

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

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

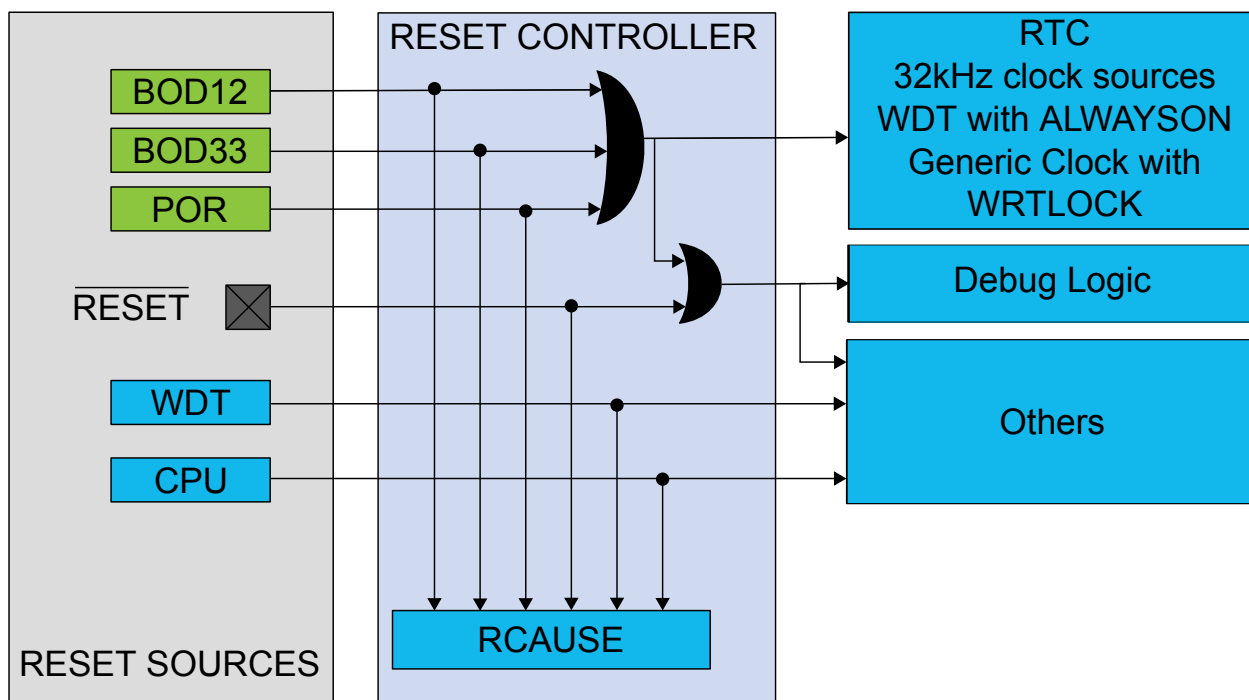
"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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	52
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	32K x 8
Voltage - Supply (Vcc/Vdd)	1.62V ~ 3.6V
Data Converters	A/D 20x12b; D/A 1x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-UFBGA
Supplier Device Package	64-UFBGA (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsamd21j18a-cut

Figure 16-3. Reset Controller



16.6.2.8 Sleep Mode Controller

Sleep mode is activated by the Wait For Interrupt instruction (WFI). The Idle bits in the Sleep Mode register (SLEEP.IDLE) and the SLEEPDEEP bit of the System Control register of the CPU should be used as argument to select the level of the sleep mode.

There are two main types of sleep mode:

- **IDLE mode:** The CPU is stopped. Optionally, some synchronous clock domains are stopped, depending on the IDLE argument. Regulator operates in normal mode.
- **STANDBY mode:** All clock sources are stopped, except those where the RUNSTDBY bit is set. Regulator operates in low-power mode. Before entering standby mode the user must make sure that a significant amount of clocks and peripherals are disabled, so that the voltage regulator is not overloaded.

Table 16-3. Sleep Mode Entry and Exit Table

Mode	Level	Mode Entry	Wake-Up Sources
IDLE	0	SCR.SLEEPDEEP = 0	Synchronous ⁽²⁾ (APB, AHB), asynchronous ⁽¹⁾
	1	SLEEP.IDLE=Level	Synchronous (APB), asynchronous
	2	WFI	Asynchronous
STANDBY		SCR.SLEEPDEEP = 1 WFI	Asynchronous

Note:

1. Asynchronous: interrupt generated on generic clock or external clock or external event.
2. Synchronous: interrupt generated on the APB clock.

17.5.1 I/O Lines

I/O lines are configured by SYSCTRL when either XOSC or XOSC32K are enabled, and need no user configuration.

17.5.2 Power Management

The SYSCTRL can continue to operate in any sleep mode where the selected source clock is running. The SYSCTRL interrupts can be used to wake up the device from sleep modes. The events can trigger other operations in the system without exiting sleep modes. Refer to *PM – Power Manager* on the different sleep modes.

Related Links

[PM – Power Manager](#)

17.5.3 Clocks

The SYSCTRL gathers controls for all device oscillators and provides clock sources to the Generic Clock Controller (GCLK). The available clock sources are: XOSC, XOSC32K, OSC32K, OSCULP32K, OSC8M, DFLL48M and FDPLL96M.

