

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®-M23
Core Size	32-Bit Single-Core
Speed	32MHz
Connectivity	I ² C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	17
Program Memory Size	16KB (16K 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-VFQFN Exposed Pad
Supplier Device Package	24-VQFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsaml11d14a-mu

14.4.2.1.3 CRC Computation and Programming

The CRCs need to be recalculated and updated in their respective NVM row as soon as a data from any of the checked regions is changed.



Important: USERCRC and BOCORCRC CRCs programming must be done by any programming tool supporting the SAM L11 devices.

The algorithm is a CRC-32 module embedded in the DSU peripheral and that uses for both CRC calculation with the following parameters:

- Width = 32 bits
- Polynomial = 0x04C11DB7 (Poly)
- Initial Value = 0xFFFFFFFF (Init)
- Input Data is reflected (RefIn)
- Output Data is reflected (RefOut)
- No XOR is performed on the output CRC (XorOut)

Example: the DSU CRC of 0x31, 0x32, 0x33, 0x34, 0x35, 0x36, 0x37, 0x38, 0x39 is 0x340BC6D9

14.4.2.2 Memories and Peripherals Configurations Initialization

For SAM L11 devices, memories and peripherals security attributions are done by reading the different fuses values from the NVM User (UROW) and Boot Configuration (BOCOR) rows.

The Boot ROM is responsible for setting these attributions on the different concerned memory and peripheral controllers:

- Set memory security attribution according to AS, ANSC, DS, RS, BS, BSNC and BOOTPROT fuses
- Set peripherals security attribution according to NONSECA, NONSECB and NONSECC fuses



Important: The Boot ROM does not perform any consistency checks on the configured memory attributions (e.g setting BS>BOOTPROT will not trigger any errors during Boot ROM execution).

14.4.2.3 Secure Boot

Depending on the BOOTOPT fuse value (from BOCOR NVM row), the following secure boot integrity checks will be performed on:

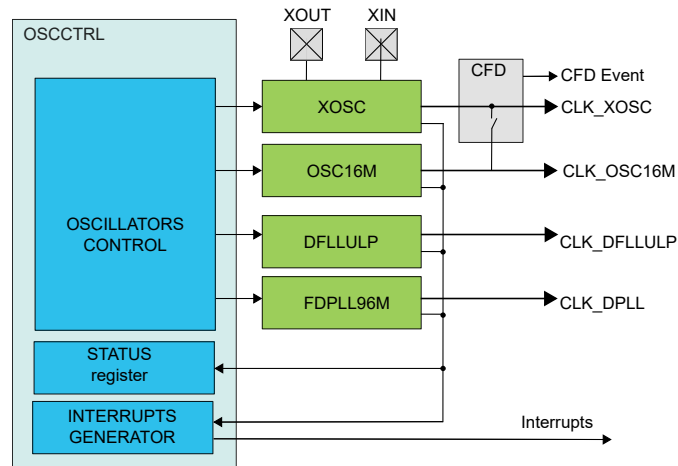
- The Flash BS memory area which is composed by:
 - The Flash Secure BOOT memory region
 - The Flash Non-Secure Callable BOOT memory region
- And the NVM Boot Configuration row (BOCOR)

Table 14-4. Secure Boot Options

BOOTOPT	Verified Areas	Verification Method
0	None	-
1	Flash BS Memory Region + NVM BOCOR row	SHA-256

23.3 Block Diagram

Figure 23-1. OSCCTRL Block Diagram



23.4 Signal Description

Signal	Description	Type
XIN	Multipurpose Crystal Oscillator or external clock generator input	Analog input
XOUT	Multipurpose Crystal Oscillator output	Analog output

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

23.5 Product Dependencies

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

23.5.1 I/O Lines

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

23.5.2 Power Management

The OSCCTRL can continue to operate in any sleep mode where the selected source clock is running. The OSCCTRL 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.

Related Links

[22. PM – Power Manager](#)

23.5.3 Clocks

The OSCCTRL gathers controls for all device oscillators and provides clock sources to the Generic Clock Controller (GCLK). The available clock sources are XOSC, OSC16M, DFLLULP and FDPLL96M.

The OSCCTRL bus clock (CLK_OSCCTRL_APB) can be enabled and disabled in the Main Clock module (MCLK).

