Welcome to 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®-M4F
Core Size	32-Bit Single-Core
Speed	120MHz
Connectivity	EBI/EMI, I²C, IrDA, LINbus, MMC/SD, QSPI, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, I²S, POR, PWM
Number of I/O	51
Program Memory Size	1MB (1M x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256K x 8
Voltage - Supply (Vcc/Vdd)	1.71V ~ 3.63V
Data Converters	A/D 24x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-VFQFN Exposed Pad
Supplier Device Package	64-VQFN (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsamd51j20a-mu

Related Links

1. Configuration Summary

21.8.11 Compare n Value in COUNT32 mode (CTRLA.MODE=0)

Name: COMP
Offset: 0x20 + n*0x04 [n=0..1]
Reset: 0x00000000
Property: PAC Write-Protection, Write-Synchronized

Bit	31	30	29	28	27	26	25	24
	COMP[31:24]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

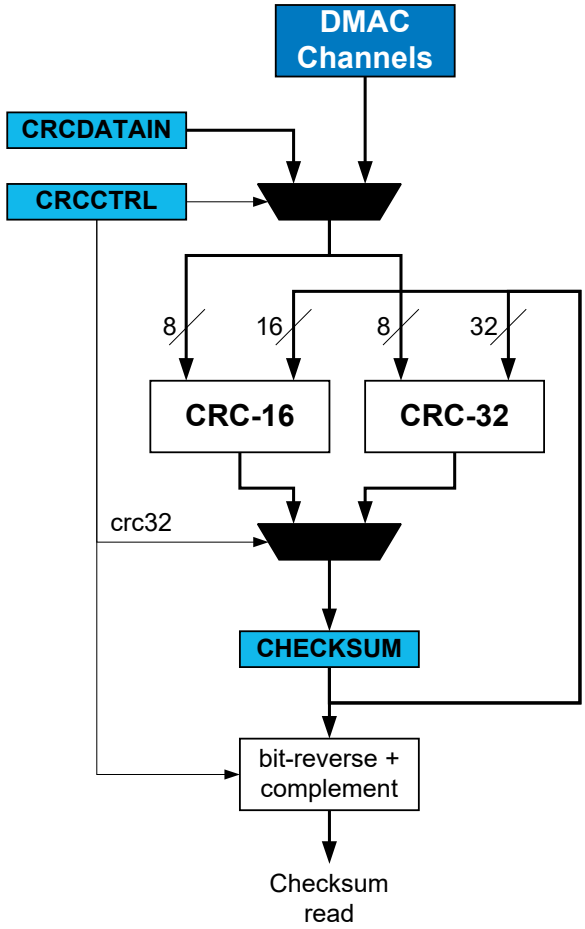
Bit	23	22	21	20	19	18	17	16
	COMP[23:16]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	COMP[15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	COMP[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – COMP[31:0] Compare Value
The 32-bit value of COMPn is continuously compared with the 32-bit COUNT value. When a match occurs, the Compare n interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.CMPn) is set on the next counter cycle, and the counter value is cleared if CTRLA.MATCHCLR is one.

Figure 22-19. CRC Generator Block Diagram



CRC on DMA data CRC-16 or CRC-32 calculations can be performed on data passing through any DMA channel. Once a DMA channel is selected as the source, the CRC engine will continuously generate the CRC on the data passing through the DMA channel. The checksum is available for readout once the DMA transaction is completed or aborted. A CRC can also be generated on SRAM, Flash, or I/O memory by passing these data through a DMA channel. If the latter is done, the destination register for the DMA data can be the data input (CRCDATAIN) register in the CRC engine.

CRC using the I/O interface Before using the CRC engine with the I/O interface, the application must set the CRC Beat Size bits in the CRC Control register (CRCCTRL.CRCBEATSIZE). 8/16/32-bit bus transfer type can be selected.

CRC can be performed on any data by loading them into the CRC engine using the CPU and writing the data to the CRCDATAIN register. Using this method, an arbitrary number of bytes can be written to the register by the CPU, and CRC is done continuously for each byte. This means if a 32-bit data is written to the CRCDATAIN register the CRC engine takes four cycles to calculate the CRC. The CRC complete is signaled by a set CRCBUSY bit in the CRCSTATUS register. New data can be written only when CRCBUSY flag is not set.

22.6.3.9 Memory CRC Generation

When enabled, it is possible to automatically calculate a memory block checksum. When the channel is enabled and the descriptor is fetched, the CRC Checksum register (CRCCHKSUM) is reloaded with the

