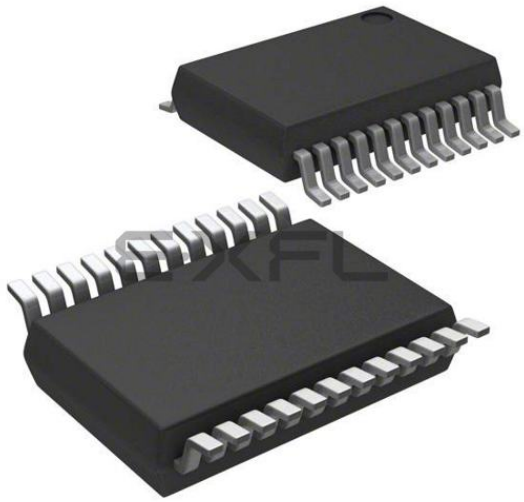


Welcome to E-XFL.COM

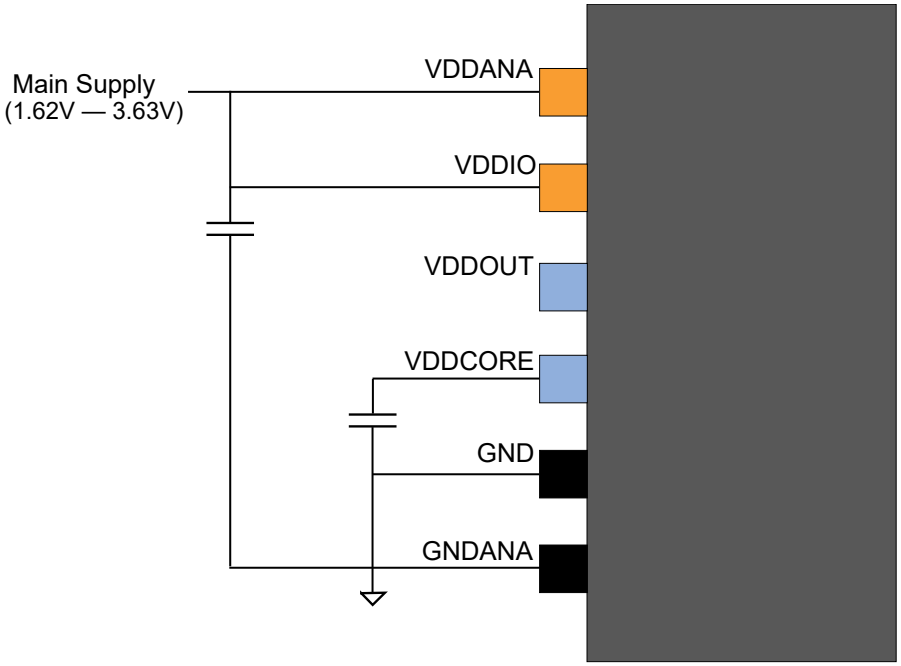
What is "Embedded - Microcontrollers"?

"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"

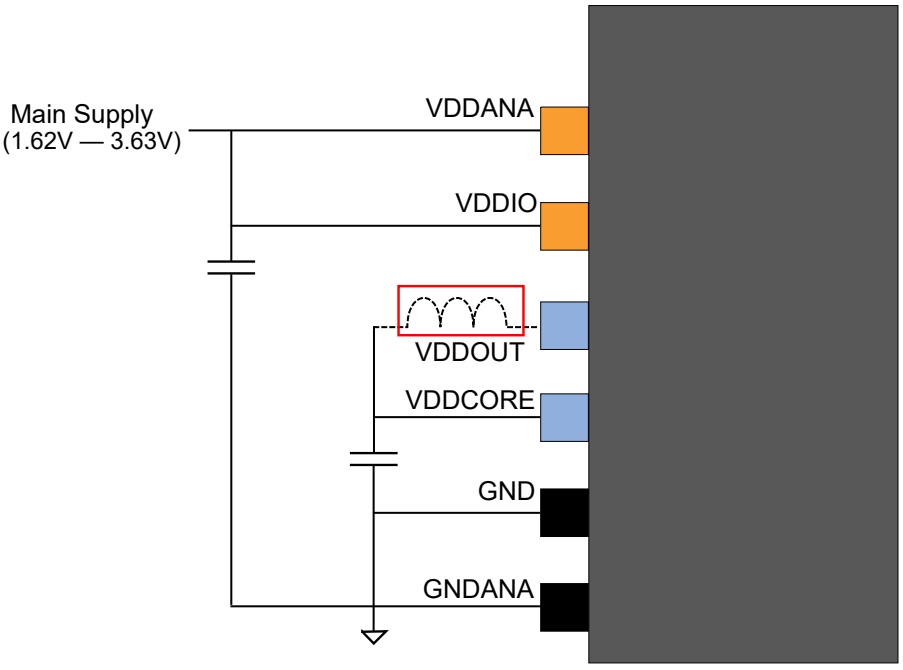
Details	
Product Status	Active
Core Processor	ARM® Cortex®-M23
Core Size	32-Bit Single-Core
Speed	32MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	17
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	1.62V ~ 3.63V
Data Converters	A/D 5x12b; D/A 1x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	24-SSOP (0.209", 5.30mm Width)
Supplier Device Package	24-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsaml10d15a-yu