23.8.14 DFLLULP Synchronization Busy

Name: DFLLULPSYNCBUSY
Offset: 0x28
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	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	
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	
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	
					DELAY			ENABLE	
Access	R	R	R	R	R		R	R	
Reset	0	0	0	0	0		0	0	

Bit 3 – DELAY Delay Register Synchronization Busy

This bit is cleared when the synchronization of DFLLULPDLY is complete.

This bit is set when the synchronization of DFLLULPDLY is started.

Writing this bit has no effect.

Bit 1 – ENABLE Enable Bit Synchronization Busy

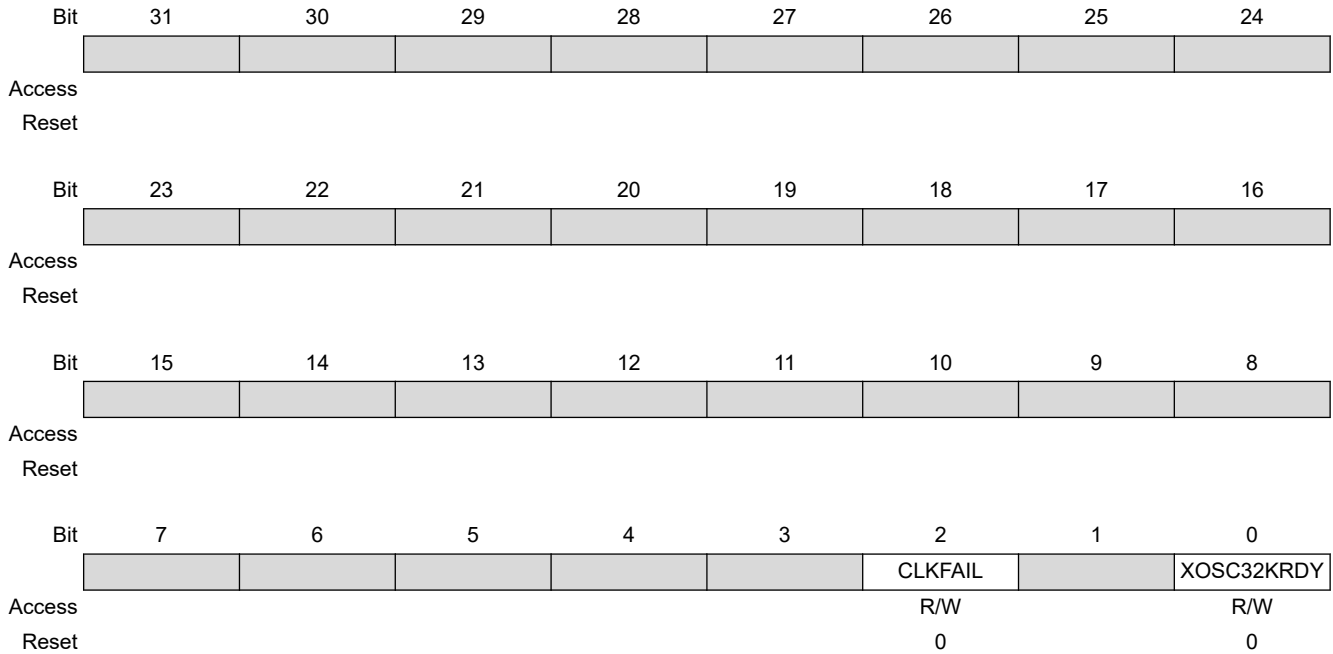
This bit is cleared when the synchronization of DFLLULPCTRL.ENABLE is complete.

This bit is set when the synchronization of DFLLULPCTRL.ENABLE is started.

Writing this bit has no effect.

24.8.3 Interrupt Flag Status and Clear

Name: INTFLAG
Offset: 0x08
Reset: 0x00000000
Property: –



Bit 2 – CLKFAIL XOSC32K Clock Failure Detection

This flag is cleared by writing a '1' to it.

This flag is set on a zero-to-one transition of the XOSC32K Clock Failure Detection bit in the Status register (STATUS.CLKFAIL) and will generate an interrupt request if INTENSET.CLKFAIL is '1'.

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

Writing a '1' to this bit will clear the XOSC32K Clock Failure Detection flag.

Bit 0 – XOSC32KRDY XOSC32K Ready

This flag is cleared by writing a '1' to it.