22.7 Register Summary

Offset	Name	Bit Pos.								
0x00	CTRL	7:0							DMAENABLE	SWRST
		15:8					LVLENx3	LVLENx2	LVLENx1	LVLENx0
0x02	CRCCTRL	7:0					CRCPOLY[1:0]		CRCBEATSIZE[1:0]	
		15:8	CRCMODE[1:0]		CRCSRC[5:0]					
0x04	CRCDATAIN	7:0	CRCDATAIN[7:0]							
		15:8	CRCDATAIN[15:8]							
		23:16	CRCDATAIN[23:16]							
		31:24	CRCDATAIN[31:24]							
0x08	CRCCHKSUM	7:0	CRCCHKSUM[7:0]							
		15:8	CRCCHKSUM[15:8]							
		23:16	CRCCHKSUM[23:16]							
		31:24	CRCCHKSUM[31:24]							
0x0C	CRCSTATUS	7:0						CRCERR	CRCZERO	CRCBUSY
0x0D	DBGCTRL	7:0								DBGRUN
0x0E ... 0x0F	Reserved									
0x10	SWTRIGCTRL	7:0	SWTRIGn[7:0]							
		15:8	SWTRIGn[15:8]							
		23:16	SWTRIGn[23:16]							
		31:24	SWTRIGn[31:24]							
0x14	PRICTRL0	7:0	RRLVLEN0	QOS00[1:0]		LVLPRI0[4:0]				
		15:8	RRLVLEN1	QOS01[1:0]		LVLPRI1[4:0]				
		23:16	RRLVLEN2	QOS02[1:0]		LVLPRI2[4:0]				
		31:24	RRLVLEN3	QOS03[1:0]		LVLPRI3[4:0]				
0x18 ... 0x1F	Reserved									
0x20	INTPEND	7:0				ID[4:0]				
		15:8	PEND	BUSY	FERR	CRCERR		SUSP	TCMPL	TERR
0x22 ... 0x23	Reserved									
0x24	INTSTATUS	7:0	CHINTn[7:0]							
		15:8	CHINTn[15:8]							
		23:16	CHINTn[23:16]							
		31:24	CHINTn[31:24]							
0x28	BUSYCH	7:0	BUSYCHn[7:0]							
		15:8	BUSYCHn[15:8]							
		23:16	BUSYCHn[23:16]							
		31:24	BUSYCHn[31:24]							
0x2C	PENDCH	7:0	PENDCH7	PENDCH6	PENDCH5	PENDCH4	PENDCH3	PENDCH2	PENDCH1	PENDCH0
		15:8	PENDCH15	PENDCH14	PENDCH13	PENDCH12	PENDCH11	PENDCH10	PENDCH9	PENDCH8

24.9.35 GMAC PTP Event Frame Received Seconds High Register

Name: EFRSH
Offset: 0x0EC
Reset: 0x00000000
Property: Read-only

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	RUD[15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	RUD[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 15:0 – RUD[15:0] Register Update
The register is updated with the value that the IEEE 1588 timer seconds register held when the SFD of a PTP transmit primary event crosses the MII interface. An interrupt is issued when the register is updated.

26.7 Register Summary - ICM

Offset	Name	Bit Pos.								
0x00	CFG	7:0	BBC[3:0]					SLBDIS	EOMDIS	WBDIS
		15:8	UALGO[2:0]			UIHASH			DUALBUFF	ASCD
		23:16								
		31:24								
0x04	CTRL	7:0	REHASH[3:0]					SWRST	DISABLE	ENABLE
		15:8	RMEN[3:0]				RMDIS[3:0]			
		23:16								
		31:24								
0x08	SR	7:0								ENABLE
		15:8	RMDIS[3:0]				RAWRMDIS[3:0]			
		23:16								
		31:24								
0x0C ... 0x0F	Reserved									
0x10	IER	7:0	RDM[3:0]				RHC[3:0]			
		15:8	RWC[3:0]				RBE[3:0]			
		23:16	RSU[3:0]				REC[3:0]			
		31:24								URAD
0x14	IDR	7:0	RDM[3:0]				RHC[3:0]			
		15:8	RWC[3:0]				RBE[3:0]			
		23:16	RSU[3:0]				REC[3:0]			
		31:24								URAD
0x18	IMR	7:0	RDM[3:0]				RHC[3:0]			
		15:8	RWC[3:0]				RBE[3:0]			
		23:16	RSU[3:0]				REC[3:0]			
		31:24								URAD
0x1C	ISR	7:0	RDM[3:0]				RHC[3:0]			
		15:8	RWC[3:0]				RBE[3:0]			
		23:16	RSU[3:0]				REC[3:0]			
		31:24								URAD
0x20	UASR	7:0						URAT[2:0]		
		15:8								
		23:16								
		31:24								
0x24 ... 0x2F	Reserved									
0x30	DSCR	7:0	DASA[1:0]							
		15:8	DASA[9:2]							
		23:16	DASA[17:10]							
		31:24	DASA[25:18]							
0x34	HASH	7:0	HASA[0:0]							

Value	Description
0	Peripheral is not write protected
1	Peripheral is write protected

Bit 11 – ICM ICM APB Protection Enable

Value	Description
0	Peripheral is not write protected
1	Peripheral is write protected

Bit 10 – TRNG TRNG APB Protection Enable

Value	Description
0	Peripheral is not write protected
1	Peripheral is write protected

Bit 9 – AES AES APB Protection Enable

Value	Description
0	Peripheral is not write protected
1	Peripheral is write protected

Bit 8 – AC AC APB Protection Enable

Value	Description
0	Peripheral is not write protected
1	Peripheral is write protected

Bit 7 – PDEC PDEC APB Protection Enable

Value	Description
0	Peripheral is not write protected
1	Peripheral is write protected

Bit 6 – TC5 TC5 APB Protection Enable

Value	Description
0	Peripheral is not write protected
1	Peripheral is write protected

Bit 5 – TC4 TC4 APB Protection Enable

Value	Description
0	Peripheral is not write protected
1	Peripheral is write protected

Bit 4 – TCC3 TCC3 APB Protection Enable

Value	Description
0	Peripheral is not write protected
1	Peripheral is write protected

The DFLL48M requires a reference clock (GCLK_DFLL48M_REF) from the GCLK. The control logic uses the oscillator output, which is also asynchronous to the user interface clock (CLK_OSCCTRL_APB). Due to this asynchronicity, writes to certain registers will require synchronization between the clock domains. Refer to Synchronization for further details.

