all zeros' will match a single unique address, while writing ADDR.ADDRMASK to 'all ones' will result in all addresses being accepted.

Figure 25-4. Address With Mask



Two Unique Addresses

The two addresses written to ADDR and ADDRMASK will cause a match.

32-bit ARM-Based Microcontrollers

Property: PAC Write-Protection, Enable-Protected, Write-Synchronized

Bit	31	30	29	28	27	26	25	24
Access								
Reset								
Bit	23	22	21	20	19	18	17	16
							RXEN	TXEN
Access							R/W	R/W
Reset							0	0
Bit	15	14	13	12	11	10	9	8
			PMODE			ENC	SFDE	COLDEN
Access			R/W			R/W	R/W	R/W
Reset			0			0	0	0
Bit	7	6	5	4	3	2	1	0
		SBMODE					CHSIZE[2:0]	
Access		R/W				R/W	R/W	R/W
Reset		0				0	0	0

Bit 17 – RXEN: Receiver Enable

Writing '0' to this bit will disable the USART receiver. Disabling the receiver will flush the receive buffer and clear the FERR, PERR and BUFOVF bits in the STATUS register.

Writing '1' to CTRLB.RXEN when the USART is disabled will set CTRLB.RXEN immediately. When the USART is enabled, CTRLB.RXEN will be cleared, and SYNCBUSY.CTRLB will be set and remain set until the receiver is enabled. When the receiver is enabled, CTRLB.RXEN will read back as '1'.

Writing '1' to CTRLB.RXEN when the USART is enabled will set SYNCBUSY.CTRLB, which will remain set until the receiver is enabled, and CTRLB.RXEN will read back as '1'.

This bit is not enable-protected.

Value	Description
0	The receiver is disabled or being enabled.
1	The receiver is enabled or will be enabled when the USART is enabled.

Bit 16 – TXEN: Transmitter Enable

Writing '0' to this bit will disable the USART transmitter. Disabling the transmitter will not become effective until ongoing and pending transmissions are completed.

Writing '1' to CTRLB.TXEN when the USART is disabled will set CTRLB.TXEN immediately. When the USART is enabled, CTRLB.TXEN will be cleared, and SYNCBUSY.CTRLB will be set and remain set until the transmitter is enabled. When the transmitter is enabled, CTRLB.TXEN will read back as '1'.

Writing '1' to CTRLB.TXEN when the USART is enabled will set SYNCBUSY.CTRLB, which will remain set until the receiver is enabled, and CTRLB.TXEN will read back as '1'.

This bit is not enable-protected.

register (INTFLAG.DRDY), indicating data are needed for transmit. If a NACK is sent, the I²C slave will wait for a new start condition and address match.

Typically, software will immediately acknowledge the address packet by sending an ACK/NACK bit. The I²C slave Command bit field in the Control B register (CTRLB.CMD) can be written to '0x3' for both read and write operations as the command execution is dependent on the STATUS.DIR bit. Writing '1' to INTFLAG.AMATCH will also cause an ACK/NACK to be sent corresponding to the CTRLB.ACKACT bit.

Case 2: Address packet accepted – Write flag set

The STATUS.DIR bit is cleared, indicating an I²C master write operation. The SCL line is forced low, stretching the bus clock. If an ACK is sent, the I²C slave will wait for data to be received. Data, repeated start or stop can be received.

If a NACK is sent, the I²C slave will wait for a new start condition and address match. Typically, software will immediately acknowledge the address packet by sending an ACK/NACK. The I²C slave command CTRLB.CMD = 3 can be used for both read and write operation as the command execution is dependent on STATUS.DIR.

Writing '1' to INTFLAG.AMATCH will also cause an ACK/NACK to be sent corresponding to the CTRLB.ACKACT bit.

Receiving Address Packets (SCLSM=1)

When SCLSM=1, the I²C slave will stretch the SCL line only after an ACK, see [Slave Behavioral Diagram \(SCLSM=1\)](#). When the I²C slave is properly configured, it will wait for a start condition to be detected.