Figure 6-2. Power Supply Connections for Linear (LDO) Mode Only

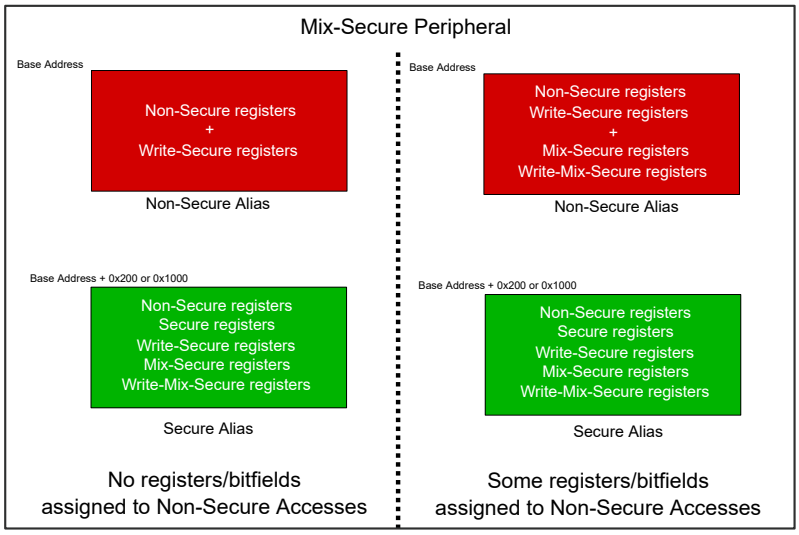


Note: Refer to "Schematic Checklist" chapter for additional information.

Figure 6-3. Power Supply Connections for Switching (BUCK) / Linear (LDO) Modes



Note: Refer to "Schematic Checklist" chapter for additional information.



Mix-Secure peripherals have always the following registers:

- NONSEC register is a generic register that tells the Non-Secure application which resources inside a Mix-Secure peripheral can be used
- NSCHK register is a register allowing the Non-Secure application to be notified when the security configuration of a Mix-Secure peripheral is being modified during application execution

Important: It is recommended that the Non-Secure application first copy the content of NONSEC register inside NSCHK register, and then enable the NSCHK interrupt flags. Once done, any changes to the NONSEC register by the Secure application will trigger an interrupt so that Non-Secure application can take appropriate actions. This mechanism allows the Secure application to dynamically change the security attribution of a Mix-Secure peripheral and avoid illegal accesses from the Non-Secure application. The interrupt handler should always copy the NONSEC register to NSCHK register before exiting it.

Mix-Secure peripherals can have five type of registers:

- **Non-Secure**: these registers will always be available in both the Secure and Non-Secure aliases
- **Secure**: these registers will never be available in the Non-Secure alias and always available in the Secure alias
- **Write-Secure**: these are registers than can:
  - Be written or read by the Secure application only in the Secure alias
  - Only read by the Non-Secure application in Non-Secure alias. Write is forbidden.
- **Mix-Secure** registers : these ones are used when a resource can be allocated to either the Secure and Non-Secure alias
  - Note that, in some cases, the Mix-Secure properties apply to a bitfield only (like one I/O bit in the PORT peripheral register)
- **Write-Mix-Secure** registers (NVMCTRL peripheral only): these are Mix-Secure registers, which:
  - can be written or read by the Secure application only in the Secure alias
  - can only be read by the Non-Secure application in Non-Secure alias **except** if Non-Secure writes are authorized in NVMCTRL.NONSEC register

14. Boot ROM

The Boot ROM allows to ensure the integrity of the device at boot.

The Boot ROM features Boot Interactive mode, which allows the user to perform several actions on the device, such as NVM areas integrity check and chip erase via a debugger connection.

Unless a debugger is connected and places the Boot ROM in Boot Interactive mode, the CPU will jump to the Flash memory, loading the Program Counter (PC) and Stack Pointer (SP) values, and will start fetching Flash user code.
Note: Before jumping to the Flash, the Boot ROM resets the two first 2kB of SRAM. The Clocks remain unchanged.

In addition, the SAM L11 Boot ROM has extra security features, such as device integrity checks, memories/peripherals security attributions, and secure boot, which can be executed before jumping to the Flash in Secure state.

For security reasons, while the Boot ROM is executing, no debug is possible except when entering a specific Boot ROM mode called CPU Park mode.

Related Links

13.1 Features

14.1 Features

- Command interface for the host debugger supporting:
  - Chip erase commands to provide secure transitions between the different Debug Access Levels (DAL)
  - Device integrity check of the NVM memory regions
  - Debugger read access of the NVM rows
- CPU Park mode to get access for a debugger to the resources of the device depending on Debug Access Level (DAL)
- SAM L11 Added features:
  - Device integrity checks
  - Memory and peripheral security attributions from user configuration stored in NVM rows
  - Secure Boot on Flash BS Memory Area

