

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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

E·XF

Product Status	Active
Core Processor	ARM® Cortex®-M0+
Core Size	32-Bit Single-Core
Speed	48MHz
Connectivity	CANbus, I ² C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, WDT
Number of I/O	52
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 20x12b, 3x16b; D/A 1x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	64-VFQFN Exposed Pad
Supplier Device Package	64-QFN (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsamc21j15a-mnt

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

7.3.3 Fetching of Initial Instructions

After reset has been released, the CPU starts fetching PC and SP values from the reset address, which is 0x00000000. This address points to the first executable address in the internal flash. The code read from the internal flash is free to configure the clock system and clock sources. Refer to the ARM Architecture Reference Manual for more information on CPU startup (http://www.arm.com).

7.4 Power-On Reset and Brown-Out Detector

The SAM C20/C21 embed three features to monitor, warn, and/or reset the device:

- POR: Power-on reset on VDDIN and VDDIO
- BODVDD: Brown-out detector on VDDIN
- BODCORE: Voltage Regulator Internal Brown-out detector on VDDCORE. The Voltage Regulator Internal BOD is calibrated in production and its calibration configuration is stored in the NVM User Row. This configuration should not be changed if the user row is written to assure the correct behavior of the BODCORE.

7.4.1 Power-On Reset on VDDIN

POR monitors VDDIN. It is always activated and monitors voltage at startup and also during all the sleep modes. If VDDIN goes below the threshold voltage, the entire chip is reset.

7.4.2 Power-On Reset on VDDIO

POR monitors VDDIO. It is always activated and monitors voltage at startup and also during all the sleep modes. If VDDIO goes below the threshold voltage, all IOs supplied by VDDIO are reset.

7.4.3 Brown-Out Detector on VDDIN

BODVDD monitors VDDIN.

7.4.4 Brown-Out Detector on VDDCORE

Once the device has started up, BODCORE monitors the internal VDDCORE.

SAM C20/C21

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
Access								
Reset								
Bit	7	6	5	4	3	2	1	0
	FKBC[3:0]				JEPC	C[3:0]		
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 7:4 – FKBC[3:0]: 4KB Count

These bits will always return zero when read, indicating that this debug component occupies one 4KB block.

Bits 3:0 – JEPCC[3:0]: JEP-106 Continuation Code

These bits will always return zero when read.

13.13.15 Peripheral Identification 0

 Name:
 PID0

 Offset:
 0x1FE0

 Reset:
 0x0000000

 Property:

SAM C20/C21

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
Access		I	I					
Reset								
Bit	7	6	5	4	3	2	1	0
[PARTN	BL[7:0]			
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 7:0 – PARTNBL[7:0]: Part Number Low

These bits will always return 0xD0 when read, indicating that this device implements a DSU module instance.

13.13.16 Peripheral Identification 1

Name: PID1 Offset: 0x1FE4 Reset: 0x00000FC Property: -

SAM C20/C21



Bits 7:4 – JEPIDCL[3:0]: Low part of the JEP-106 Identity Code

These bits will always return 0xF when read (JEP-106 identity code is 0x1F).

Bits 3:0 – PARTNBH[3:0]: Part Number High

These bits will always return 0xC when read, indicating that this device implements a DSU module instance.

13.13.17 Peripheral Identification 2

 Name:
 PID2

 Offset:
 0x1FE8

 Reset:
 0x0000009

 Property:

Bit	31	30	29	28	27	26	25	24
				REMAIND	ER[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
				REMAIND)ER[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
				REMAINI	DER[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
				REMAIN	DER[7:0]			
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – REMAINDER[31:0]: Remainder of Operation

Holds the 32-bit remainder of the last performed operation. For a divide operation this is the division remainder. If the Signed bit in Control A register (CTRLA.SIGNED) is zero, the quotient is unsigned. If CTRLA.SIGNED = 1, the quotient is signed two's complement. For a square root operation this is the square root remainder. Refer to Performing Division, Operand Size and Signed Division.

14.8.7 Square Root Input

Name:SQRNUMOffset:0x18Reset:0x0000Property: -

REGA, REGB are 8-bit core registers. REGC is 16-bit core register.

Offset	Register
0x00	REGA
0x01	REGB
0x02	RECC
0x03	REGO

Since synchronization is per register, user can write REGA (8-bit access) then immediately write REGB (8-bit access) without error.