When a start condition is detected, the successive address packet will be received and checked by the address match logic.

If the received address is not a match, the packet will be rejected and the I²C slave will wait for a new start condition.

If the address matches, the acknowledge action as configured by the Acknowledge Action bit Control B register (CTRLB.ACKACT) will be sent and the Address Match bit in the Interrupt Flag register (INTFLAG.AMATCH) is set. SCL will be stretched until the I²C slave clears INTFLAG.AMATCH. As the I²C slave holds the clock by forcing SCL low, the software is given unlimited time to respond to the address.

The direction of a transaction is determined by reading the Read/Write Direction bit in the Status register (STATUS.DIR). This bit will be updated only when a valid address packet is received.

If the Transmit Collision bit in the Status register (STATUS.COLL) is set, the last packet addressed to the I²C slave had a packet collision. A collision causes the SDA and SCL lines to be released without any notification to software. The next AMATCH interrupt is, therefore, the first indication of the previous packet's collision. Collisions are intended to follow the SMBus Address Resolution Protocol (ARP).

After the address packet has been received from the I²C master, INTFLAG.AMATCH be set to '1' to clear it.

Receiving and Transmitting Data Packets

After the I²C slave has received an address packet, it will respond according to the direction either by waiting for the data packet to be received or by starting to send a data packet by writing to DATA.DATA. When a data packet is received or sent, INTFLAG.DRDY will be set. After receiving data, the I²C slave will send an acknowledge according to CTRLB.ACKACT.

Case 1: Data received

INTFLAG.DRDY is set, and SCL is held low, pending for SW interaction.

Writing a zero to these bits has no effect.

Writing a valid value to these bits will set a command.

Value	Name	Description
0x0	DISABLE	Command disabled: IDX toggles between cycles A and B
0x1	SET	Set IDX: cycle B will be forced in the next cycle
0x2	CLEAR	Clear IDX: cycle A will be forced in next cycle
0x3	HOLD	Hold IDX: the next cycle will be the same as the current cycle.

Bit 2 – ONESHOT: One-Shot

This bit controls one-shot operation of the TCC. When in one-shot operation, the TCC will stop counting on the next overflow/underflow condition or a stop command.

Writing a '0' to this bit has no effect.

Writing a '1' to this bit will enable the one-shot operation.

Value	Description
0	The TCC will count continuously.
1	The TCC will stop counting on the next underflow/overflow condition.

Bit 1 – LUPD: Lock Update

This bit controls the update operation of the TCC buffered registers.

When CTRLB.LUPD is set, no any update of the registers with value of its buffered register is performed on hardware UPDATE condition. Locking the update ensures that all buffer registers are valid before an hardware update is performed. After all the buffer registers are loaded correctly, the buffered registers can be unlocked.

This bit has no effect when input capture operation is enabled.

Writing a '0' to this bit has no effect.

Writing a '1' to this bit will lock updating.

Value	Description
0	The CCBx, PERB, PGVB, PGOB, and SWAPBx buffer registers values <i>are</i> copied into the corresponding CCx, PER, PGV, PGO and SWAPx registers on hardware update condition.
1	The CCBx, PERB, PGVB, PGOB, and SWAPBx buffer registers values are <i>not</i> copied into CCx, PER, PGV, PGO and SWAPx registers on hardware update condition.

Bit 0 – DIR: Counter Direction

This bit is used to change the direction of the counter.

Writing a '0' to this bit has no effect

Writing a '1' to this bit will clear the bit and make the counter count up.

Value	Description
0	The timer/counter is counting up (incrementing).
1	The timer/counter is counting down (decrementing).