Related Links

13.1 Features

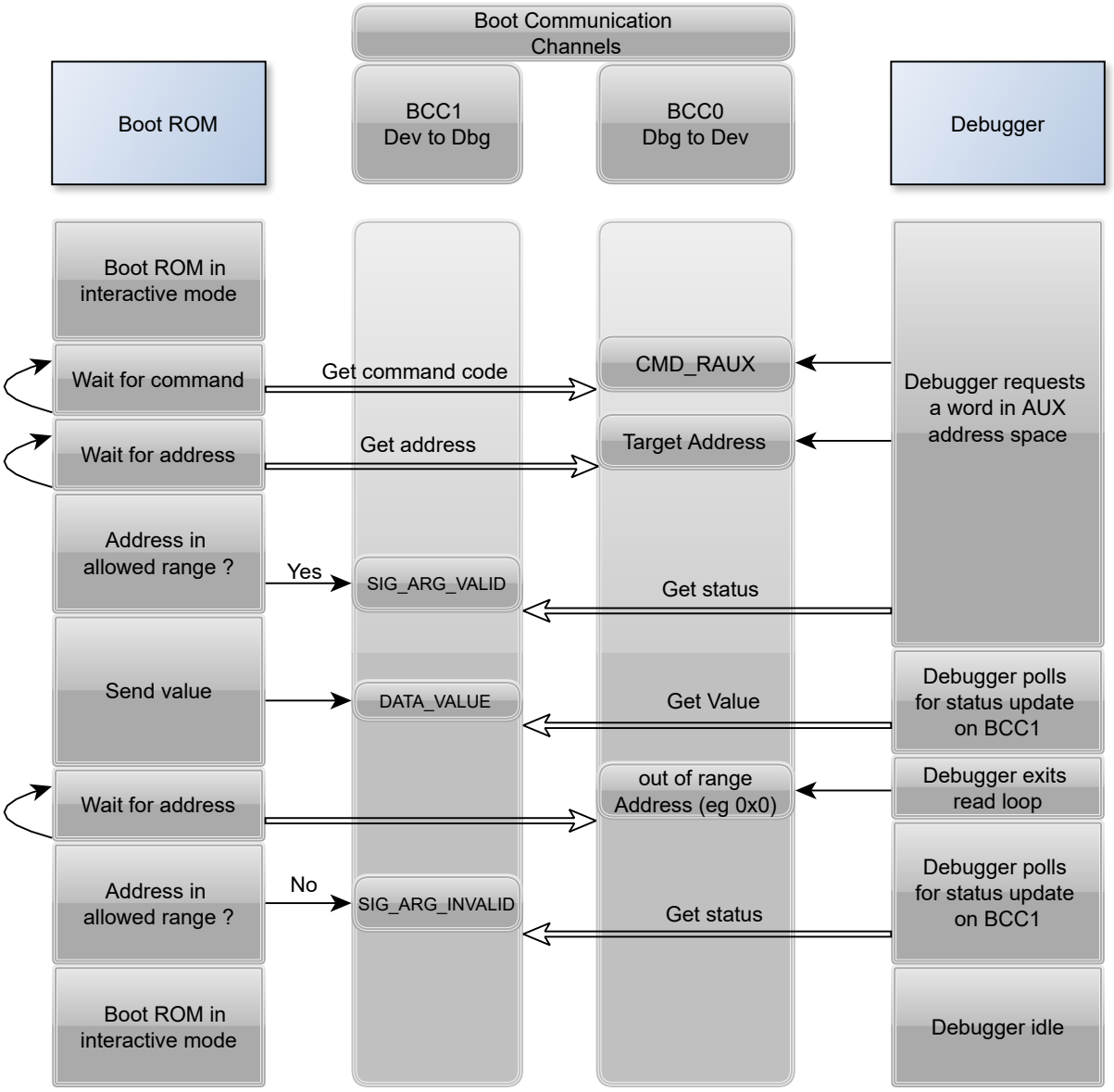
The following areas are accessible:

Table 14-8. Accessible Memory Range by Read Auxiliary Row Command

Area	Start address	End address
User row (UROW)	0x00804000	0x0080401F
Software Calibration row	0x00806020	0x0080602F
Temperature Log row	0x00806038	0x0080603F
Boot Configuration row (BOCOR)	0x0080C000	0x0080C0FF

14.4.5.8.1 CMD_RAUX

Figure 14-15. CMD_RAUX Flow diagram



Note: After the CMD_RAUX is sent, the debugger can read multiple data, the read loop is exit when an out of range address is sent.

16.12.25 Component Identification 3

Name: CID3
Offset: 0x1FFC
Reset: 0x000000B1
Property: -

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
	PREAMBLEB3[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	1	0	1	1	0	0	0	1

Bits 7:0 – PREAMBLEB3[7:0] Preamble Byte 3
These bits will always return 0x000000B1 when read.

18.8.2 Synchronization Busy

Name: SYNCBUSY
Offset: 0x04
Reset: 0x00000000
Property: –

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
		GENCTRL4	GENCTRL3	GENCTRL2	GENCTRL1	GENCTRL0		SWRST
Access		R	R	R	R	R		R
Reset		0	0	0	0	0		0

Bits 2, 3, 4, 5, 6 – GENCTRL Generator Control n Synchronization Busy
This bit is cleared when the synchronization of the Generator Control n register (GENCTRLn) between clock domains is complete, or when clock switching operation is complete.

This bit is set when the synchronization of the Generator Control n register (GENCTRLn) between clock domains is started.

Bit 0 – SWRST Software Reset Synchronization Busy
This bit is cleared when the synchronization of the CTRLA.SWRST register bit between clock domains is complete.

This bit is set when the synchronization of the CTRLA.SWRST register bit between clock domains is started.