User can write REGC (16-bit access) without affecting REGA or REGB. But if user writes REGC in two consecutive 8-bit accesses, second write will be discarded and generate an error.

When user makes a 32-bit access to offset 0x00, all registers are written but REGA, REGB, REGC can be updated at a different time because of independent write synchronization

15.3.3 General Read-Synchronization

Before any read of a core register, the user must check that the related bit in SYNCBUSY register is cleared.

Read access to core register is always immediate but the return value is reliable only if a synchronization of this core register is not going.

15.3.4 Completion of Synchronization

The user can either poll SYNCBUSY register or use the Synchronization Ready interrupt (if available) to check when the synchronization is complete.

15.3.5 Enable Write-Synchronization

Writing to the Enable bit in the Control register (CTRL.ENABLE) will also trigger write-synchronization and set SYNCBUSY.ENABLE. CTRL.ENABLE will read its new value immediately after being written. The Synchronisation Ready interrupt (if available) cannot be used for Enable write-synchronization.

15.3.6 Software Reset Write-Synchronization

Writing a one to the Software Reset bit in CTRL (CTRL.SWRST) will also trigger write-synchronization and set SYNCBUSY.SWRST. When writing a one to the CTRL.SWRST bit it will immediately read as one. CTRL.SWRST and SYNCBUSY.SWRST will be cleared by hardware when the peripheral has been reset. Writing a zero to the CTRL.SWRST bit has no effect. The Synchronisation Ready interrupt (if available) cannot be used for Software Reset write-synchronization.

15.3.7 Synchronization Delay

The synchronization will delay the write or read access duration by a delay D, given by the equation:

 $5 \cdot P_{\text{GCLK}} + 2 \cdot P_{\text{APB}} < D < 6 \cdot P_{\text{GCLK}} + 3 \cdot P_{\text{APB}}$

Where P_{GCLK} is the period of the generic clock and P_{APB} is the period of the peripheral bus clock. A normal peripheral bus register access duration is $2 \cdot P_{\text{APB}}$.

20.6.2 External Multipurpose Crystal Oscillator (XOSC) Operation

The XOSC can operate in two different modes:

- External clock, with an external clock signal connected to the XIN pin
- Crystal oscillator, with an external 0.4-32MHz crystal

The XOSC can be used as a clock source for generic clock generators. This is configured by the Generic Clock Controller.

At reset, the XOSC is disabled, and the XIN/XOUT pins can be used as General Purpose I/O (GPIO) pins or by other peripherals in the system. When XOSC is enabled, the operating mode determines the GPIO usage. When in crystal oscillator mode, the XIN and XOUT pins are controlled by the OSCCTRL, and GPIO functions are overridden on both pins. When in external clock mode, only the XIN pin will be overridden and controlled by the OSCCTRL, while the XOUT pin can still be used as a GPIO pin.

The XOSC is enabled by writing a '1' to the Enable bit in the External Multipurpose Crystal Oscillator Control register (XOSCCTRL.ENABLE).

To enable XOSC as an external crystal oscillator, the XTAL Enable bit (XOSCCTRL.XTALEN) must be written to '1'. If XOSCCTRL.XTALEN is zero, the external clock input on XIN will be enabled.

When in crystal oscillator mode (XOSCCTRL.XTALEN=1), the External Multipurpose Crystal Oscillator Gain (XOSCCTRL.GAIN) must be set to match the external crystal oscillator frequency. If the External Multipurpose Crystal Oscillator Automatic Amplitude Gain Control (XOSCCTRL.AMPGC) is '1', the oscillator amplitude will be automatically adjusted, and in most cases result in a lower power consumption.