The SYSCTRL bus clock (CLK_SYSCTRL_APB) can be enabled and disabled in the Power Manager, and the default state of CLK_SYSCTRL_APB can be found in the Peripheral Clock Masking section in the PM – Power Manager.

The clock used by BOD33 and BOD12 in sampled mode is asynchronous to the user interface clock (CLK_SYSCTRL_APB). Likewise, the DFLL48M control logic uses the DFLL oscillator output, which is also asynchronous to the user interface clock (CLK_SYSCTRL_APB). Due to this asynchronicity, writes to certain registers will require synchronization between the clock domains. Refer to [Synchronization](#) for further details.

Related Links

[Peripheral Clock Masking](#)

17.5.4 Interrupts

The interrupt request line is connected to the Interrupt Controller. Using the SYSCTRL interrupts requires the Interrupt Controller to be configured first. Refer to *Nested Vector Interrupt Controller* for details.

Related Links

[Nested Vector Interrupt Controller](#)

17.5.5 Debug Operation

When the CPU is halted in debug mode, the SYSCTRL continues normal operation. If the SYSCTRL 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.

If debugger cold-plugging is detected by the system, BOD33 and BOD12 resets will be masked. The BOD resets keep running under hot-plugging. This allows to correct a BOD33 user level too high for the available supply.

17.5.6 Register Access Protection

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

- Interrupt Flag Status and Clear register (INTFLAG)

Writing a one to this bit will clear the DFLL Reference Clock Stopped Interrupt Enable bit, which disables the DFLL Reference Clock Stopped interrupt.

Value	Description
0	The DFLL Reference Clock Stopped interrupt is disabled.
1	The DFLL Reference Clock Stopped interrupt is enabled, and an interrupt request will be generated when the DFLL Reference Clock Stopped Interrupt flag is set.

Bit 7 – DFLLCKC: DFLL Lock Coarse Interrupt Enable

Writing a zero to this bit has no effect.

Writing a one to this bit will clear the DFLL Lock Coarse Interrupt Enable bit, which disables the DFLL Lock Coarse interrupt.

Value	Description
0	The DFLL Lock Coarse interrupt is disabled.
1	The DFLL Lock Coarse interrupt is enabled, and an interrupt request will be generated when the DFLL Lock Coarse Interrupt flag is set.

Bit 6 – DFLLCKF: DFLL Lock Fine Interrupt Enable

Writing a zero to this bit has no effect.

Writing a one to this bit will clear the DFLL Lock Fine Interrupt Enable bit, which disables the DFLL Lock Fine interrupt.

Value	Description
0	The DFLL Lock Fine interrupt is disabled.
1	The DFLL Lock Fine interrupt is enabled, and an interrupt request will be generated when the DFLL Lock Fine Interrupt flag is set.

Bit 5 – DFLL0OB: DFLL Out Of Bounds Interrupt Enable

Writing a zero to this bit has no effect.

Writing a one to this bit will clear the DFLL Out Of Bounds Interrupt Enable bit, which disables the DFLL Out Of Bounds interrupt.

Value	Description
0	The DFLL Out Of Bounds interrupt is disabled.
1	The DFLL Out Of Bounds interrupt is enabled, and an interrupt request will be generated when the DFLL Out Of Bounds Interrupt flag is set.

Bit 4 – DFLLRDY: DFLL Ready Interrupt Enable

Writing a zero to this bit has no effect.

Writing a one to this bit will clear the DFLL Ready Interrupt Enable bit, which disables the DFLL Ready interrupt.

Value	Description
0	The DFLL Ready interrupt is disabled.
1	The DFLL Ready interrupt is enabled, and an interrupt request will be generated when the DFLL Ready Interrupt flag is set.

Bit 3 – OSC8MRDY: OSC8M Ready Interrupt Enable

Writing a zero to this bit has no effect.

Table 17-7. Start-Up Time for 32kHz Internal Oscillator

STARTUP[2:0]	Number of OSC32K clock cycles	Approximate Equivalent Time (OSCULP= 32 kHz) ⁽¹⁾⁽²⁾⁽³⁾
0x0	3	92µs
0x1	4	122µs
0x2	6	183µs
0x3	10	305µs
0x4	18	549µs
0x5	34	1038µs
0x6	66	2014µs
0x7	130	3967µs

Notes: 1. Number of cycles for the start-up counter.

2. Number of cycles for the synchronization delay, before PCLKSR.OSC32KRDY is set.

3. Start-up time is n OSC32K cycles + 2 OSC32K cycles.

Bit 7 – ONDEMAND: On Demand Control

The On Demand operation mode allows an oscillator to be enabled or disabled depending on peripheral clock requests.

In On Demand operation mode, i.e., if the ONDEMAND bit has been previously written to one, the oscillator will only be running when requested by a peripheral. If there is no peripheral requesting the oscillator's clock source, the oscillator will be in a disabled state.

If On Demand is disabled the oscillator will always be running when enabled.

In standby sleep mode, the On Demand operation is still active if the OSC32K.RUNSTDBY bit is one. If OSC32K.RUNSTDBY is zero, the oscillator is disabled.