31.8.4 Synchronization Busy

Name: SYNCBUSY

Offset: 0x08

32-bit ARM-Based Microcontrollers

Offset: 0x28

Reset: 0x00000000

Property: PAC Write-Protection

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
					MC3	MC2	MC1	MC0
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	15	14	13	12	11	10	9	8
	FAULT1	FAULT0	FAULTB	FAULTA	DFS			
Access	R/W	R/W	R/W	R/W	R/W			
Reset	0	0	0	0	0			

Bit	7	6	5	4	3	2	1	0
					ERR	CNT	TRG	OVF
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bits 19,18,17,16 – MCx: Match or Capture Channel x Interrupt Enable

Writing a '0' to this bit has no effect.

Writing a '1' to this bit will set the corresponding Match or Capture Channel x Interrupt Disable/Enable bit, which enables the Match or Capture Channel x interrupt.

Value	Description
0	The Match or Capture Channel x interrupt is disabled.
1	The Match or Capture Channel x interrupt is enabled.

Bits 15,14 – FAULTx: Non-Recoverable Fault x Interrupt Enable

Writing a '0' to this bit has no effect.

Writing a '1' to this bit will set the Non-Recoverable Fault x Interrupt Disable/Enable bit, which enables the Non-Recoverable Fault x interrupt.

Value	Description
0	The Non-Recoverable Fault x interrupt is disabled.
1	The Non-Recoverable Fault x interrupt is enabled.

Bit 13 – FAULTB: Recoverable Fault B Interrupt Enable

Writing a '0' to this bit has no effect.

Writing a '1' to this bit will set the Recoverable Fault B Interrupt Disable/Enable bit, which enables the Recoverable Fault B interrupt.

Bits 16, 17, 18, 19 – POLn: Channel Polarity x

Setting these bits enables the output polarity in single-slope and dual-slope PWM operations.

Value	Name	Description
0	(single-slope PWM waveform generation)	Compare output is initialized to ~DIR and set to DIR when TCC counter matches CCx value
1	(single-slope PWM waveform generation)	Compare output is initialized to DIR and set to ~DIR when TCC counter matches CCx value.
0	(dual-slope PWM waveform generation)	Compare output is set to ~DIR when TCC counter matches CCx value
1	(dual-slope PWM waveform generation)	Compare output is set to DIR when TCC counter matches CCx value.

Bits 8, 9, 10, 11 – CICCENn: Circular CC Enable x

Setting this bits enables the compare circular buffer option on channel. When the bit is set, CCx register value is copied-back into the CCx register on UPDATE condition.

Bit 7 – CIPEREN: Circular Period Enable

Setting this bits enable the period circular buffer option. When the bit is set, the PER register value is copied-back into the PERB register on UPDATE condition.

Bits 5:4 – RAMP[1:0]: Ramp Operation

These bits select Ramp operation (RAMP). These bits are not synchronized.

Value	Name	Description
0x0	RAMP1	RAMP1 operation
0x1	RAMP2A	Alternative RAMP2 operation
0x2	RAMP2	RAMP2 operation
0x3	-	Reserved

Bits 2:0 – WAVEGEN[2:0]: Waveform Generation Operation

These bits select the waveform generation operation. The settings impact the top value and control if frequency or PWM waveform generation should be used. These bits are not synchronized.

Value	Name	Description						
		Operation	Top	Update	Waveform Output On Match	Waveform Output On Update	OVFIF/Event Up Down	
0x0	NFRQ	Normal Frequency	PER	TOP/Zero	Toggle	Stable	TOP	Zero
0x1	MFRQ	Match Frequency	CC0	TOP/Zero	Toggle	Stable	TOP	Zero
0x2	NPWM	Normal PWM	PER	TOP/Zero	Set	Clear	TOP	Zero
0x3	Reserved	–	–	–	–	–	–	–
0x4	DSCRITICAL	Dual-slope PWM	PER	Zero	~DIR	Stable	–	Zero
0x5	DSBOTTOM	Dual-slope PWM	PER	Zero	~DIR	Stable	–	Zero
0x6	DSBOTH	Dual-slope PWM	PER	TOP & Zero	~DIR	Stable	TOP	Zero
0x7	DSTOP	Dual-slope PWM	PER	Zero	~DIR	Stable	TOP	–

31.8.17 Period Value

Name: PER

This bit is also set by the hardware:

- When a STALL handshake has been received.
- After a PIPE has been enabled (rising of bit PEN.N).
- When an LPM transaction has completed whatever handshake is returned or the transaction was timed-out.
- When a pipe transfer was completed with a pipe error. See [PINTFLAG](#) register.