The XOSC will behave differently in different sleep modes, based on the settings of XOSCCTRL.RUNSTDBY, XOSCCTRL.ONDEMAND, and XOSCCTRL.ENABLE. If XOSCCTRL.ENABLE=0, the XOSC will be always stopped. For XOSCCTRL.ENABLE=1, this table is valid:

CPU Mode	XOSCCTRL.RUNST DBY	XOSCCTRL.ONDEM AND	Sleep Behavior
Active or Idle	-	0	Always run
Active or Idle	-	1	Run if requested by peripheral
Standby	1	0	Always run
Standby	1	1	Run if requested by peripheral
Standby	0	_	Run if requested by peripheral

Table 20-1. XOSC Sleep Behavior

After a hard reset, or when waking up from a sleep mode where the XOSC was disabled, the XOSC will need a certain amount of time to stabilize on the correct frequency. This start-up time can be configured by changing the Oscillator Start-Up Time bit group (XOSCCTRL.STARTUP) in the External Multipurpose Crystal Oscillator Control register. During the start-up time, the oscillator output is masked to ensure that no unstable clock propagates to the digital logic.

The External Multipurpose Crystal Oscillator Ready bit in the Status register (STATUS.XOSCRDY) is set once the external clock or crystal oscillator is stable and ready to be used as a clock source. An interrupt is generated on a zero-to-one transition on STATUS.XOSCRDY if the External Multipurpose Crystal Oscillator Ready bit in the Interrupt Enable Set register (INTENSET.XOSCRDY) is set.

DPLL96M

Due to the multiple clock domains, some registers in the DPLL96M must be synchronized when accessed.

When executing an operation that requires synchronization, the relevant synchronization bit in the Synchronization Busy register (DPLLSYNCBUSY) will be set immediately, and cleared when synchronization is complete.

The following bits need synchronization when written:

- Enable bit in control register A (DPLLCTRLA.ENABLE)
- DPLL Ratio register (DPLLRATIO)
- DPLL Prescaler register (DPLLPRESC)

Related Links

Register Synchronization

Bit	31	30	29	28	27	26	25	24
Access								
Reset								
Bit	23	22	21	20	19	18	17	16
						LDRFR	AC[3:0]	
Access		•	•		R/W	R/W	R/W	R/W
Reset					0	0	0	0
Bit	15	14	13	12	11	10	9	8
						LDR[[11:8]	
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0
Bit	7	6	5	4	3	2	1	0
				LDR	[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 19:16 – LDRFRAC[3:0]: Loop Divider Ratio Fractional Part

Writing these bits selects the fractional part of the frequency multiplier. Due to synchronization there is a delay between writing these bits and the effect on the DPLL output clock. The value written will read back immediately and the DPLLRATIO bit in the DPLL Synchronization Busy register (DPLLSYNCBUSY.DPLLRATIO) will be set. DPLLSYNCBUSY.DPLLRATIO will be cleared when the operation is completed.

Bits 11:0 – LDR[11:0]: Loop Divider Ratio

Writing these bits selects the integer part of the frequency multiplier. The value written to these bits will read back immediately, and the DPLLRATIO bit in the DPLL Synchronization busy register (DPLLSYNCBUSY.DPLLRATIO), will be set. DPLLSYNCBUSY.DPLLRATIO will be cleared when the operation is completed.

20.8.14 DPLL Control B

Name:DPLLCTRLBOffset:0x24 [ID-00001eee]Reset:0x00Property:Enable-Protected, PAC Write-Protection

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.

24.8.1 Control A in COUNT32 mode (CTRLA.MODE=0)

 Name:
 CTRLA

 Offset:
 0x00

 Reset:
 0x0000

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

Bit	15	14	13	12	11	10	9	8
	COUNTSYNC					PRESCA	LER[3:0]	
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
	MATCHCLR				MODE[1:0] ENABLE SV		SWRST	
Access	R/W			·	R/W	R/W	R/W	R/W
Reset	0				0	0	0	0

Bit 15 – COUNTSYNC: COUNT Read Synchronization Enable

The COUNT register requires synchronization when reading. Disabling the synchronization will prevent reading valid values from the COUNT register.

This bit is not enable-protected.

Value	Description
0	COUNT read synchronization is disabled
1	COUNT read synchronization is enabled

Bits 11:8 - PRESCALER[3:0]: Prescaler

These bits define the prescaling factor for the RTC clock source (GCLK_RTC) to generate the counter clock (CLK_RTC_CNT). Periodic events and interrupts are not available when the prescaler is off. These bits are not synchronized.