Value	Description
0	The oscillator is always on, if enabled.
1	The oscillator is enabled when a peripheral is requesting the oscillator to be used as a clock source. The oscillator is disabled if no peripheral is requesting the clock source.

Bit 6 – RUNSTDBY: Run in Standby

This bit controls how the OSC32K behaves during standby sleep mode:

Value	Description
0	The oscillator is disabled in standby sleep mode.
1	The oscillator is not stopped in standby sleep mode. If OSC32K.ONDEMAND is one, the clock source will be running when a peripheral is requesting the clock. If OSC32K.ONDEMAND is zero, the clock source will always be running in standby sleep mode.

Bit 2 – EN32K: 32kHz Output Enable

19.3 Block Diagram

Figure 19-1. RTC Block Diagram (Mode 0 — 32-Bit Counter)

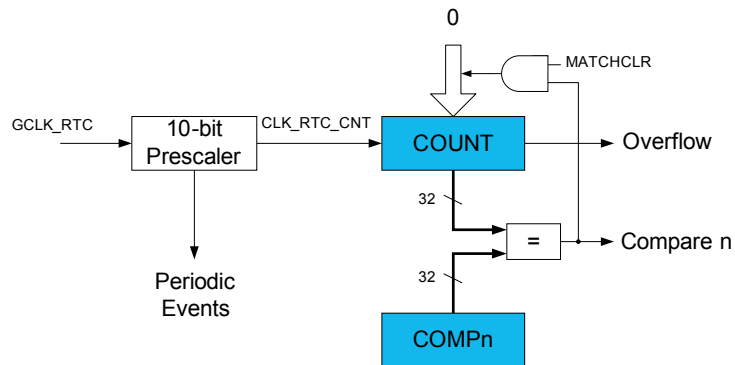


Figure 19-2. RTC Block Diagram (Mode 1 — 16-Bit Counter)

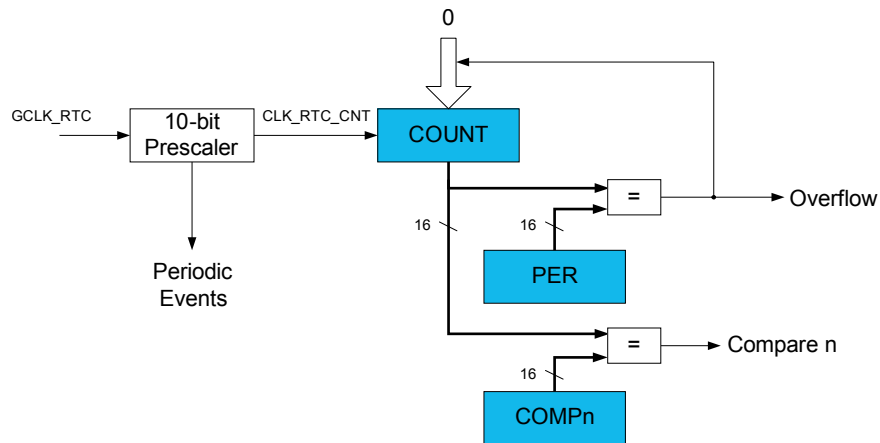
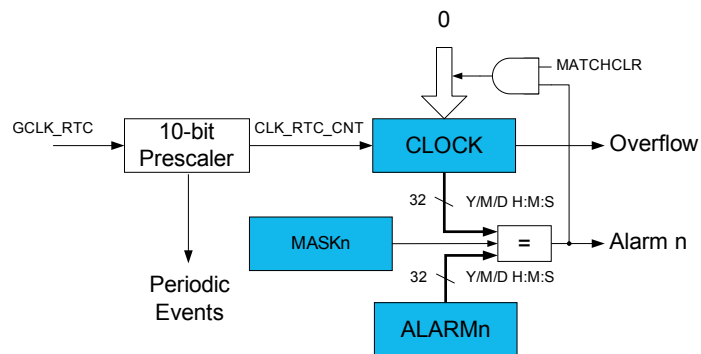


Figure 19-3. RTC Block Diagram (Mode 2 — Clock/Calendar)



19.4 Signal Description

Not applicable.

19.5 Product Dependencies

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

19.5.1 I/O Lines

Not applicable.

20.8.23 Channel Status

This register affects the DMA channel that is selected in the Channel ID register (CHID.ID).

Name: CHSTATUS

Offset: 0x4F

Reset: 0x00

Property: -

Bit	7	6	5	4	3	2	1	0
						FERR	BUSY	PEND
Access						R	R	R
Reset						0	0	0

Bit 2 – FERR: Channel Fetch Error

This bit is cleared when a software resume command is executed.

This bit is set when an invalid descriptor is fetched.

Bit 1 – BUSY: Channel Busy

This bit is cleared when the channel trigger action is completed, when a bus error is detected or when the channel is disabled.

This bit is set when the DMA channel starts a DMA transfer.

Bit 0 – PEND: Channel Pending

This bit is cleared when the channel trigger action is started, when a bus error is detected or when the channel is disabled. For details on trigger action settings, refer to CHCTRLB.TRIGACT.

This bit is set when a transfer is pending on the DMA channel, as soon as the transfer request is received.

32-bit ARM-Based Microcontrollers

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.