The FDPLL200Mn require a reference clock (GCLK_DPLL) for the FDPLL output. When the optional lock timer timeout function is used, a 32KHz reference clock (GCLK_DPLL_32K) is also required. Both reference clocks can either stem from the GCLK and/or from external oscillators.

28.5.4 DMA

Not applicable.

28.5.5 Interrupts

The interrupt request line is connected to the Interrupt Controller. Using the OSCCTRL interrupts requires the interrupt controller to be configured first.

Related Links

10.2 Nested Vector Interrupt Controller

28.5.6 Events

The events of this peripheral are connected to the Event System.

Related Links

31. EVSYS – Event System

28.5.7 Debug Operation

When the CPU is halted in debug mode the OSCCTRL continues normal operation. If the OSCCTRL is configured in a way that requires it to be periodically serviced by the CPU through interrupts or similar, improper operation or data loss may result during debugging.

28.5.8 Register Access Protection

All registers with write-access can be write-protected optionally by the Peripheral Access Controller (PAC), except for the following registers:

- Interrupt Flag Status and Clear register (INTFLAG)

Note: Optional write-protection is indicated by the "PAC Write-Protection" property in the register description.

Write-protection does not apply for accesses through an external debugger.

28.5.9 Analog Connections

The 8-48 MHz crystal must be connected between the XIN and XOUT pins, along with any required load capacitors.

Note: Refer to the Electrical Characteristics for more information about load capacitors.

28.6 Functional Description

28.6.1 Principle of Operation

XOSC, DFLL48M, and DPLL200M are configured via OSCCTRL control registers. Through this interface, the oscillators are enabled, disabled, or have their calibration values updated.

Bit 25 – DPLL1LCKF DPLL1 Lock Fall Interrupt Enable
0: The DPLL1 Lock Fall interrupt is disabled.

1: The DPLL1 Lock Fall interrupt is enabled, and an interrupt request will be generated when the DPLL1 Lock Fall Interrupt flag is set.

Writing a zero to this bit has no effect.

Writing a '1' to this bit will set the DPLL1 Lock Fall Interrupt Enable bit, which enables the DPLL1 Lock Fall interrupt.

Bit 24 – DPLL1LCKR DPLL1 Lock Rise Interrupt Enable
0: The DPLL1 Lock Rise interrupt is disabled.

1: The DPLL1 Lock Rise interrupt is enabled, and an interrupt request will be generated when the DPLL1 Lock Rise Interrupt flag is set.

Writing a zero to this bit has no effect.

Writing a '1' to this bit will set the DPLL1 Lock Rise Interrupt Enable bit, which enables the DPLL1 Lock Rise interrupt.

Bit 19 – DPLL0LDRTO DPLL0 Loop Divider Ratio Update Complete Interrupt Enable
0: The DPLL0 Loop Divider Ratio Update Complete interrupt is disabled.

1: The DPLL0 Loop Divider Ratio Update Complete interrupt is enabled, and an interrupt request will be generated when the DPLL0 Loop Divider Ratio Update Complete Interrupt flag is set.

Writing a zero to this bit has no effect.

Writing a '1' to this bit will set the DPLL0 Loop Divider Ratio Update Complete Interrupt Enable bit, which enables the DPLL0 Loop Divider Ratio Update Complete interrupt.

Bit 18 – DPLL0LTO DPLL0 Lock Timeout Interrupt Enable
0: The DPLL0 Lock Timeout interrupt is disabled.

1: The DPLL0 Lock Timeout interrupt is enabled, and an interrupt request will be generated when the DPLL0 Lock Timeout Interrupt flag is set.

Writing a zero to this bit has no effect.

Writing a '1' to this bit will set the DPLL0 Lock Timeout Interrupt Enable bit, which enables the DPLL0 Lock Timeout interrupt.

Bit 17 – DPLL0LCKF DPLL0 Lock Fall Interrupt Enable
0: The DPLL0 Lock Fall interrupt is disabled.