Value	Name	Description
0x0	OFF	CLK_RTC_CNT = GCLK_RTC/1
0x1	DIV1	CLK_RTC_CNT = GCLK_RTC/1
0x2	DIV2	CLK_RTC_CNT = GCLK_RTC/2
0x3	DIV4	CLK_RTC_CNT = GCLK_RTC/4
0x4	DIV8	CLK_RTC_CNT = GCLK_RTC/8
0x5	DIV16	CLK_RTC_CNT = GCLK_RTC/16
0x6	DIV32	CLK_RTC_CNT = GCLK_RTC/32
0x7	DIV64	CLK_RTC_CNT = GCLK_RTC/64
0x8	DIV128	CLK_RTC_CNT = GCLK_RTC/128
0x9	DIV256	CLK_RTC_CNT = GCLK_RTC/256

Name:SWEVTOffset:0x1C [ID-0000120d]Reset:0x00000000Property:PAC Write-Protection



Bits 11:0 – CHANNELn: Channel n Software [n=11..0] Selection

Writing '0' to this bit has no effect.

Writing '1' to this bit will trigger a software event for the channel n.

These bits will always return zero when read.

Related Links

PAC - Peripheral Access Controller

29.8.7 Channel

This register allows the user to configure channel n. To write to this register, do a single, 32-bit write of all the configuration data.

 Name:
 CHANNELn

 Offset:
 0x20+n*0x4 [0..11n=0..11] [ID-0000120d]

 Reset:
 0x0000000

 Property:
 PAC Write-Protection

Bit 1 – FERR: Frame Error

Reading this bit before reading the Data register will indicate the error status of the next character to be read.

This bit is cleared by writing '1' to the bit or by disabling the receiver.

This bit is set if the received character had a frame error, i.e., when the first stop bit is zero.

Writing '0' to this bit has no effect.

Writing '1' to this bit will clear it.

Bit 0 – PERR: Parity Error

Reading this bit before reading the Data register will indicate the error status of the next character to be read.

This bit is cleared by writing '1' to the bit or by disabling the receiver.

This bit is set if parity checking is enabled (CTRLA.FORM is 0x1, 0x5) and a parity error is detected.

Writing '0' to this bit has no effect.

Writing '1' to this bit will clear it.

31.8.10 Synchronization Busy

Name:	SYNCBUSY			
Offset:	0x1C [ID-00000fa7]			
Reset:	0x0000000			
Property: -				

CCCR		Tx Buffer Element		Frame Transmission	
BRSE	FDOE	FDF	BRS		
ignored	0	ignored	ignored	Classic CAN	
0	1	0	ignored	Classic CAN	
0	1	1	ignored	FD without bit rate switching	
1	1	0	ignored	Classic CAN	
1	1	1	0	FD without bit rate switching	
1	1	1	1	FD with bit rate switching	

Table 34-6. Possi	ible Configurations	for Frame	Transmission
-------------------	---------------------	-----------	--------------

Note: AUTOSAR requires at least three Tx Queue Buffers and support of transmit cancellation

The Tx Handler starts a Tx scan to check for the highest priority pending Tx request (Tx Buffer with lowest Message ID) when the Tx Buffer Request Pending register TXBRP is updated, or when a transmission has been started.

34.6.6.1 Transmit Pause

The transmit pause feature is intended for use in CAN systems where the CAN message identifiers are (permanently) specified to specific values and cannot easily be changed. These message identifiers may have a higher CAN arbitration priority than other defined messages, while in a specific application their relative arbitration priority should be inverse. This may lead to a case where one ECU sends a burst of CAN messages that cause another ECU's CAN messages to be delayed because that other messages have a lower CAN arbitration priority.

If e.g. CAN ECU-1 has the transmit pause feature enabled and is requested by its application software to transmit four messages, it will, after the first successful message transmission, wait for two CAN bit times of bus idle before it is allowed to start the next requested message. If there are other ECUs with pending messages, those messages are started in the idle time, they would not need to arbitrate with the next message of ECU-1. After having received a message, ECU-1 is allowed to start its next transmission as soon as the received message releases the CAN bus.

The transmit pause feature is controlled by bit CCCR.TXP. If the bit is set, the CAN will, each time it has successfully transmitted a message, pause for two CAN bit times before starting the next transmission. This enables other CAN nodes in the network to transmit messages even if their messages have lower prior identifiers. Default is transmit pause disabled (CCCR.TXP = '0').

This feature looses up burst transmissions coming from a single node and it protects against "babbling idiot" scenarios where the application program erroneously requests too many transmissions.