22.8.1 Control A

Name: CTRLA

Offset: 0x00

Reset: 0x0000

Property: PAC Write-Protection

Bit	15	14	13	12	11	10	9	8
	CMDEX[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
Bit	7	6	5	4	3	2	1	0
	CMD[6:0]							
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 15:8 – CMDEX[7:0]: Command Execution

When this bit group is written to the key value 0xA5, the command written to CMD will be executed. If a value different from the key value is tried, the write will not be performed and the Programming Error bit in the Status register (STATUS.PROGE) will be set. PROGE is also set if a previously written command is not completed yet.

The key value must be written at the same time as CMD. If a command is issued through the APB bus on the same cycle as an AHB bus access, the AHB bus access will be given priority. The command will then be executed when the NVM block and the AHB bus are idle.

INTFLAG.READY must be '1' when the command is issued.

Bit 0 of the CMDEX bit group will read back as '1' until the command is issued.

Note: The NVM Address bit field in the Address register (ADDR.ADDR) uses 16-bit addressing.

Bits 6:0 – CMD[6:0]: Command

These bits define the command to be executed when the CMDEX key is written.

CMD[6:0]	Group Configuration	Description
0x00-0x01	-	Reserved
0x02	ER	Erase Row - Erases the row addressed by the ADDR register in the NVM main array.
0x03	-	Reserved
0x04	WP	Write Page - Writes the contents of the page buffer to the page addressed by the ADDR register.

32-bit ARM-Based Microcontrollers

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.

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" 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.

26.8.1 Control A

Name: CTRLA

Offset: 0x00

Reset: 0x00000000

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

Bit	31	30	29	28	27	26	25	24
		DORD	CPOL	CMODE	FORM[3:0]			
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0
Bit	23	22	21	20	19	18	17	16
	SAMPA[1:0]		RXPO[1:0]				TXPO[1:0]	
Access	R/W	R/W	R/W	R/W			R/W	R/W
Reset	0	0	0	0			0	0
Bit	15	14	13	12	11	10	9	8
	SAMPR[2:0]							IBON
Access	R/W	R/W	R/W					R
Reset	0	0	0					0
Bit	7	6	5	4	3	2	1	0
	RUNSTDBY			MODE[2:0]			ENABLE	SWRST
Access	R/W			R/W	R/W	R/W	R/W	R/W
Reset	0			0	0	0	0	0

Bit 30 – DORD: Data Order

This bit selects the data order when a character is shifted out from the Data register.

This bit is not synchronized.

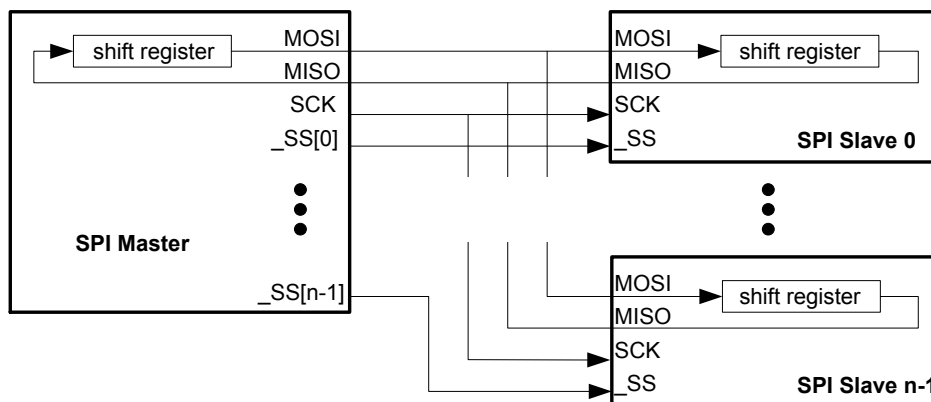
Value	Description
0	MSB is transmitted first.
1	LSB is transmitted first.

Bit 29 – CPOL: Clock Polarity

This bit selects the relationship between data output change and data input sampling in synchronous mode.

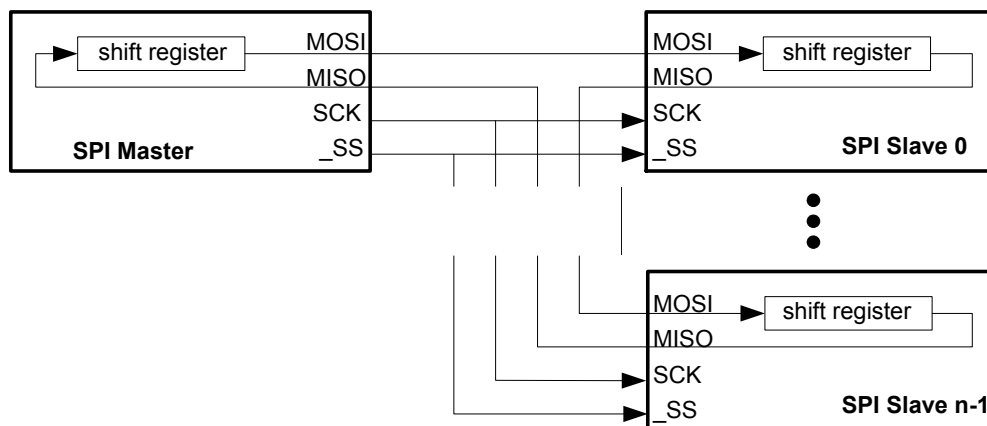
This bit is not synchronized.