20.8.1 Control A

Name: CTRLA
Offset: 0x00
Reset: 0x00
Property: PAC Write-Protection, Write-Synchronized, Read-Synchronized

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

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 bit in the Synchronization Busy 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 FREQM to their initial state, and the FREQM will be disabled. Writing a '1' to this bit will always take precedence, meaning that 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 ongoing Reset operation.
1	The Reset operation is ongoing.

Table 22-4. Regulators, RAMs, and NVM state in Sleep Mode

Sleep Mode	PDSW	SRAM Mode(1)	NVM	Regulators	
				VDDCORE	
				main	ULP
Active	active	normal	normal	on	on
Idle	active	auto(2)	on	on	on
Standby - PDSW in Active mode	active	normal(6)	auto(2)	auto(3)	on(5)
Standby - PDSW in Retention mode	retention	low power(6)	low power	auto(4)	on(5)
OFF	off	off	off	off	off

Note:

1. RAMs mode by default: STDBYCFG.BBIAS bits are set to their default value.
2. auto: by default, NVM is in low-power mode if not accessed.
3. auto: by default, the main voltage regulator is on if GCLK, APBx, or AHBx clock is running during SleepWalking.
4. auto: by default ULP regulator is selected in retention, but main regulator will be selected if VREG RUNSTDBY register bit in Supply Controller is set to 1.
5. on: low power voltage reference must be ready, and this is confirmed if STATUS.ULPVREFRDY register bit in SUPC equals to 1
6. SRAM can be partially retained in STANDBY using SRAM Power Switch

Related Links

22.6.4.4 Regulator Automatic Low Power Mode

22.6.4 Advanced Features

22.6.4.1 Power Domain Configuration

When entering Standby Sleep mode, a power domain is set automatically to retention state if no activity is required in it, refer to 22.6.3.5 Power Domain Controller for details. This behavior can be changed by writing the Power Domain Configuration bit group in the Standby Configuration register (STDBYCFG.PDCFG). For example, all power domains can be forced to remain in active state during Standby Sleep mode, this will accelerate wake-up time.

22.6.4.2 RAM Automatic Low Power Mode

The RAM is by default put in Low-Power mode (back-biased) if its power domain is in retention state and the device is in Standby Sleep mode.

This behavior can be changed by configuring BBIASxx bit groups in the Standby Configuration register (STDBYCFG.BBIASxx), refer to the table below for details.

Note: in Standby Sleep mode, the DMAC can access the SRAM in Standby Sleep mode only when the power domain PDSW is not in retention and PM.STDBYCFG.BBIASxx=0x0.

25.8.7 Voltage References System (VREF) Control

Name: VREF
Offset: 0x1C
Reset: 0x00000000
Property: PAC Write-Protection

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
					SEL[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
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
	ONDEMAND	RUNSTDBY				VREFOE	TSEN	
Access	R/W	R/W				R/W	R/W	
Reset	0	0				0	0	

Bits 19:16 – SEL[3:0] Voltage Reference Selection
These bits select the Voltage Reference for the ADC/DAC.

Value	Name	Description
0x0	1V0	1.0V voltage reference typical value
0x1	1V1	1.1V voltage reference typical valueThe 1.1V voltage reference typical value must be selected for DAC use. Other values are not permitted.
0x2	1V2	1.2V voltage reference typical value
0x3	1V25	1.25V voltage reference typical value
0x4	2V0	2.0V voltage reference typical value
0x5	2V2	2.2V voltage reference typical value
0x6	2V4	2.4V voltage reference typical value
0x7	2V5	2.5V voltage reference typical value
Others		Reserved

Bit 7 – ONDEMAND On Demand Control
The On Demand operation mode allows to enable or disable the voltage reference depending on peripheral requests.