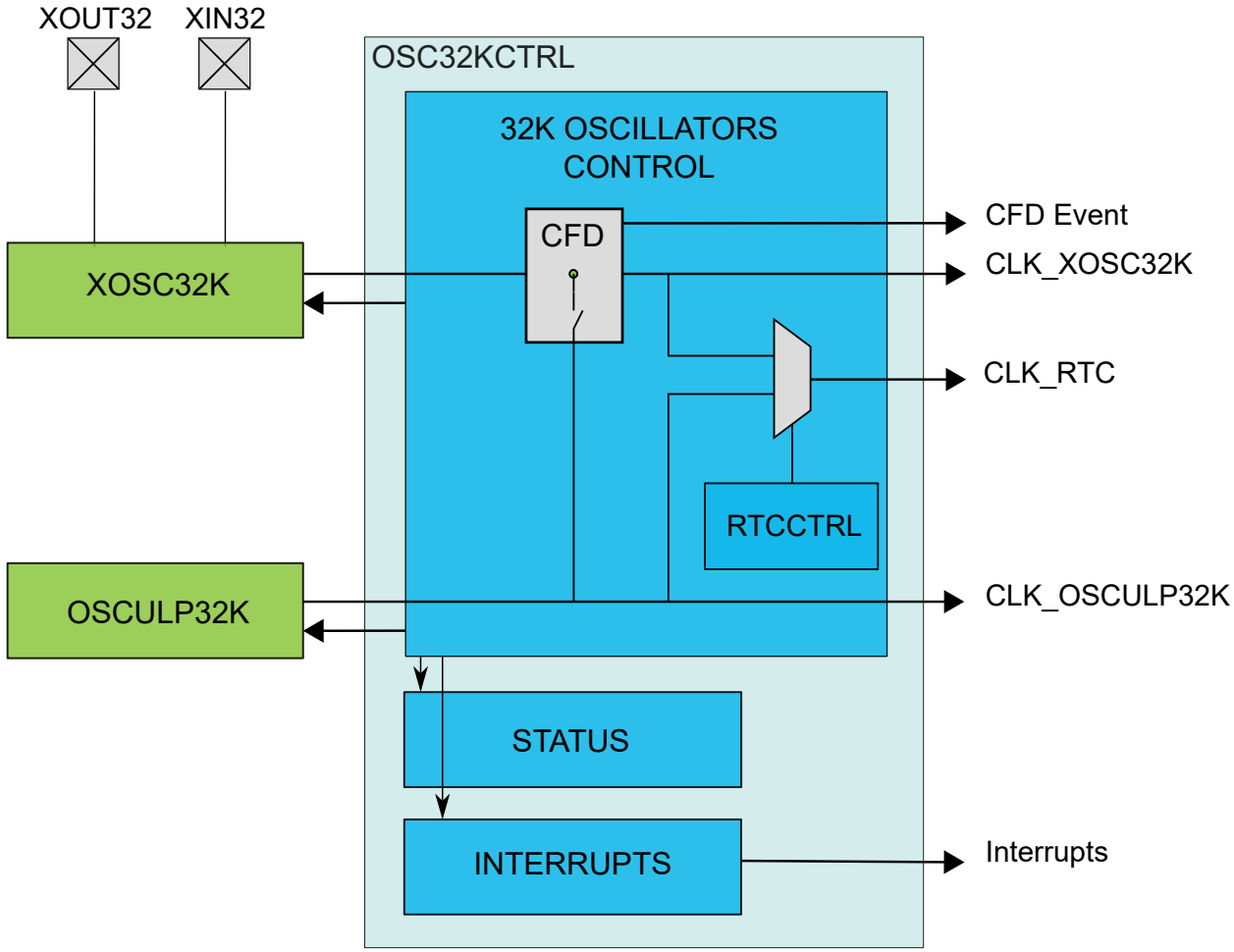
1: The DPLL0 Lock Fall interrupt is enabled, and an interrupt request will be generated when the DPLL0 Lock Fall Interrupt flag is set.

Writing a zero to this bit has no effect.

Writing a '1' to this bit will set the DPLL0 Lock Fall Interrupt Enable bit, which enables the DPLL0 Lock Fall interrupt.

Bit 16 – DPLL0LCKR DPLL0 Lock Rise Interrupt Enable
0: The DPLL0 Lock Rise interrupt is disabled.

29.3 Block Diagram



29.4 Signal Description

Signal	Description	Type
XIN32	Analog Input	32.768 kHz Crystal Oscillator or external clock input
XOUT32	Analog Output	32.768 kHz Crystal Oscillator output

The I/O lines are automatically selected when XOSC32K is enabled.

Note: The signal of the external crystal oscillator may affect the jitter of neighboring pads.

29.5 Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

29.5.1 I/O Lines

I/O lines are configured by OSC32KCTRL when XOSC32K is enabled, and need no user configuration.

32.9.9 Data Input Value

Name: IN
Offset: 0x20
Reset: 0x40000000
Property: -

Tip: The I/O pins are assembled in pin groups (”PORT groups”) with up to 32 pins. Group 0 consists of the PA pins, group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

Bit	31	30	29	28	27	26	25	24
	IN[31:24]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	IN[23:16]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	IN[15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	IN[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – IN[31:0] PORT Data Input Value
These bits are cleared when the corresponding I/O pin input sampler detects a logical low level on the input pin.

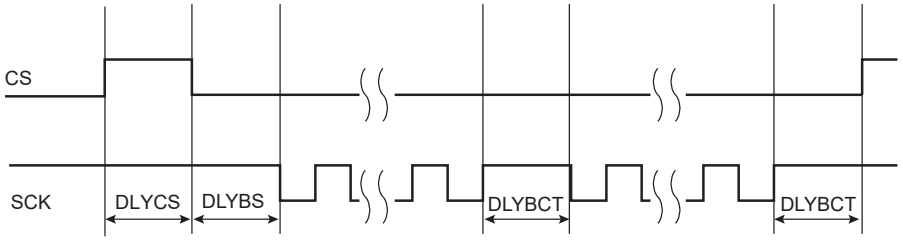
These bits are set when the corresponding I/O pin input sampler detects a logical high level on the input pin.

delay between two consecutive transfers. In Serial Memory mode, this delay is not programmable and DLYBCT settings are ignored.