Figure 27-5. Multiple Slaves in Parallel



Another configuration is multiple slaves in series, as in [Multiple Slaves in Series](#). In this configuration, all n attached slaves are connected in series. A common \overline{SS} line is provided to all slaves, enabling them simultaneously. The master must shift n characters for a complete transaction. Depending on the Master Slave Select Enable bit (CTRLB.MSEN), the \overline{SS} line can be controlled either by hardware or user software and normal GPIO.

Figure 27-6. Multiple Slaves in Series



27.6.3.4 Loop-Back Mode

For loop-back mode, configure the Data In Pinout (CTRLA.DIPO) and Data Out Pinout (CTRLA.DOPO) to use the same data pins for transmit and receive. The loop-back is through the pad, so the signal is also available externally.

27.6.3.5 Hardware Controlled \overline{SS}

In master mode, a single \overline{SS} chip select can be controlled by hardware by writing the Master Slave Select Enable (CTRLB.MSEN) bit to '1'. In this mode, the \overline{SS} pin is driven low for a minimum of one baud cycle before transmission begins, and stays low for a minimum of one baud cycle after transmission completes. If back-to-back frames are transmitted, the \overline{SS} pin will always be driven high for a minimum of one baud cycle between frames.

In [Hardware Controlled \$\overline{SS}\$](#) , the time T is between one and two baud cycles depending on the SPI transfer mode.

Bit 3 – SSL: Slave Select Low Interrupt Enable

Writing '0' to this bit has no effect.

Writing '1' to this bit will clear the Slave Select Low Interrupt Enable bit, which disables the Slave Select Low interrupt.

Value	Description
0	Slave Select Low interrupt is disabled.
1	Slave Select Low interrupt is enabled.

Bit 2 – RXC: Receive Complete Interrupt Enable

Writing '0' to this bit has no effect.

Writing '1' to this bit will clear the Receive Complete Interrupt Enable bit, which disables the Receive Complete interrupt.

Value	Description
0	Receive Complete interrupt is disabled.
1	Receive Complete interrupt is enabled.

Bit 1 – TXC: Transmit Complete Interrupt Enable

Writing '0' to this bit has no effect.

Writing '1' to this bit will clear the Transmit Complete Interrupt Enable bit, which disable the Transmit Complete interrupt.

Value	Description
0	Transmit Complete interrupt is disabled.
1	Transmit Complete interrupt is enabled.

Bit 0 – DRE: Data Register Empty Interrupt Enable

Writing '0' to this bit has no effect.

Writing '1' to this bit will clear the Data Register Empty Interrupt Enable bit, which disables the Data Register Empty interrupt.

Value	Description
0	Data Register Empty interrupt is disabled.
1	Data Register Empty interrupt is enabled.

27.8.5 Interrupt Enable Set

Name: INTENSET

Offset: 0x16

Reset: 0x00

Property: PAC Write-Protection

Bit	7	6	5	4	3	2	1	0
	ERROR				SSL	RXC	TXC	DRE
Access	R/W				R/W	R/W	R/W	R/W
Reset	0				0	0	0	0

32-bit ARM-Based Microcontrollers

Bit	15	14	13	12	11	10	9	8
	COUNT[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
	COUNT[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 – COUNT[31:0]: Counter Value

These bits contain the current counter value.

30.8.13 Period Value

30.8.13.1 Period Value, 8-bit Mode

Name: PER
Offset: 0x14
Reset: 0xFF
Property: Write-Synchronized

Bit	7	6	5	4	3	2	1	0
	PER[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	1

Bits 7:0 – PER[7:0]: Period Value

These bits hold the value of the Period Buffer register PERBUF. The value is copied to PER register on UPDATE condition.

30.8.14 Compare/Capture

30.8.14.1 Channel x Compare/Capture Value, 8-bit Mode

Name: CCx
Offset: 0x18+i*0x1 [i=0..1]
Reset: 0x00
Property: Write-Synchronized

Bit	7	6	5	4	3	2	1	0
	CC[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 7:0 – CC[7:0]: Channel x Compare/Capture Value

These bits contain the compare/capture value in 8-bit TC mode. In Match frequency (MFRQ) or Match PWM (MPWM) waveform operation (CTRLA.WAVEGEN), the CC0 register is used as a period register.

30.8.14.2 Channel x Compare/Capture Value, 16-bit Mode

Pin Name	Type	Description
...
TCCx/WO[WO_NUM-1]	Digital output	Compare channel n waveform output

Refer to *I/O Multiplexing and Considerations* for details on the pin mapping for this peripheral. One signal can be mapped on several pins.

Related Links

[I/O Multiplexing and Considerations](#)

31.5 Product Dependencies

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

31.5.1 I/O Lines

In order to use the I/O lines of this peripheral, the I/O pins must be configured using the I/O Pin Controller (PORT).

Related Links

[PORT: IO Pin Controller](#)

31.5.2 Power Management

This peripheral can continue to operate in any sleep mode where its source clock is running. The interrupts can wake up the device from sleep modes. Events connected to the event system can trigger other operations in the system without exiting sleep modes.