34.6.6.2 Dedicated Tx Buffers

Dedicated Tx Buffers are intended for message transmission under complete control of the CPU. Each Dedicated Tx Buffer is configured with a specific Message ID. In case that multiple Tx Buffers are configured with the same Message ID, the Tx Buffer with the lowest buffer number is transmitted first.

If the data section has been updated, a transmission is requested by an Add Request via TXBAR.ARn. The requested messages arbitrate internally with messages from an optional Tx FIFO or Tx Queue and externally with messages on the CAN bus, and are sent out according to their Message ID.

A Dedicated Tx Buffer allocates Element Size 32-bit words in the Message RAM (refer to table below). Therefore the start address of a dedicated Tx Buffer in the Message RAM is calculated by adding transmit buffer index (0...31) • Element Size to the Tx Buffer Start Address TXBC.TBSA.

Bits 5:4 – PRESCSYNC[1:0]: Prescaler and Counter Synchronization

These bits select whether the counter should wrap around on the next GCLK_TCx clock or the next prescaled GCLK_TCx clock. It also makes it possible to reset the prescaler.

These bits are not synchronized.

Value	Name	Description
0x0	GCLK	Reload or reset the counter on next generic clock
0x1	PRESC	Reload or reset the counter on next prescaler clock
0x2	RESYNC	Reload or reset the counter on next generic clock. Reset the prescaler counter
0x3	-	Reserved

Bits 3:2 – MODE[1:0]: Timer Counter Mode

These bits select the counter mode.

These bits are not synchronized.

Value	Name	Description
0x0	COUNT16	Counter in 16-bit mode
0x1	COUNT8	Counter in 8-bit mode
0x2	COUNT32	Counter in 32-bit mode
0x3	-	Reserved

Bit 1 – ENABLE: Enable

Due to synchronization, there is delay from writing CTRLA.ENABLE until the peripheral is enabled/ disabled. The value written to CTRLA.ENABLE will read back immediately, and the ENABLE Synchronization Busy bit in the SYNCBUSY register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.

This bit is not enable protected.

Value	Description
0	The peripheral is disabled.
1	The peripheral is enabled.

Bit 0 – SWRST: Software Reset

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

Writing a '1' to this bit resets all registers in the TC, except DBGCTRL, to their initial state, and the TC will be disabled.

Writing a '1' to CTRLA.SWRST will always take precedence; all other writes in the same write-operation will be discarded.

Due to synchronization there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the reset is complete.

This bit is not enable protected.

Value	Description
0	There is no reset operation ongoing.
1	The reset operation is ongoing.

35.7.2.2 Control B Clear

- Dead-time insertion
- Swap
- Pattern generation

See also Figure 36-1.

The output matrix (OTMX) can distribute and route out the TCC waveform outputs across the port pins in different configurations, each optimized for different application types. The Dead-Time Insertion (DTI) unit splits the four lower OTMX outputs into two non-overlapping signals: the non-inverted low side (LS) and inverted high side (HS) of the waveform output with optional dead-time insertion between LS and HS switching. The SWAP unit can swap the LS and HS pin outputs, and can be used for fast decay motor control.

The pattern generation unit can be used to generate synchronized waveforms with constant logic level on TCC UPDATE conditions. This is useful for easy stepper motor and full bridge control.

The non-recoverable fault module enables event controlled fault protection by acting directly on the generated waveforms of the timer/counter compare channel outputs. When a non-recoverable fault condition is detected, the output waveforms are forced to a preconfigured value that is safe for the application. This is typically used for instant and predictable shut down and disabling high current or voltage drives.

The count event sources (TCE0 and TCE1) are shared with the non-recoverable fault extension. The events can be optionally filtered. If the filter options are not used, the non-recoverable faults provide an immediate asynchronous action on waveform output, even for cases where the clock is not present. For further details on how to configure asynchronous events routing, refer to section *EVSYS – Event System*.