When PFREEZE bit is set while a transaction is in progress on the USB bus, this transaction will be properly completed. PFREEZE bit will be read as “1” only when the ongoing transaction will have been completed.

Value	Description
0	The Pipe operates in normal operation.
1	The Pipe is frozen and no additional requests will be sent to the device on this pipe address.

Bit 2 – CURBK: Current Bank

Value	Description
0	The bank0 is the bank that will be used in the next single/multi USB packet.
1	The bank1 is the bank that will be used in the next single/multi USB packet.

Bit 0 – DTGL: Data Toggle Sequence

Writing a one to the bit EPSTATUSCLR.DTGL will clear this bit.

Writing a one to the bit EPSTATUSSET.DTGL will set this bit.

This bit is toggled automatically by hardware after a data transaction.

This bit will reflect the data toggle in regards of the token type (IN/OUT/SETUP).

Value	Description
0	The PID of the next expected transaction will be zero: data 0.
1	The PID of the next expected transaction will be one: data 1.

32.8.6.6 Host Pipe Interrupt Flag Register

Name: PINTFLAG

Offset: 0x107 + (n x 0x20)

Reset: 0x00

Property: -

Bit	7	6	5	4	3	2	1	0
			STALL	TXSTP	PERR	TRFAIL		TRCPT
Access			R/W	R/W	R/W	R/W		R/W
Reset			0	0	0	0		0

Bit 5 – STALL: STALL Received Interrupt Flag

This flag is cleared by writing a one to the flag.

This flag is set when a stall occurs and will generate an interrupt if PINTENCLR/SET.STALL is one.

Writing a zero to this bit has no effect.

Writing a one to this bit clears the STALL Interrupt Flag.

32.8.6.8 Host Interrupt Pipe Set Register

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Pipe Interrupt Enable Set (PINTENCLR) register.

This register is cleared by USB reset or when PEN[n] is zero.

Name: PINTENSET
Offset: 0x109 + (n x 0x20)
Reset: 0x00
Property: PAC Write-Protection

Bit	7	6	5	4	3	2	1	0
			STALL	TXSTP	PERR	TRFAIL		TRCPT
Access			R/W	R/W	R/W	R/W		R/W
Reset			0	0	0	0		0

Bit 5 – STALL: Stall Interrupt Enable

Writing a zero to this bit has no effect.

Writing a one to this bit will enable the Stall interrupt.

Value	Description
0	The Stall interrupt is disabled.
1	The Stall interrupt is enabled.

Bit 4 – TXSTP: Transmitted Setup Interrupt Enable

Writing a zero to this bit has no effect.

Writing a one to this bit will enable the Transmitted Setup interrupt.

Value	Description
0	The Transmitted Setup interrupt is disabled.
1	The Transmitted Setup interrupt is enabled.

Bit 3 – PERR: Pipe Error Interrupt Enable

Writing a zero to this bit has no effect.

Writing a one to this bit will enable the Pipe Error interrupt.

Value	Description
0	The Pipe Error interrupt is disabled.
1	The Pipe Error interrupt is enabled.

Bit 2 – TRFAIL: Transfer Fail Interrupt Enable

Writing a zero to this bit has no effect.

Writing a one to this bit will enable the Transfer Fail interrupt.

Value	Description
0	The Transfer Fail interrupt is disabled.
1	The Transfer Fail interrupt is enabled.