31.5.3 Clocks

The TCC bus clock (CLK_TCCx_APB, with x instance number of the TCCx) is enabled by default, and can be enabled and disabled in the Power Manager.

A generic clock (GCLK_TCCx) is required to clock the TCC. This clock must be configured and enabled in the generic clock controller before using the TCC. Note that TCC0 and TCC1 share a peripheral clock generator.

The generic clocks (GCLK_TCCx) are asynchronous to the bus clock (CLK_TCCx_APB). Due to this asynchronicity, writing certain registers will require synchronization between the clock domains. Refer to [Synchronization](#) for further details.

Related Links

[GCLK - Generic Clock Controller](#)

[Peripheral Clock Masking](#)

31.5.4 DMA

The DMA request lines are connected to the DMA Controller (DMAC). In order to use DMA requests with this peripheral the DMAC must be configured first. Refer to *DMAC – Direct Memory Access Controller* for details.

Related Links

[DMAC – Direct Memory Access Controller](#)

Bit 11 – DFS: Non-Recoverable Debug Fault Interrupt Enable

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

Writing a '1' to this bit will clear the Debug Fault State Interrupt Disable/Enable bit, which disables the Debug Fault State interrupt.

Value	Description
0	The Debug Fault State interrupt is disabled.
1	The Debug Fault State interrupt is enabled.

Bit 3 – ERR: Error Interrupt Enable

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

Writing a '1' to this bit will clear the Error Interrupt Disable/Enable bit, which disables the Compare interrupt.

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

Bit 2 – CNT: Counter Interrupt Enable

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

Writing a '1' to this bit will clear the Counter Interrupt Disable/Enable bit, which disables the Counter interrupt.

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

Bit 1 – TRG: Retrigger Interrupt Enable

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

Writing a '1' to this bit will clear the Retrigger Interrupt Disable/Enable bit, which disables the Retrigger interrupt.

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

Bit 0 – OVF: Overflow Interrupt Enable

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

Writing a '1' to this bit will clear the Overflow Interrupt Disable/Enable bit, which disables the Overflow interrupt request.

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

31.8.11 Interrupt Enable Set

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 Interrupt Enable Clear (INTENCLR) register.

Name: INTENSET

31.8.20 Waveform Buffer

Name: WAVEB

Offset: 0x68

Reset: 0x00000000

Property: Write-Synchronized, Read-Synchronized

Bit	31	30	29	28	27	26	25	24
					SWAPB 3	SWAPB 2	SWAPB 1	SWAPB 0
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0
Bit	23	22	21	20	19	18	17	16
					POLB3	POLB2	POLB1	POLB0
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0
Bit	15	14	13	12	11	10	9	8
					CICCENB3	CICCENB2	CICCENB1	CICCENB0
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0
Bit	7	6	5	4	3	2	1	0
	CIPERENB		RAMPB[1:0]			WAVEGENB[2:0]		
Access	R/W		R/W	R/W		R/W	R/W	R/W
Reset	0		0	0		0	0	0

Bits 24, 25, 26, 27 – SWAPB n: Swap DTI output pair x Buffer

These register bits are the buffer bits for the SWAP register bits. If double buffering is used, valid content in these bits is copied to the corresponding SWAPx bits on an UPDATE condition.

Bits 16, 17, 18, 19 – POLBn: Channel Polarity x Buffer

These register bits are the buffer bits for POLx register bits. If double buffering is used, valid content in these bits is copied to the corresponding POBx bits on an UPDATE condition.

Bits 8, 9, 10, 11 – CICCENBn: Circular CCx Buffer Enable

These register bits are the buffer bits for CICCENx register bits. If double buffering is used, valid content in these bits is copied to the corresponding CICCENx bits on a UPDATE condition.

Bit 7 – CIPERENB: Circular Period Enable Buffer

This register bit is the buffer bit for CIPEREN register bit. If double buffering is used, valid content in this bit is copied to the corresponding CIPEREN bit on a UPDATE condition.

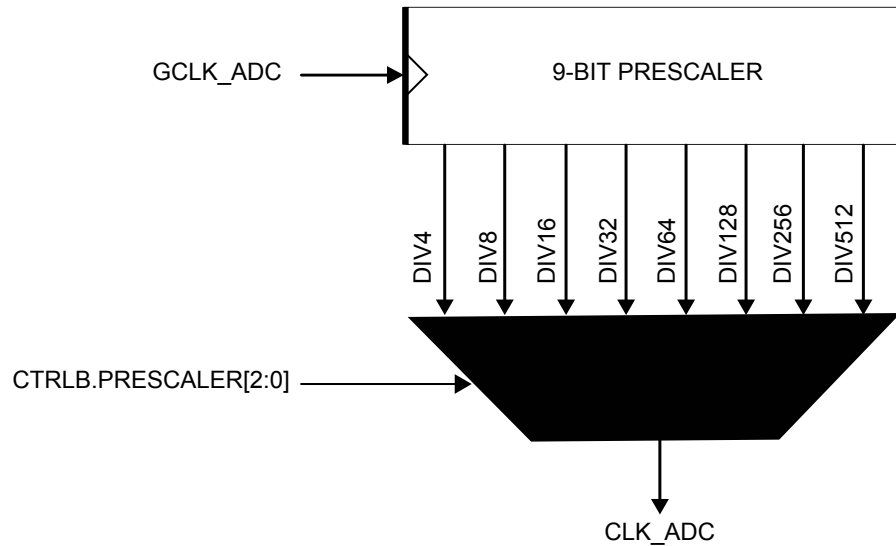
Bits 5:4 – RAMPB[1:0]: Ramp Operation Buffer

These register bits are the buffer bits for RAMP register bits. If double buffering is used, valid content in these bits is copied to the corresponding RAMP bits on a UPDATE condition.