- The delay before SCK is programmed by writing the Delay Before SCK bit field in the BAUD register (BAUD.DLYBS), allowing to delay the start of SPCK after the chip select has been asserted.

These delays allow the QSPI to be adapted to the interfaced peripherals and their speed and bus release time.

Figure 37-5. Programmable Delay



37.6.7 QSPI SPI Mode

In this mode, the QSPI acts as a regular SPI Master.

To activate this mode, the MODE bit in Control B register must be cleared (CTRLB.MODE=0).

37.6.7.1 SPI Mode Operations

The QSPI in standard SPI mode operates on the clock generated by the internal programmable baud rate generator. It fully controls the data transfers to and from the slave connected to the SPI bus. The QSPI drives the chip select line to the slave (CS) and the serial clock signal (SCK).

The QSPI features a single internal shift register and two holding registers: the Transmit Data Register (TXDATA) and the Receive Data Register (RXDATA). The holding registers maintain the data flow at a constant rate.

After enabling the QSPI, a data transfer begins when the processor writes to the TXDATA. The written data is immediately transferred into the internal shift register and transfer on the SPI bus starts. While the data in the internal shift register is shifted on the MOSI line, the MISO line is sampled and shifted into the internal shift register. Receiving data cannot occur without transmitting data.

If new data is written in TXDATA during the transfer, it stays in TXDATA until the current transfer is completed. Then, the received data is transferred from the internal shift register to the RXDATA, the data in TXDATA is loaded into the internal shift register, and a new transfer starts.

The transfer of data written in TXDATA in the internal shift register is indicated by the Transmit Data Register Empty (DRE) bit in the Interrupt Flag Status and Clear register (INTFLAG.DRE). When new data is written in TXDATA, this bit is cleared. The DRE bit is used to trigger the Transmit DMA channel.

The end of transfer is indicated by the Transmission Complete flag (INTFLAG.TXC). If the transfer delay for the last transfer was configured to be greater than 0 (CTRLB.DLYBCT), TXC is set after the completion of the delay. The module clock (CLK_QSPI_AHB) can be switched off at this time.

Ongoing transfer of received data from the internal shift register into RXDATA is indicated by the Receive Data Register Full flag (INTFLAG.RXC). When the received data is read, the RXC bit is cleared.

If the RXDATA has not been read before new data is received, the Overrun Error flag in INTFLAG register (INTFLAG.ERROR) is set. As long as this flag is set, data is loaded in RXDATA.

The SPI Mode Block Diagram shows a flow chart describing how transfers are handled.

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

Value	Description
0	Interrupt assigned to CAN interrupt line 0.
1	Interrupt assigned to CAN interrupt line 1.

Bit 25 – BOL Bus_Off Status Interrupt Line

Value	Description
0	Interrupt assigned to CAN interrupt line 0.
1	Interrupt assigned to CAN interrupt line 1.

Bit 24 – EWL Error Warning Status Interrupt Line

Value	Description
0	Interrupt assigned to CAN interrupt line 0.
1	Interrupt assigned to CAN interrupt line 1.

Bit 23 – EPL Error Passive Interrupt Line

Value	Description
0	Interrupt assigned to CAN interrupt line 0.
1	Interrupt assigned to CAN interrupt line 1.

Bit 22 – ELOL Error Logging Overflow Interrupt Line

Value	Description
0	Interrupt assigned to CAN interrupt line 0.
1	Interrupt assigned to CAN interrupt line 1.

Bit 21 – BEUL Bit Error Uncorrected Interrupt Line

Value	Description
0	Interrupt assigned to CAN interrupt line 0.
1	Interrupt assigned to CAN interrupt line 1.

Bit 20 – BECL Bit Error Corrected Interrupt Line

Value	Description
0	Interrupt assigned to CAN interrupt line 0.
1	Interrupt assigned to CAN interrupt line 1.