Figure 27-8. Edge Detection with Asynchronous Stability Debouncing

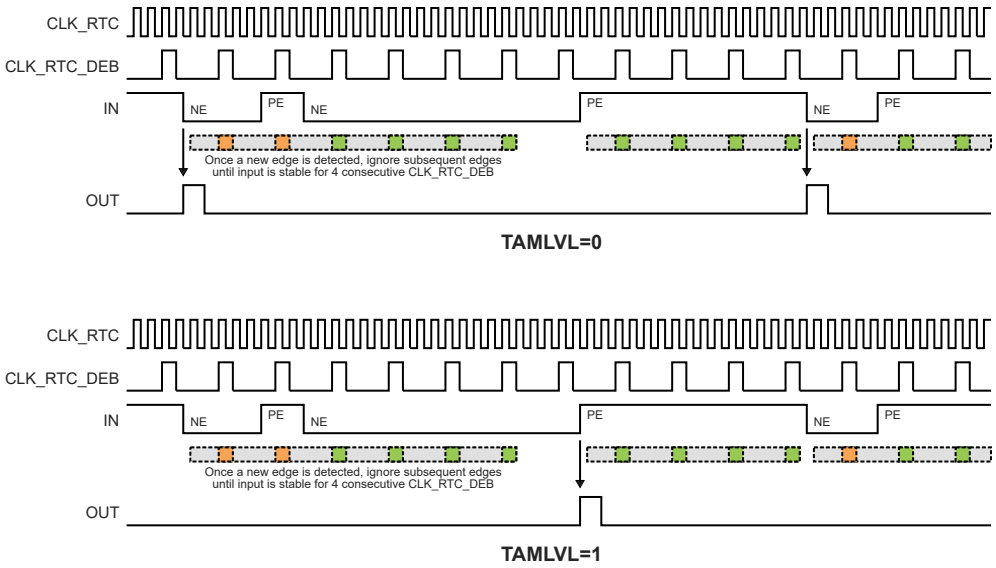
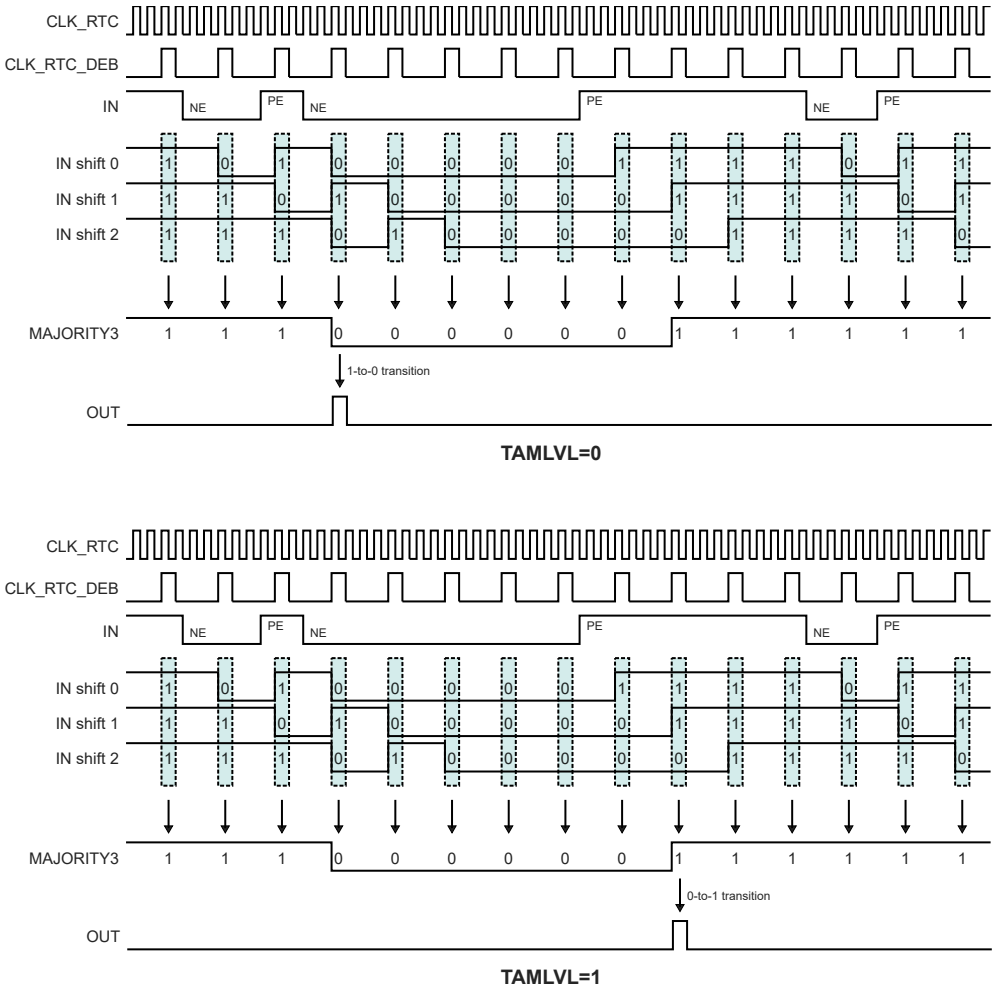


Figure 27-9. Edge Detection with Majority Debouncing



27.12.4 Interrupt Enable Clear in Clock/Calendar mode (CTRLA.MODE=2)

Name: INTENCLR
Offset: 0x08
Reset: 0x0000
Property: PAC Write-Protection

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.

Bit	15	14	13	12	11	10	9	8
	OVF	TAMPER						ALARM0
Access	R/W	R/W						R/W
Reset	0	0						0

Bit	7	6	5	4	3	2	1	0
	PER7	PER6	PER5	PER4	PER3	PER2	PER1	PER0
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 – OVF Overflow Interrupt Enable
Writing a '0' to this bit has no effect. Writing a '1' to this bit will clear the Overflow Interrupt Enable bit, which disables the Overflow interrupt.