Bits 2:0 – WAVEGENB[2:0]: Waveform Generation Operation Buffer

These register bits are the buffer bits for WAVEGEN register bits. If double buffering is used, valid content in these bits is copied to the corresponding WAVEGEN bits on a UPDATE condition.

Figure 33-2. ADC Prescaler



The propagation delay of an ADC measurement depends on the selected mode and is given by:

- Single-shot mode:

$$\text{PropagationDelay} = \frac{1 + \frac{\text{Resolution}}{2} + \text{DelayGain}}{f_{\text{CLK+ - ADC}}}$$

- Free-running mode:

$$\text{PropagationDelay} = \frac{\frac{\text{Resolution}}{2} + \text{DelayGain}}{f_{\text{CLK+ - ADC}}}$$

Table 33-1. Delay Gain

Name	INTPUTCTRL.GAIN[3:0]	Delay Gain (in CLK_ADC Period)			
		Free-running mode		Single shot mode	
		Differential Mode	Single-Ended Mode	Differential mode	Single-Ended mode
1X	0x0	0	0	0	1
2X	0x1	0	1	0.5	1.5
4X	0x2	1	1	1	2
8X	0x3	1	2	1.5	2.5
16X	0x4	2	2	2	3
Reserved	0x5 ... 0xE	Reserved	Reserved	Reserved	Reserved
DIV2	0xF	0	1	0.5	1.5

33.6.4 ADC Resolution

The ADC supports 8-bit, 10-bit or 12-bit resolution. Resolution can be changed by writing the Resolution bit group in the Control B register (CTRLB.RESSEL). By default, the ADC resolution is set to 12 bits.

33.6.5 Differential and Single-Ended Conversions

The ADC has two conversion options: differential and single-ended:

32-bit ARM-Based Microcontrollers

Bit	7	6	5	4	3	2	1	0
								DBGRUN
Access								R/W
Reset								0

Bit 0 – DBGRUN: Debug Run

This bit can be changed only while the ADC is disabled.

This bit should be written only while a conversion is not ongoing.

Value	Description
0	The ADC is halted during debug mode.
1	The ADC continues normal operation during debug mode.

35.5.8 Register Access Protection

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

- Interrupt Flag Status and Clear (INTFLAG) register
- Data Buffer (DATABUF) register

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](#)

35.5.9 Analog Connections

The DAC has one output pin (VOUT) and one analog input pin (VREFA) that must be configured first.

When internal input is used, it must be enabled before DAC Controller is enabled.

35.6 Functional Description

35.6.1 Principle of Operation

The DAC converts the digital value located in the Data register (DATA) into an analog voltage on the DAC output (VOUT).

A conversion is started when new data is written to the Data register. The resulting voltage is available on the DAC output after the conversion time. A conversion can also be started by input events from the Event System.

35.6.2 Basic Operation

35.6.2.1 Initialization

The following registers are enable-protected, meaning they can only be written when the DAC is disabled (CTRLA.ENABLE is zero):

- Control B register (CTRLB)
- Event Control register (EVCTRL)

Enable-protection is denoted by the Enable-Protected property in the register description.

Before enabling the DAC, it must be configured by selecting the voltage reference using the Reference Selection bits in the Control B register (CTRLB.REFSEL).

35.6.2.2 Enabling, Disabling and Resetting

The DAC Controller is enabled by writing a '1' to the Enable bit in the Control A register (CTRLA.ENABLE). The DAC Controller is disabled by writing a '0' to CTRLA.ENABLE.

The DAC Controller is reset by writing a '1' to the Software Reset bit in the Control A register (CTRLA.SWRST). All registers in the DAC will be reset to their initial state, and the DAC Controller will be disabled. Refer to the CTRLA register for details.

35.6.2.3 Enabling the Output Buffer

To enable the DAC output on the V_{OUT} pin, the output driver must be enabled by writing a one to the External Output Enable bit in the Control B register (CTRLB.EOEN).

Errata reference: 12368

Fix/Workaround:

For pin PA24 and PA25, the GPIO pull-up and pull-down must be disabled before enabling alternative functions on them.

5 – If APB clock is stopped and GCLK clock is running, APB read access to read-synchronized registers will freeze the system. The CPU and the DAP AHB-AP are stalled, as a consequence debug operation is impossible.

Errata reference: 10416

Fix/Workaround:

Do not make read access to read-synchronized registers when APB clock is stopped and GCLK is running. To recover from this situation, power cycle the device or reset the device using the RESETN pin.