Bit 19 – DRXL Message stored to Dedicated Rx Buffer Interrupt Line

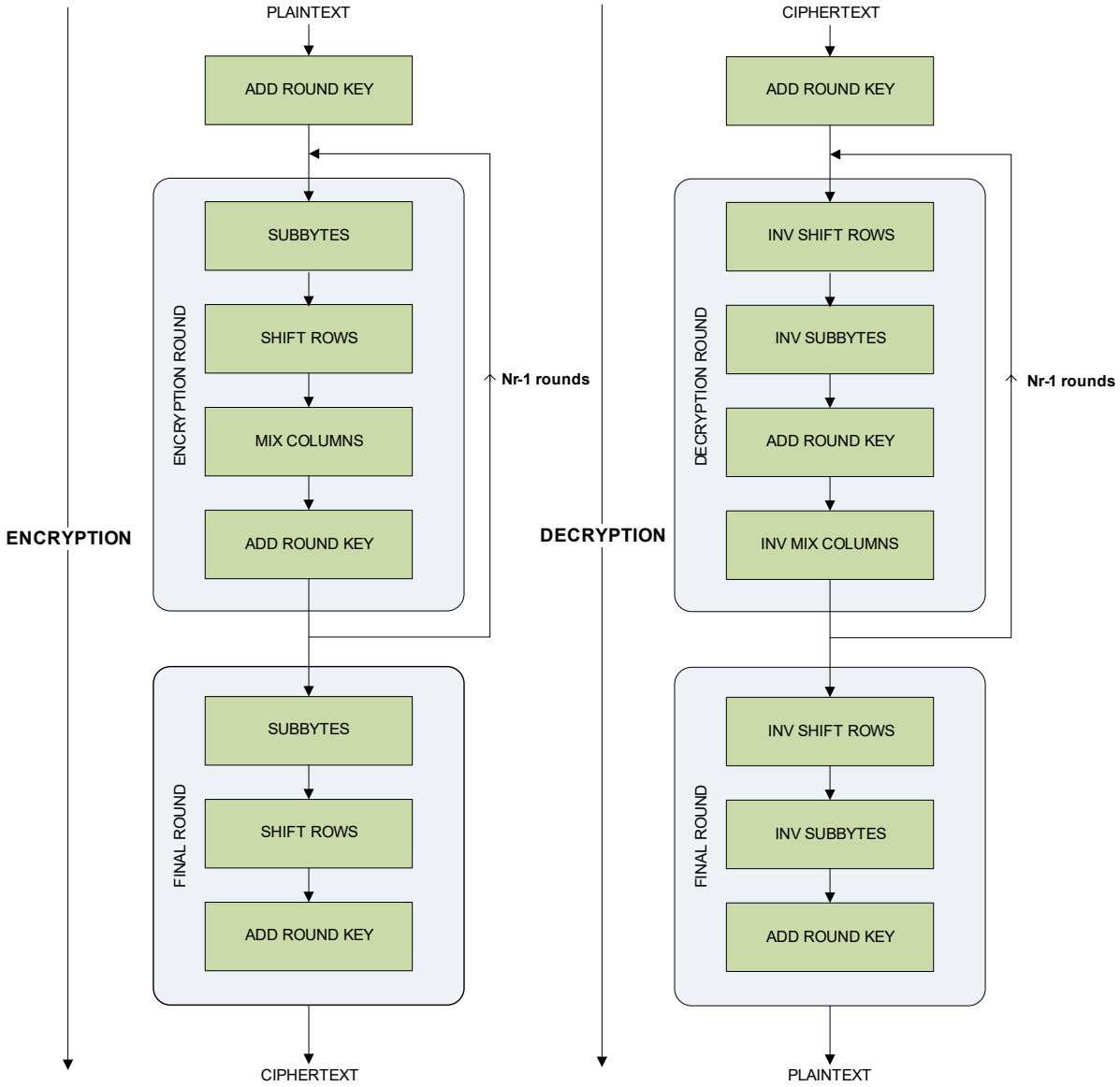
Value	Description
0	Interrupt assigned to CAN interrupt line 0.
1	Interrupt assigned to CAN interrupt line 1.

Bit 18 – TOOL Timeout Occurred Interrupt Line

Value	Description
0	Interrupt assigned to CAN interrupt line 0.
1	Interrupt assigned to CAN interrupt line 1.

42.3 Block Diagram

Figure 42-1. AES Block Diagram

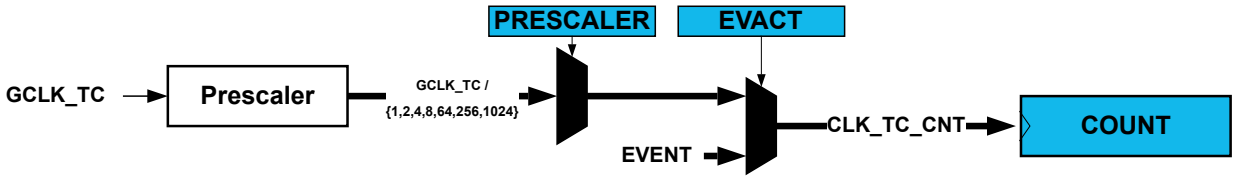


Prescaler outputs from 1 to 1/1024 are available. For a complete list of available prescaler outputs, see the register description for the Prescaler bit group in the Control A register (CTRLA.PRESCALER).

Note: When counting events, the prescaler is bypassed.

The joint stream of prescaler ticks and event action ticks is called CLK_TC_CNT.

Figure 48-2. Prescaler



48.6.2.4 Counter Mode

The counter mode is selected by the Mode bit group in the Control A register (CTRLA.MODE). By default, the counter is enabled in the 16-bit counter resolution. Three counter resolutions are available:

- COUNT8: The 8-bit TC has its own Period Value and Period Buffer Value registers (PER and PERBUF).
- COUNT16: 16-bit is the default counter mode. There is no dedicated period register in this mode.
- COUNT32: This mode is achieved by pairing two 16-bit TC peripherals. TCn is paired with TCn+1. does not support 32-bit resolution.
  When paired, the TC peripherals are configured using the registers of the even-numbered TC. The odd-numbered partner will act as a slave, and the Slave bit in the Status register (STATUS.SLAVE) will be set. The register values of a slave will not reflect the registers of the 32-bit counter. Writing to any of the slave registers will not affect the 32-bit counter. Normal access to the slave COUNT and CCx registers is not allowed.