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

Bit 14 – TAMPER Tamper Interrupt Enable

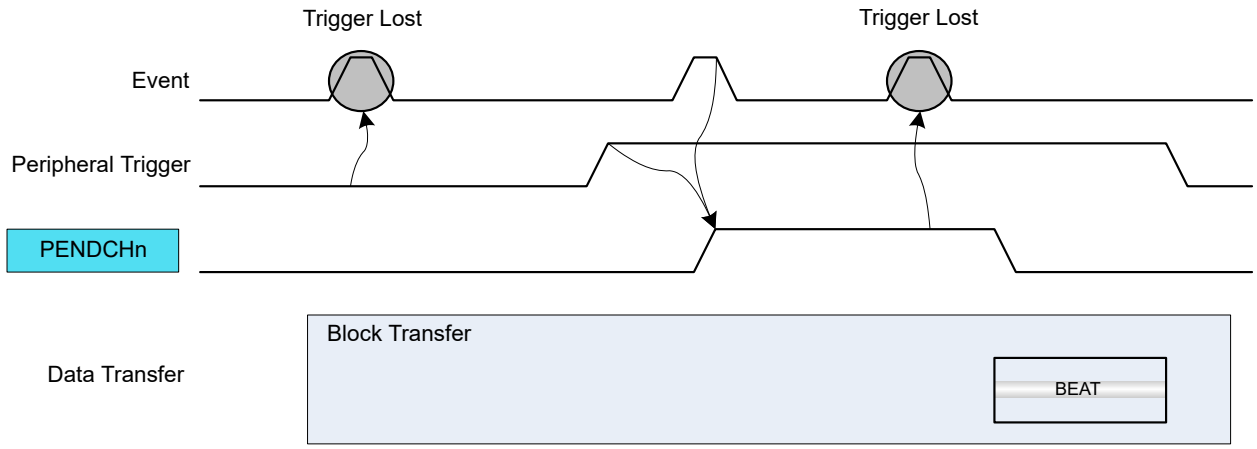
Bit 8 – ALARM0 Alarm 0 Interrupt Enable
Writing a '0' to this bit has no effect. Writing a '1' to this bit will clear the Alarm 0 Interrupt Enable bit, which disables the Alarm interrupt.

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

Bits 0, 1, 2, 3, 4, 5, 6, 7 – PERn Periodic Interval n Interrupt Enable [n = 7..0]
Writing a '0' to this bit has no effect. Writing a '1' to this bit will clear the Periodic Interval n Interrupt Enable bit, which disables the Periodic Interval n interrupt.

Value	Description
0	Periodic Interval n interrupt is disabled.
1	Periodic Interval n interrupt is enabled.

Figure 28-12. Periodic Event with Beat Peripheral Triggers



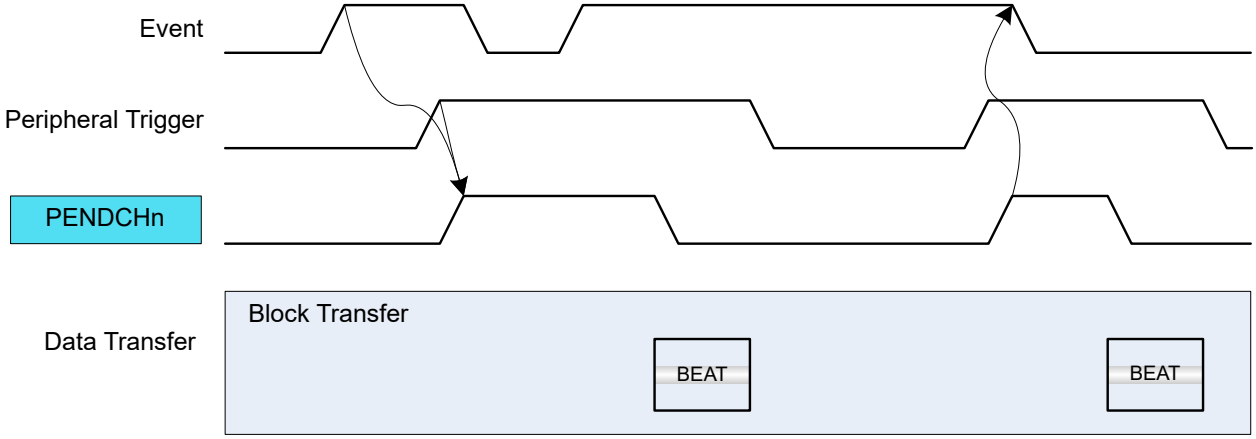
Conditional Transfer

The event input is used to trigger a conditional transfer on peripherals with pending transfer requests. As example, this type of event can be used for peripheral-to-peripheral transfers, where one peripheral is the source of event and the second peripheral is the source of the trigger.

Each peripheral trigger is stored internally when the event is received. When the peripheral trigger is stored internally, the Channel Pending status bit is set (CHSTATUS.PEND), the respective Pending Channel n Bit in the Pending Channels register is set (28.8.13 PENDCH.PENDCHn), and the event is acknowledged. A software trigger will now trigger a transfer.

The figure below shows an example where conditional event is enabled with peripheral beat trigger requests.

Figure 28-13. Conditional Event with Beat Peripheral Triggers



Conditional Block Transfer

The event input is used to trigger a conditional block transfer on peripherals.

Before starting transfers within a block, an event must be received. When received, the event is acknowledged when the block transfer is completed. A software trigger will trigger a transfer.

The figure below shows an example where conditional event block transfer is started with peripheral beat trigger requests.

Bits 3:0 – ID[3:0] Channel ID
These bits store the lowest channel number with pending interrupts. The number is valid if Suspend (SUSP), Transfer Complete (TCMPL) or Transfer Error (TERR) bits are set. The Channel ID field is refreshed when a new channel (with channel number less than the current one) with pending interrupts is detected, or when the application clears the corresponding channel interrupt sources. When no pending channels interrupts are available, these bits will always return zero value when read.

When the bits are written, indirect access to the corresponding Channel Interrupt Flag register is enabled.