6 – In I2C Slave mode, writing the CTRLB register when in the AMATCH or DRDY interrupt service routines can cause the state machine to reset.

Errata reference: 13574

Fix/Workaround:

Write CTRLB.ACKACT to 0 using the following sequence:

```
// If higher priority interrupts exist, then disable so that the
// following two writes are atomic.
```

```
SERCOM - STATUS.reg = 0;
```

```
SERCOM - CTRLB.reg = 0;
```

```
// Re-enable interrupts if applicable.
```

Write CTRLB.ACKACT to 1 using the following sequence:

```
// If higher priority interrupts exist, then disable so that the
// following two writes are atomic.
```

```
SERCOM - STATUS.reg = 0;
```

```
SERCOM - CTRLB.reg = SERCOM_I2CS_CTRLB_ACKACT;
```

```
// Re-enable interrupts if applicable.
```

Otherwise, only write to CTRLB in the AMATCH or DRDY interrupts if it is to close out a transaction.

When not closing a transaction, clear the AMATCH interrupt by writing a 1 to its bit position instead of using CTRLB.CMD. The DRDY interrupt is automatically cleared by reading/writing to the DATA register in smart mode. If not in smart mode, DRDY should be cleared by writing a 1 to its bit position.

Code replacements examples:

Current:

```
SERCOM - CTRLB.reg |= SERCOM_I2CS_CTRLB_ACKACT;
```

Change to:

```
// If higher priority interrupts exist, then disable so that the
// following two writes are atomic.
```

```
SERCOM - STATUS.reg = 0;
```

```
SERCOM - CTRLB.reg = SERCOM_I2CS_CTRLB_ACKACT;
```

```
// Re-enable interrupts if applicable.
```

Current:

```
SERCOM - CTRLB.reg &= ~SERCOM_I2CS_CTRLB_ACKACT;
```

Change to:

```
// If higher priority interrupts exist, then disable so that the
```

Fix/Workaround:

None

12 – Pulldown functionality is not available on GPIO pin PA24 and PA25

Errata reference: 13883

Fix/Workaround:

None

13 – The voltage regulator in low power mode is not functional at temperatures above 85C.

Errata reference: 12291

Fix/Workaround:

Enable normal mode on the voltage regulator in standby sleep mode.

Example code:

```
// Set the voltage regulator in normal mode configuration in standby sleep mode
```

```
SYSCTRL->VREG.bit.RUNSTDBY = 1;
```

14 – If the external XOSC32K is broken, neither the external pin RST nor the GCLK software reset can reset the GCLK generators using XOSC32K as source clock.

Errata reference: 12164

Fix/Workaround:

Do a power cycle to reset the GCLK generators after an external XOSC32K failure.

40.1.3.2 DSU

1 – If a debugger has issued a DSU Cold-Plugging procedure and then released the CPU from the resulting ""CPU Reset Extension"", the CPU will be held in ""CPU Reset Extension"" after any upcoming reset event.

Errata reference: 12015

Fix/workaround:

The CPU must be released from the ""CPU Reset Extension"" either by writing a one in the DSU STATUSA.CRSTEXT register or by applying an external reset with SWCLK high or by power cycling the device.

2 – The MBIST ""Pause-on-Error"" feature is not functional on this device.

Errata reference: 14324

Fix/Workaround:

Do not use the ""Pause-on-Error"" feature.

40.1.3.3 PM

1 – In debug mode, if a watchdog reset occurs, the debug session is lost.

Errata reference: 12196

Fix/Workaround:

A new debug session must be restart after a watchdog reset.

40.1.3.4 DFLL48M

1 – The DFLL clock must be requested before being configured otherwise a write access to a DFLL register can freeze the device.

Errata reference: 9905

4 – In TWI master mode, an ongoing transaction should be stalled immediately when DBGCTRL.DBGSTOP is set and the CPU enters debug mode. Instead, it is stopped when the current byte transaction is completed and the corresponding interrupt is triggered if enabled.

Errata reference: 12499

Fix/Workaround:

In TWI master mode, keep DBGCTRL.DBGSTOP=0 when in debug mode.

40.1.4.12 TC

1 – Spurious TC overflow and Match/Capture events may occur.

Errata reference: 13268

Fix/Workaround:

Do not use the TC overflow and Match/Capture events. Use the corresponding Interrupts instead.

40.1.4.13 TCC

1 – Using TCC in dithering mode with external retrigger events can lead to unexpected stretch of right aligned pulses, or shrink of left aligned pulses.

Errata reference: 15625

Fix/Workaround:

Do not use retrigger events/actions when TCC is configured in dithering mode.

2 – Advance capture mode (CAPTMIN CAPTMAX LOCMIN LOCMAX DERIV0) doesn't work if an upper channel is not in one of these mode.

Example: when CC[0]=CAPTMIN, CC[1]=CAPTMAX, CC[2]=CAPTEN, and CC[3]=CAPTEN, CAPTMIN and CAPTMAX won't work.

Errata reference: 14817

Fix/Workaround:

Basic capture mode must be set in lower channel and advance capture mode in upper channel.

Example: CC[0]=CAPTEN , CC[1]=CAPTEN , CC[2]=CAPTMIN, CC[3]=CAPTMAX

All capture will be done as expected.

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