48.6.2.5 Counter Operations

Depending on the mode of operation, the counter is cleared, reloaded, incremented, or decremented at each TC clock input (CLK_TC_CNT). A counter clear or reload marks the end of the current counter cycle and the start of a new one.

The counting direction is set by the Direction bit in the Control B register (CTRLB.DIR). If this bit is zero the counter is counting up, and counting down if CTRLB.DIR=1. The counter will count up or down for each tick (clock or event) until it reaches TOP or ZERO. When it is counting up and TOP is reached, the counter will be set to zero at the next tick (overflow) and the Overflow Interrupt Flag in the Interrupt Flag Status and Clear register (INTFLAG.OVF) will be set. When it is counting down, the counter is reloaded with the TOP value when ZERO is reached (underflow), and INTFLAG.OVF is set.

INTFLAG.OVF can be used to trigger an interrupt, a DMA request, or an event. An overflow/underflow occurrence (i.e., a compare match with TOP/ZERO) will stop counting if the One-Shot bit in the Control B register is set (CTRLBSET.ONESHOT).

It is possible to change the counter value (by writing directly in the COUNT register) even when the counter is running. When starting the TC, the COUNT value will be either ZERO or TOP (depending on the counting direction set by CTRLBSET.DIR or CTRLBCLR.DIR), unless a different value has been written to it, or the TC has been stopped at a value other than ZERO. The write access has higher priority than count, clear, or reload. The direction of the counter can also be changed when the counter is running. See also the following figure.

Condition	Interrupt request	Event output	Event input	DMA request	DMA request is cleared
Count	Yes	Yes			
Capture Overflow Error	Yes				
Debug Fault State	Yes				
Recoverable Faults	Yes				
Non-Recoverable Faults	Yes				
TCCx Event 0 input			Yes(4)		
TCCx Event 1 input			Yes(5)		

Notes:

1. DMA request set on overflow, underflow or re-trigger conditions.
2. Can perform capture or generate recoverable fault on an event input.
3. In capture or circular modes.
4. On event input, either action can be executed:
   - re-trigger counter
   - control counter direction
   - stop the counter
   - decrement the counter
   - perform period and pulse width capture
   - generate non-recoverable fault
5. On event input, either action can be executed:
   - re-trigger counter
   - increment or decrement counter depending on direction
   - start the counter
   - increment or decrement counter based on direction
   - increment counter regardless of direction
   - generate non-recoverable fault

49.6.5.1 DMA Operation

The TCC can generate the following DMA requests:

Counter overflow (OVF)
If the Ones-shot Trigger mode in the control A register (CTRLA.DMAOS) is written to '0', the TCC generates a DMA request on each cycle when an update condition (overflow, underflow or re-trigger) is detected.
When an update condition (overflow, underflow or re-trigger) is detected while CTRLA.DMAOS=1, the TCC generates a DMA trigger on the cycle following the DMA One-Shot Command written to the Control B register (CTRLBSET.CMD=DMAOS).

In both cases, the request is cleared by hardware on DMA acknowledge.

Channel Match (MCx)
A DMA request is set only on a compare match if CTRLA.DMAOS=0. The request is cleared by hardware on DMA acknowledge.
When CTRLA.DMAOS=1, the DMA requests are not generated.

The following bits are synchronized when written:

- Software Reset and Enable bits in Control A register (CTRLA.SWRST and CTRLA.ENABLE)

The following registers are synchronized when written:

- Control B Clear and Control B Set registers (CTRLBCLR and CTRLBSET)
- Status register (STATUS)
- Pattern and Pattern Buffer registers (PATT and PATTBUF)
- Waveform register (WAVE)
- Count Value register (COUNT)
- Period Value and Period Buffer Value registers (PER and PERBUF)
- Compare/Capture Channel x and Channel x Compare/Capture Buffer Value registers (CCx and CCBUFx)

The following registers are synchronized when read:

- Control B Clear and Control B Set registers (CTRLBCLR and CTRLBSET)
- Count Value register (COUNT): synchronization is done on demand through READSYNC command (CTRLBSET.CMD)
- Pattern and Pattern Buffer registers (PATT and PATTBUF)
- Waveform register (WAVE)
- Period Value and Period Buffer Value registers (PER and PERBUF)
- Compare/Capture Channel x and Channel x Compare/Capture Buffer Value registers (CCx and CCBUFx)

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

Required read-synchronization is denoted by the "Read-Synchronized" property in the register description.