Figure 32-6. I/O Configuration - Totem-Pole Output with Disabled Input

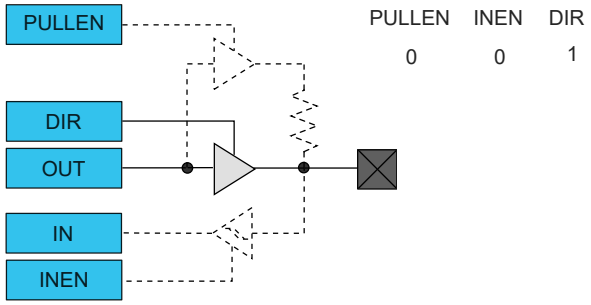


Figure 32-7. I/O Configuration - Totem-Pole Output with Enabled Input

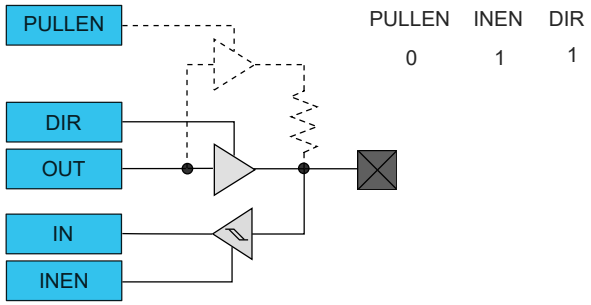
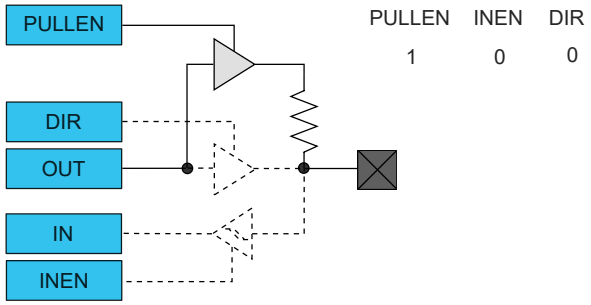


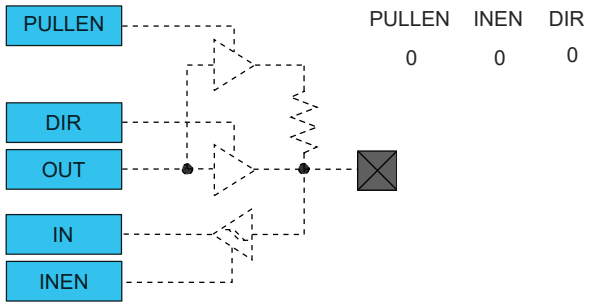
Figure 32-8. I/O Configuration - Output with Pull



32.6.3.4 Digital Functionality Disabled

Neither Input nor Output functionality are enabled.

Figure 32-9. I/O Configuration - Reset or Analog I/O: Digital Output, Input and Pull Disabled



32.6.4 SAM L11 Secure Access Rights

Non-secure write to CTRL, EVCTRL, or NONSEC registers is prohibited.

Non-secure read to CTRL or EVCTRL registers will return zero with no error resulting.

Value	Description
0	The corresponding I/O pin in the PORT group is configured as an input.
1	The corresponding I/O pin in the PORT group is configured as an output.

33.5.3 Interrupts

The EVSYS has the following interrupt sources for each channel:

- Overrun Channel n interrupt (OVR)
- Event Detected Channel n interrupt (EVD)

These interrupts events are asynchronous wake-up sources.

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the corresponding Channel n Interrupt Flag Status and Clear (CHINTFLAG) register is set when the interrupt condition occurs.

Note: Interrupts must be globally enabled to allow the generation of interrupt requests.

Each interrupt can be individually enabled by writing a '1' to the corresponding bit in the Channel n Interrupt Enable Set (CHINTENSET) register, and disabled by writing a '1' to the corresponding bit in the Channel n Interrupt Enable Clear (CHINTENCLR) register. An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled or the Event System is reset. All interrupt requests are ORed together on system level to generate one combined interrupt request to the NVIC.

The user must read the Channel Interrupt Status (INTSTATUS) register to identify the channels with pending interrupts, and must read the Channel n Interrupt Flag Status and Clear (CHINTFLAG) register to determine which interrupt condition is present for the corresponding channel. It is also possible to read the Interrupt Pending register (INTPEND), which provides the highest priority channel with pending interrupt and the respective interrupt flags.

33.5.4 Sleep Mode Operation

The Event System can generate interrupts to wake up the device from IDLE or STANDBY sleep mode.

To be able to run in standby, the Run in Standby bit in the Channel register (CHANNELn.RUNSTDBY) must be set to '1'. When the Generic Clock On Demand bit in Channel register (CHANNELn.ONDEMAND) is set to '1' and the event generator is detected, the event channel will request its clock (GCLK_EVSYS_CHANNEL_n). The event latency for a resynchronized channel path will increase by two GCLK_EVSYS_CHANNEL_n clock (i.e., up to five GCLK_EVSYS_CHANNEL_n clock cycles).