Bit 0 – TRCPT: Transfer Complete x interrupt Enable

Writing a zero to this bit has no effect.

(OFFSETCORR). The offset correction value is subtracted from the converted data before writing the Result register (RESULT).

The gain error is defined as the deviation of the last output step's midpoint from the ideal straight line, after compensating for offset error. The gain error cancellation is handled by the Gain Correction register (GAINCORR).

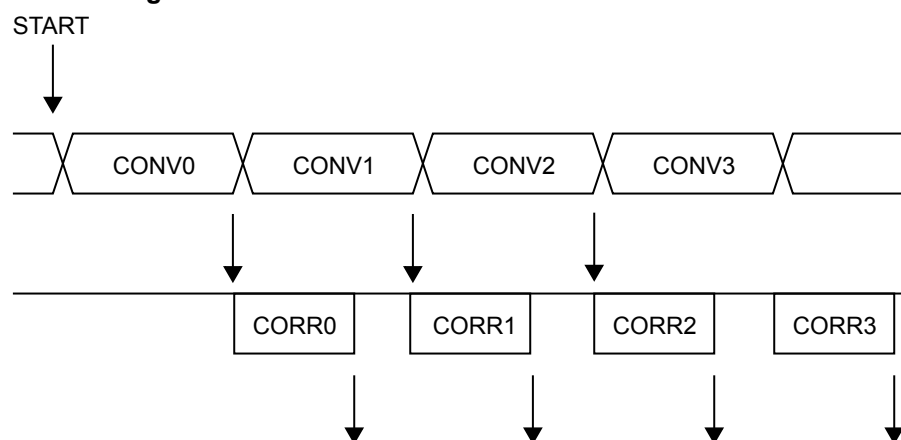
To correct these two errors, the Digital Correction Logic Enabled bit in the Control B register (CTRLB.CORREN) must be set to "1".

Offset and gain error compensation results are both calculated according to:

$$\text{Result} = (\text{Conversion value} + \text{OFFSETCORR}) \cdot \text{GAINCORR}$$

The correction will introduce a latency of 13 CLK_ADC clock cycles. In free running mode this latency is introduced on the first conversion only, since its duration is always less than the propagation delay. In single conversion mode this latency is introduced for each conversion.

Figure 33-8. ADC Timing Correction Enabled



33.6.11 DMA Operation

The ADC generates the following DMA request:

- Result Conversion Ready (RESRDY): the request is set when a conversion result is available and cleared when the RESULT register is read. When the averaging operation is enabled, the DMA request is set when the averaging is completed and result is available.