Related Links

EVSYS – Event System

36.6.2 Basic Operation

36.6.2.1 Initialization

The following registers are enable-protected, meaning that they can only be written when the TCC is disabled(CTRLA.ENABLE=0):

- Control A (CTRLA) register, except Run Standby (RUNSTDBY), Enable (ENABLE) and Software Reset (SWRST) bits
- Recoverable Fault n Control registers (FCTRLA and FCTRLB)
- Waveform Extension Control register (WEXCTRL)
- Drive Control register (DRVCTRL)
- Event Control register (EVCTRL)

Enable-protected bits in the CTRLA register can be written at the same time as CTRLA.ENABLE is written to '1', but not at the same time as CTRLA.ENABLE is written to '0'. Enable-protection is denoted by the "Enable-Protected" property in the register description.

Before the TCC is enabled, it must be configured as outlined by the following steps:

- 1. Enable the TCC bus clock (CLK_TCCx_APB).
- 2. If Capture mode is required, enable the channel in capture mode by writing a '1' to the Capture Enable bit in the Control A register (CTRLA.CPTEN).

Optionally, the following configurations can be set before enabling TCC:

- 1. Select PRESCALER setting in the Control A register (CTRLA.PRESCALER).
- 2. Select Prescaler Synchronization setting in Control A register (CTRLA.PRESCSYNC).

Value	Name	Description
0x0	OFF	Event action disabled.
0x1	RETRIGGER	Start restart or re-trigger TC on event
0x2	COUNTEV	Count on event.
0x3	START	Start TC on event
0x4	INC	Increment TC on EVENT
0x5	COUNT (async)	Count on active state of asynchronous event
0x6		
0x7	FAULT	Non-recoverable Fault

36.8.10 Interrupt Enable Clear

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 Set (INTENSET) register.

Name:INTENCLROffset:0x24 [ID-00002e48]Reset:0x0000000Property:PAC Write-Protection

Bit	23	22	21	20	19	18	17	16
					MCx	MCx	MCx	MCx
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0
Bit	15	14	13	12	11	10	9	8
	FAULTx	FAULTx	FAULTB	FAULTA	DFS	UFS		
Access	R/W	R/W	R/W	R/W	R/W	R/W		
Reset	0	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 clear the corresponding Match or Capture Channel x Interrupt Disable/Enable bit, which disables 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 clear the Non-Recoverable Fault x Interrupt Disable/Enable bit, which disables the Non-Recoverable Fault x interrupt.

The CCx register represents the 16-, 24- bit value, CCx. The register has two functions, depending of the mode of operation.

For capture operation, this register represents the second buffer level and access point for the CPU and DMA.

For compare operation, this register is continuously compared to the counter value. Normally, the output form the comparator is then used for generating waveforms.

CCx register is updated with the buffer value from their corresponding CCBUFx register when an UPDATE condition occurs.

In addition, in match frequency operation, the CC0 register controls the counter period.

Name: CC Offset: 0x44 + n*0x04 [n=0..3] Reset: 0x00000000 Property: Write-Synchronized, Read-Synchronized

Bit	31	30	29	28	27	26	25	24
Access								
Reset								
D.1	00	00	04	00	40	40	47	40
BIT	23	22	21	20	19	18	17	16
				CC[1	7:10]			
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
Γ				CC[9:2]			
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
Γ	CC	[1:0]			DITHE	R[5:0]		
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 23:6 – CC[17:0]: Channel x Compare/Capture Value

These bits hold the value of the Channel x compare/capture register.

Note: When the TCC is configured as 16-bit timer/counter, the excess bits are read zero.

Note: This bit field occupies the m MSB of the register, [23:m]. m is dependent on the Resolution bit in the Control A register (CTRLA.RESOLUTION):

CTRLA.RESOLUTION	Bits [23:m]
0x0 - NONE	23:0
0x1 - DITH4	23:4

Bit 20 – INVEI: Inverted Event Input Enable

Value	Description
0	Incoming event is not inverted.
1	Incoming event is inverted.

Bit 7 – EDGESEL: Edge Selection

Value	Description
0	Edge detector is disabled.
1	Edge detector is enabled.

Bits 5:4 – FILTSEL[1:0]: Filter Selection

These bits select the LUT output filter options:

Filter Selection

Value	Name	Description
0x0	DISABLE	Filter disabled
0x1	SYNCH	Synchronizer enabled
0x2	FILTER	Filter enabled
0x3	-	Reserved

Bit 1 – ENABLE: LUT Enable

Value	Description
0	The LUT is disabled.
1	The LUT is enabled.