A channel will behave differently in different sleep modes regarding to CHANNELn.RUNSTDBY and CHANNELn.ONDEMAND:

Table 33-1. Event Channel Sleep Behavior

CHANNELn.PATH	CHANNELn.ONDEMAND	CHANNELn.RUNSTDBY	Sleep Behavior
ASYNC	0	0	Only run in IDLE sleep modes if an event must be propagated. Disabled in STANDBY sleep mode.
SYNC/RESYNC	0	1	Run in both IDLE and STANDBY sleep modes.
SYNC/RESYNC	1	0	Only run in IDLE sleep modes if an event must be propagated. Disabled in STANDBY sleep mode. Two GCLK_EVSYS_n latency

35.8.12 Data

Name: DATA
Offset: 0x28
Reset: 0x0000
Property: -

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

Bit	7	6	5	4	3	2	1	0
	DATA[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 8:0 – DATA[8:0] Data
Reading these bits will return the contents of the Receive Data register. The register should be read only when the Receive Complete Interrupt Flag bit in the Interrupt Flag Status and Clear register (INTFLAG.RXC) is set. The status bits in STATUS should be read before reading the DATA value in order to get any corresponding error.

Writing these bits will write the Transmit Data register. This register should be written only when the Data Register Empty Interrupt Flag bit in the Interrupt Flag Status and Clear register (INTFLAG.DRE) is set.

Value	Name	Description
0x0	OFF	Event action disabled
0x1	RETRIGGER	Start, restart or retrigger TC on event
0x2	COUNT	Count on event
0x3	START	Start TC on event
0x4	STAMP	Time stamp capture
0x5	PPW	Period captured in CC0, pulse width in CC1
0x6	PWP	Period captured in CC1, pulse width in CC0
0x7	PW	Pulse width capture

41.8.9 Input Control

Name: INPUTCTRL
Offset: 0x08
Reset: 0x0000
Property: PAC Write-Protection, Write-Synchronized

Bit	15	14	13	12	11	10	9	8
				MUXNEG[4:0]				
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
				MUXPOS[4:0]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bits 12:8 – MUXNEG[4:0] Negative MUX Input Selection
These bits define the MUX selection for the negative ADC input.

Value	Name	Description
0x00	AIN0	ADC AIN0 pin
0x01	AIN1	ADC AIN1 pin
0x02	AIN2	ADC AIN2 pin
0x03	AIN3	ADC AIN3 pin
0x04	AIN4	ADC AIN4 pin
0x05	AIN5	ADC AIN5 pin
0x06	AIN6	ADC AIN6 pin
0x07	AIN7	ADC AIN7 pin
0x08 – 0x17	-	Reserved
0x18	GND	Internal ground
0x19 – 0x1F	-	Reserved

Bits 4:0 – MUXPOS[4:0] Positive MUX Input Selection
These bits define the MUX selection for the positive ADC input. If the internal bandgap voltage or temperature sensor input channel is selected, then the Sampling Time Length bit group in the Sampling Control register must be written with a corresponding value.

Value	Name	Description
0x00	AIN0	ADC AIN0 pin
0x01	AIN1	ADC AIN1 pin
0x02	AIN2	ADC AIN2 pin
0x03	AIN3	ADC AIN3 pin
0x04	AIN4	ADC AIN4 pin
0x05	AIN5	ADC AIN5 pin
0x06	AIN6	ADC AIN6 pin

1. These values are only given as a typical example.

2. The capacitors should be placed close to the device for each supply pin pair in the signal group.

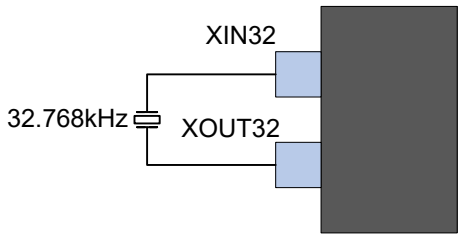
50.6.3 External Real Time Oscillator

The low frequency crystal oscillator is optimized for use with a 32.768 kHz watch crystal. When selecting crystals, load capacitance and the crystal's Equivalent Series Resistance (ESR) must be taken into consideration. Both values are specified by the crystal vendor.

SAM L10/L11 oscillator is optimized for very low power consumption, hence close attention should be made when selecting crystals.

The typical parasitic load capacitance values are available in the Electrical Characteristics chapters. This capacitance and PCB capacitance can allow using a crystal inferior to 12.5 pF load capacitance without external capacitors as shown in the following figure.

Figure 50-8. External Real Time Oscillator without Load Capacitor



To improve accuracy and Safety Factor, the crystal data sheet can recommend adding external capacitors, as shown in the following figure.

To find suitable load capacitance for a 32.768 kHz crystal, consult the crystal data sheet.

Figure 50-9. External Real Time Oscillator with Load Capacitor

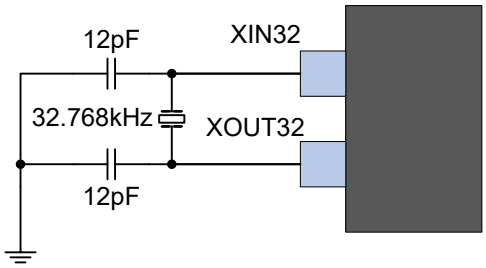


Table 50-6. External Real Time Oscillator Checklist

Signal Name	Recommended Pin Connection	Description
XIN32	Load capacitor 12 pF(1,2)	Timer oscillator input
XOUT32	Load capacitor 12 pF(1,2)	Timer oscillator output

1. These values are only given as typical examples.

2. The capacitors should be placed close to the device for each supply pin pair in the signal group.

Note: To minimize the cycle-to-cycle jitter of the external oscillator, keep the neighboring pins as steady as possible. For neighboring pin details, refer to 4.2 Oscillators Pinout.