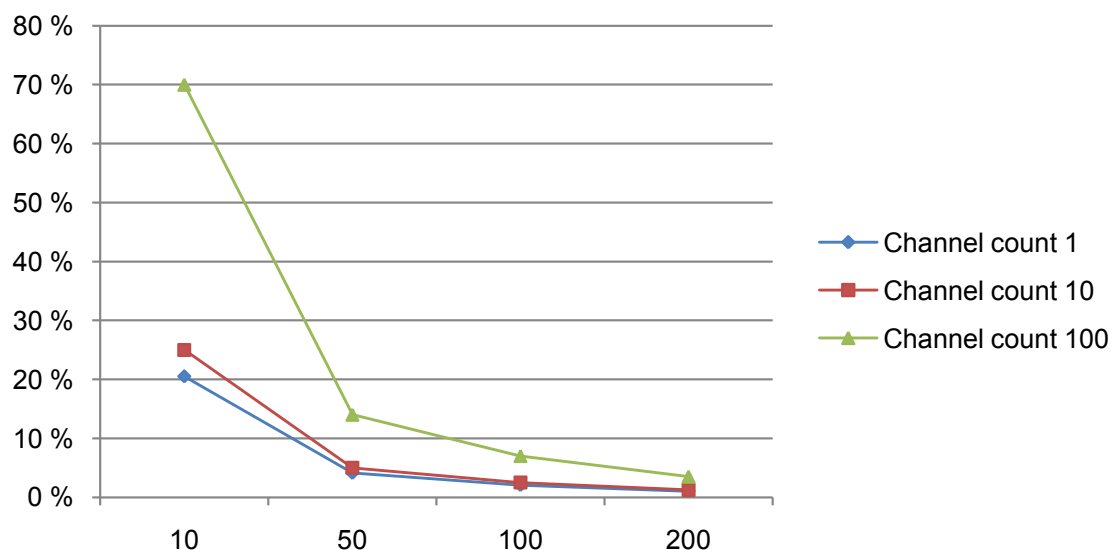
33.6.12 Interrupts

The ADC has the following interrupt sources:

- Result Conversion Ready: RESRDY
- Window Monitor: WINMON
- Overrun: OVERRUN

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear (INTFLAG) register is set when the interrupt condition occurs. Each interrupt can be individually enabled by writing a one to the corresponding bit in the Interrupt Enable Set (INTENSET) register, and disabled by writing a one to the corresponding bit in the Interrupt Enable Clear (INTENCLR) register. An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled, or the ADC is reset. An interrupt flag is cleared by writing a one to the corresponding bit in the INTFLAG register. Each peripheral can have one interrupt request line per interrupt source or one common interrupt request line for all the interrupt sources. This is device dependent.

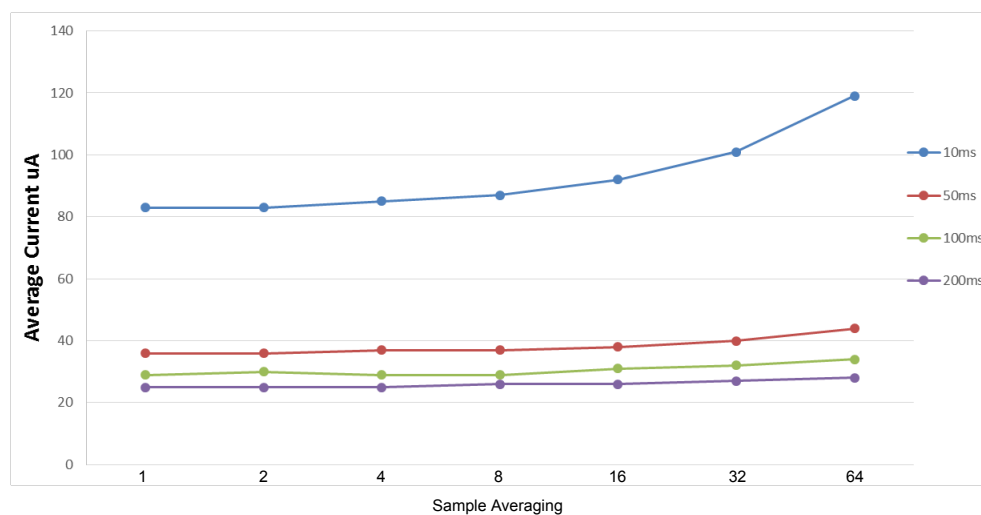
Figure 37-13. CPU Utilization



37.13.2 Device Variant B and C

$V_{CC} = 3.3V$ and $f_{CPU} = 48MHz$ for the following PTC measurements.