Bits 19:16,15:12,11:8 – INSELx: LUT Input x Source Selection

These bits select the LUT input x source:

Value	Name	Description
0x0	MASK	Masked input
0x1	FEEDBACK	Feedback input source
0x2	LINK	Linked LUT input source
0x3	EVENT	Event input source
0x4	IO	I/O pin input source
0x5	AC	AC input source
0x6	TC	TC input source
0x7	ALTTC	Alternative TC input source
0x8	TCC	TCC input source
0x9	SERCOM	SERCOM input source
0xA	ALT2TC	Alternative 2 TC input source

Offset	Name	Bit Pos.								
0x29		15:8	SEQENn	SEQENn	SEQENn	SEQENn	SEQENn	SEQENn	SEQENn	SEQENn
0x2A		23:16	SEQENn	SEQENn	SEQENn	SEQENn	SEQENn	SEQENn	SEQENn	SEQENn
0x2B		31:24	SEQENn	SEQENn	SEQENn	SEQENn	SEQENn	SEQENn	SEQENn	SEQENn
0x2C	CALID	7:0						E	BIASCOMP[2:0]
0x2D	CALIB	15:8						BIASREFBUF[2:0]		

38.8 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 are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to the section on Synchronization.

Some registers are synchronized when read and/or written. Synchronization is denoted by the "Write-Synchronized" or the "Read-Synchronized" property in each individual register description. For details, refer to Synchronization section.

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

38.8.1 Control A

Name:CTRLAOffset:0x00 [ID-0000120e]Reset:0x00Property:PAC Write-Protection, Write-Synchronized

Bit	7	6	5	4	3	2	1	0
	ONDEMAND	RUNSTDBY	SLAVEEN				ENABLE	SWRST
Access	R/W	R/W	R/W				R/W	R/W
Reset	0	0	0				0	0

Bit 7 – ONDEMAND: On Demand Control

The On Demand operation mode allows the ADC to be enabled or disabled, depending on other peripheral requests.

In On Demand operation mode, i.e., if the ONDEMAND bit has been previously set, the ADC will only be running when requested by a peripheral. If there is no peripheral requesting the ADC will be in a disable state.

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

In standby sleep mode, the On Demand operation is still active if the CTRLA.RUNSTDBY bit is '1'. If CTRLA.RUNSTDBY is '0', the ADC is disabled.

This bit is not synchronized.

For the slave ADC, this bit has no effect when the SLAVEEN bit is set (CTRLA.SLAVEEN= 1). ONDEMAND bit from master ADC instance will control the On Demand operation mode.

Bit 2 – FLUSHINV: Flush Event Invert Enable

For the slave ADC, this bit has no effect when the SLAVEEN bit is set (CTRLA.SLAVEEN= 1).

Value	Description
0	Flush event input source is not inverted.
1	Flush event input source is inverted.

Bit 1 – STARTEI: Start Conversion Event Input Enable

For the slave ADC, this bit has no effect when the SLAVEEN bit is set (CTRLA.SLAVEEN= 1).

Value	Description
0	A new conversion will not be triggered on any incoming event.
1	A new conversion will be triggered on any incoming event.

Bit 0 – FLUSHEI: Flush Event Input Enable

For the slave ADC, this bit has no effect when the SLAVEEN bit is set (CTRLA.SLAVEEN= 1).

Value	Description
0	A flush and new conversion will not be triggered on any incoming event.
1	A flush and new conversion will be triggered on any incoming event.

38.8.5 Interrupt Enable Clear

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

Name:INTENCLROffset:0x04 [ID-0000120e]Reset:0x00Property:PAC Write-Protection

Bit	7	6	5	4	3	2	1	0
						WINMON	OVERRUN	RESRDY
Access						R/W	R/W	R/W
Reset						0	0	0

Bit 2 – WINMON: Window Monitor Interrupt Enable

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

Writing a '1' to this bit will clear the Window Monitor Interrupt Enable bit, which disables the corresponding interrupt request.

Value	Description
0	The window monitor interrupt is disabled.
1	The window monitor interrupt is enabled, and an interrupt request will be generated when the
	Window Monitor interrupt flag is set.

Bit 1 – OVERRUN: Overrun Interrupt Enable

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

Writing a '1' to this bit will clear the Overrun Interrupt Enable bit, which disables the corresponding interrupt request.