Related Links

13.3 Register Synchronization

Symbol	Parameters	Conditions (see Notes 3, 4)	Min	Typ	Max	Unit
		LEVEL[7:0] = 0xFF (max)	2.946	3.040	3.112	
VBOD+ (see **Note 2**)	BOD33 threshold level Hysteresis ON at power voltage rising	LEVEL[7:0] = 0x00 (min)	1.473	1.520	1.555	
		LEVEL[7:0]= 0x19 (recommended value)	1.618	1.669	1.707	
		LEVEL[7:0] = 0x1C (fuse value)	1.636	1.687	1.725	
		LEVEL[7:0] = 0xFF (max)	2.953	3.041	3.116	
Level_Step	DC threshold step	-	-	6.00	-	mV
Tstart	Startup time (see **Note 6**)	Time from enable to RDY	-	27	-	µs

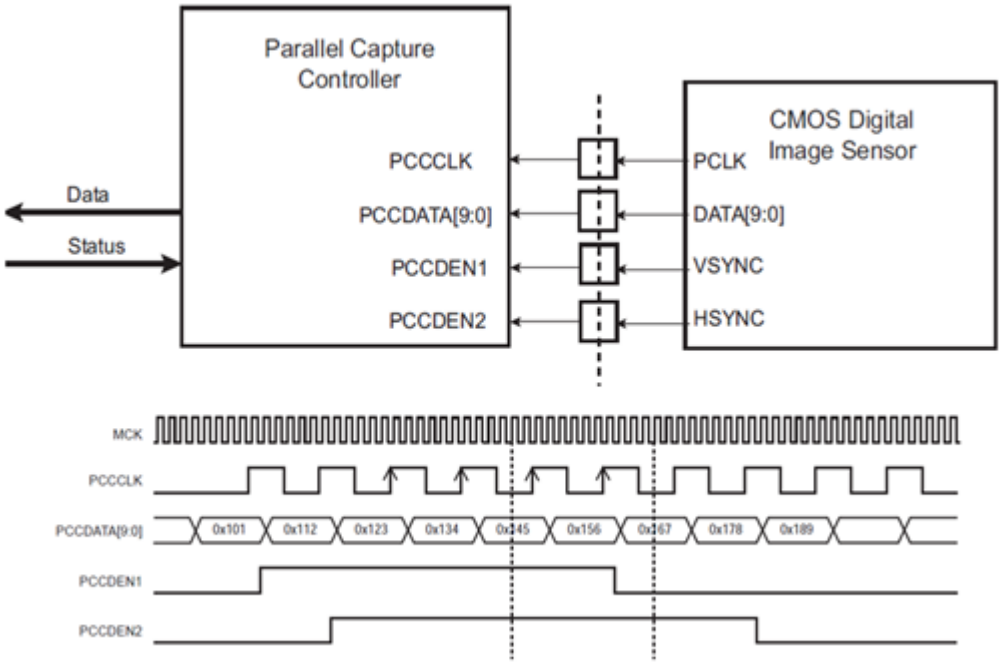
Note:

1. VBOD = VBOD- = 1.5 + LEVEL[7:0) * Level_Step LEVEL[7:0] is calibration setting bus of threshold level.
2. VBOD+ = VBOD- + N * HYST_STEP N = 0 to 15 according to HYST[3:0] value HYST_STEP = Level_Step.
3. Hysteresis OFF mode, HYST[3:0] = 0x0.
4. Hysteresis ON mode, HYST[3:0] = 0x1 to 0xf; Min/Typ/Max values given for 0x2.
5. At the upper side of LEVEL[7:0] values depending on the Hysteresis value chosen with HYST[3:0], the VBOD+ level reaches an overflow, e.g., for HYST[3:0] = 0d2 the hysteresis is 2 x Level_Step = 12 mV up to position 253 and position 254 to 255 above must not be used.
6. These are based on design simulation. They are not covered by production test limits or characterization.

Table 54-22. BOD33 Characteristics on V_{DD} and V_{BAT} Monitoring in Low-Power Mode (During Standby/Backup/Hibernate Modes)

Symbol	Parameters	Conditions (see Notes 3, 4)	Min	Typ	Max	Unit
VBOD or VBOD- (see **Note 1**)	BOD33 threshold level Hysteresis OFF or BOD33 threshold level Hysteresis ON	LEVEL[7:0] = 0x00 (min)	1.413	1.510	1.599	V
		LEVEL[7:0]= 0x19 (recommended value)	1.551	1.659	1.760	
		LEVEL[7:0] = 0x1C (fuse value)	1.569	1.677	1.778	
		LEVEL[7:0] = 0xFF (max)	2.845	3.045	3.229	

Figure 54-23. PCC Signaling



54.14 USB Characteristics

The USB on-chip buffers comply with the Universal Serial Bus (USB) v2.0 standard. All AC parameters related to these buffers can be found within the USB 2.0 electrical specifications.

The USB interface is USB-IF certified:

- TID 40001782 - Peripheral Silicon > Low/Full Speed > Silicon Building Blocks
- TID 120000724 - Embedded Hosts > Full Speed

Electrical configuration required to be USB-compliant:

- the CPU frequency must be higher than 16 MHz when USB is active (No constraint for USB suspend mode)
- the operating voltages must be 3.3V (Min. 3.0V, Max. 3.6V).
- the GCLK_USB frequency accuracy source must be less than:
  - in USB device mode, 48MHz +/-0.25%
  - in USB host mode, 48MHz +/-0.05%

Table 54-60. GCLK_USB Clock Setup Recommendations

Clock setup		USB Device	USB Host
DFLL48M	Open loop	No	No
	Close loop, Ref. internal OSC source	No	No
	Close loop, Ref. external XOSC source	**Yes**	No
	Close loop, Ref. SOF (USB recovery mode)(1)	**Yes**(2)	N/A