Figure 37-14. 1 Sensor / PTC_GCLK = 4MHz / FREQ_MODE_NONE



38.2.7 32 pin TQFP

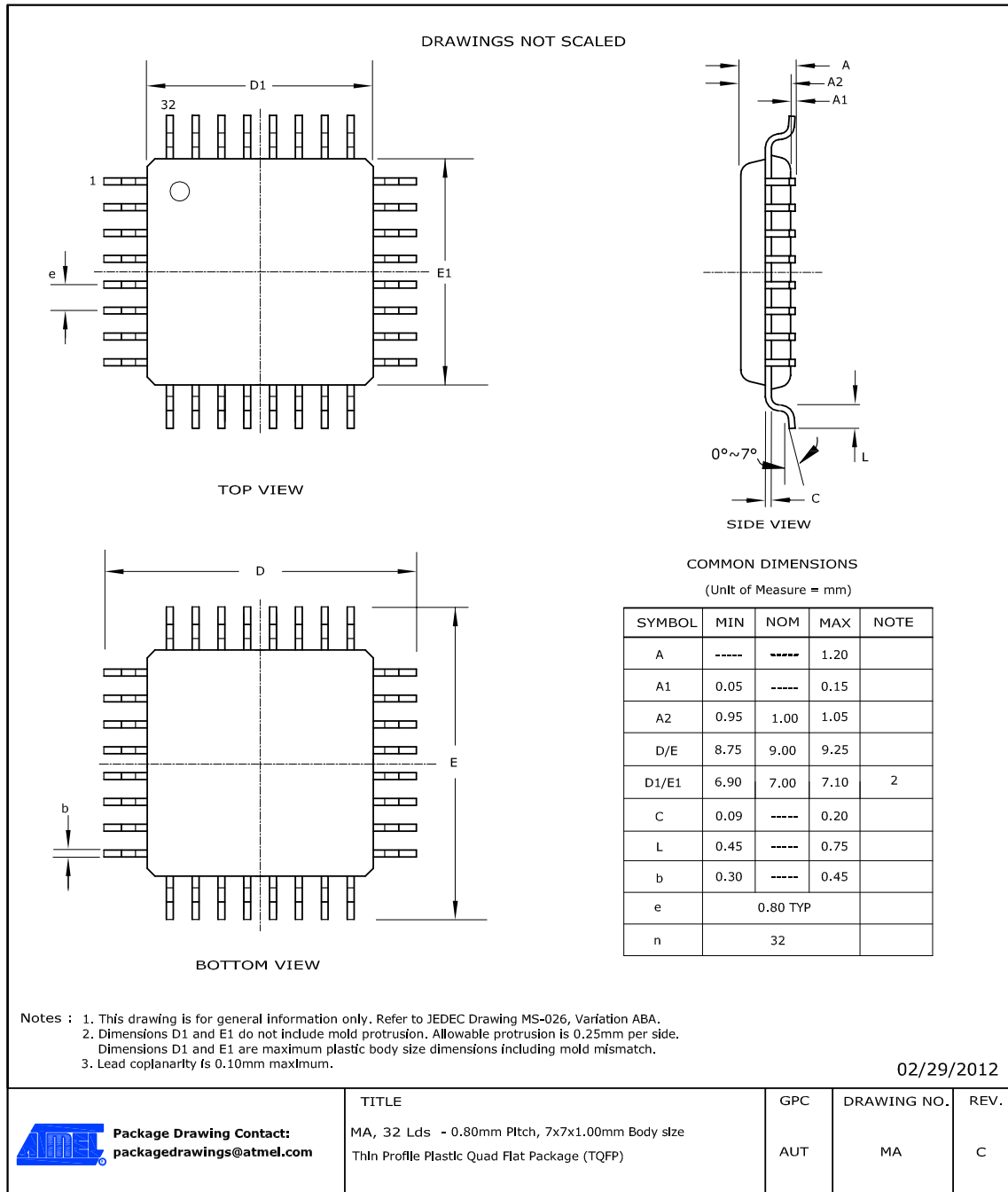


Table 38-20. Device and Package Maximum Weight

100	mg
-----	----

Table 38-21. Package Characteristics

Moisture Sensitivity Level	MSL3
----------------------------	------

10 – When the RUNSTDBY bit is written after the TCC is enabled, the respective TCC APB bus is stalled and the RUNDSTBY bit in the TCC CTRLA register is not enabled-protected.

Errata reference: 12477

Fix/Workaround:

None.

11 – TCC fault filtering on inverted fault is not working.

Errata reference: 12512

Fix/Workaround:

Use only non-inverted faults.

12 – When waking up from the STANDBY power save mode, the SYNCBUSY.CTRLB, SYNCBUSY.STATUS, SYNCBUSY.COUNT, SYNCBUSY.PATT, SYNCBUSY.WAVE, SYNCBUSY.PER and SYNCBUSY.CCx bits may be locked to 1.

Errata reference: 12227

Fix/Workaround:

After waking up from STANDBY power save mode, perform a software reset of the TCC if you are using the SYNCBUSY.CTRLB, SYNCBUSY.STATUS, SYNCBUSY.COUNT, SYNCBUSY.PATT, SYNCBUSY.WAVE, SYNCBUSY.PER or SYNCBUSY.CCx bits

13 – When the Peripheral Access Controller (PAC) protection is enabled, writing to WAVE or WAVEB registers will not cause a hardware exception.

Errata reference: 11468

Fix/Workaround:

None

14 – If the MCx flag in the INTFLAG register is set when enabling the DMA, this will trigger an immediate DMA transfer and overwrite the current buffered value in the TCC register.

Errata reference: 12155

Fix/Workaround:

None

40.1.1.13 USB

1 – The FLENC register negative sign management is not correct.

Errata reference: 11472

Fix/Workaround:

The following rule must be used for negative values:

- FLENC 8h is equal to 0 decimal.
- FLENC 9h to Fh are equal to -1 to -7 decimal instead of -7 to -1.

40.1.1.14 PTC

1 – WCOMP interrupt flag is not stable. The WCOMP interrupt flag will not always be set as described in the datasheet.

Errata reference: 12860

Fix/Workaround:

Do not use the WCOMP interrupt. Use the WCOMP event.

3 – In RAMP 2 mode with Fault keep, qualified and restart:

If a fault occurred at the end of the period during the qualified state, the switch to the next ramp can have two restarts.

Errata reference: 13262

Fix/Workaround:

Avoid faults few cycles before the end or the beginning of a ramp.

4 – With blanking enabled, a recoverable fault that occurs during the first increment of a rising TCC is not blanked.

Errata reference: 12519

Fix/Workaround:

None

5 – In Dual slope mode a Retrigger Event does not clear the TCC counter.

Errata reference: 12354

Fix/Workaround:

None

6 – In two ramp mode, two events will be generated per cycle, one on each ramp's end. EVCTRL.CNTSEL.END cannot be used to identify the end of a double ramp cycle.

Errata reference: 12224

Fix/Workaround:

None

7 – If an input event triggered STOP action is performed at the same time as the counter overflows, the first pulse width of the subsequent counter start can be altered with one prescaled clock cycle.

Errata reference: 12107

Fix/Workaround:

None

8 – When the RUNSTDBY bit is written after the TCC is enabled, the respective TCC APB bus is stalled and the RUNDSTBY bit in the TCC CTRLA register is not enabled-protected.

Errata reference: 12477

Fix/Workaround:

None.

9 – TCC fault filtering on inverted fault is not working.

Errata reference: 12512

Fix/Workaround:

Use only non-inverted faults.

10 – When waking up from the STANDBY power save mode, the SYNCBUSY.CTRLB, SYNCBUSY.STATUS, SYNCBUSY.COUNT, SYNCBUSY.PATT, SYNCBUSY.WAVE, SYNCBUSY.PER and SYNCBUSY.CCx bits may be locked to 1.

Errata reference: 12227

Fix/Workaround:

After waking up from STANDBY power save mode, perform a software reset of the TCC if you are using the SYNCBUSY.CTRLB, SYNCBUSY.STATUS, SYNCBUSY.COUNT, SYNCBUSY.PATT, SYNCBUSY.WAVE, SYNCBUSY.PER or SYNCBUSY.CCx bits