This flag is set by a zero-to-one transition of the XOSC32K Ready bit in the Status register (STATUS.XOSC32KRDY), and will generate an interrupt request if INTENSET.XOSC32KRDY=1.

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

Writing a '1' to this bit clears the XOSC32K Ready interrupt flag.

26.8.3 Early Warning Control

Name: EWCTRL
Offset: 0x02
Reset: x initially determined from NVM User Row after reset
Property: PAC Write-Protection

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

Bits 3:0 – EWOFFSET[3:0] Early Warning Interrupt Time Offset

These bits determine the number of GCLK_WDT clock cycles between the start of the watchdog time-out period and the generation of the Early Warning interrupt. These bits are loaded from NVM User Row at start-up.

Value	Name	Description
0x0	CYC8	8 clock cycles
0x1	CYC16	16 clock cycles
0x2	CYC32	32 clock cycles
0x3	CYC64	64 clock cycles
0x4	CYC128	128 clock cycles
0x5	CYC256	256 clock cycles
0x6	CYC512	512 clock cycles
0x7	CYC1024	1024 clock cycles
0x8	CYC2048	2048 clock cycles
0x9	CYC4096	4096 clock cycles
0xA	CYC8192	8192 clock cycles
0xB – 0xF	Reserved	Reserved

27.6.8 Additional Features

27.6.8.1 Periodic Intervals

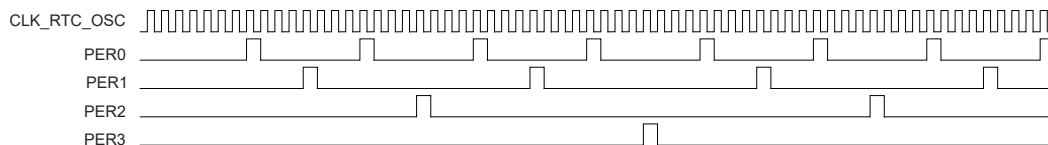
The RTC prescaler can generate interrupts and events at periodic intervals, allowing flexible system tick creation. Any of the upper eight bits of the prescaler (bits 2 to 9) can be the source of an interrupt/event. When one of the eight Periodic Event Output bits in the Event Control register (EVCTRL.PEREO[n=0..7]) is '1', an event is generated on the 0-to-1 transition of the related bit in the prescaler, resulting in a periodic event frequency of:

$$f_{\text{PERIODIC}(n)} = \frac{f_{\text{CLK_RTC_OSC}}}{2^{n+3}}$$

$f_{\text{CLK_RTC_OSC}}$ is the frequency of the internal prescaler clock CLK_RTC_OSC, and n is the position of the EVCTRL.PEREO[n] bit. For example, PER0 will generate an event every eight CLK_RTC_OSC cycles, PER1 every 16 cycles, etc. This is shown in the figure below.

Periodic events are independent of the prescaler setting used by the RTC counter, except if CTRLA.PRESCALER is zero. Then, no periodic events will be generated.

Figure 27-5. Example Periodic Events



27.6.8.2 Frequency Correction

The RTC Frequency Correction module employs periodic counter corrections to compensate for a too-slow or too-fast oscillator. Frequency correction requires that CTRLA.PRESCALER is greater than 1.

The digital correction circuit adds or subtracts cycles from the RTC prescaler to adjust the frequency in approximately 1ppm steps. Digital correction is achieved by adding or skipping a single count in the prescaler once every 8192 CLK_RTC_OSC cycles. The Value bit group in the Frequency Correction register (FREQCORR.VALUE) determines the number of times the adjustment is applied over 128 of these periods. The resulting correction is as follows:

$$\text{Correction in ppm} = \frac{\text{FREQCORR.VALUE}}{8192 \cdot 128} \cdot 10^6 \text{ ppm}$$

This results in a resolution of 0.95367ppm.

The Sign bit in the Frequency Correction register (FREQCORR.SIGN) determines the direction of the correction. A positive value will add counts and increase the period (reducing the frequency), and a negative value will reduce counts per period (speeding up the frequency).

Digital correction also affects the generation of the periodic events from the prescaler. When the correction is applied at the end of the correction cycle period, the interval between the previous periodic event and the next occurrence may also be shortened or lengthened depending on the correction value.

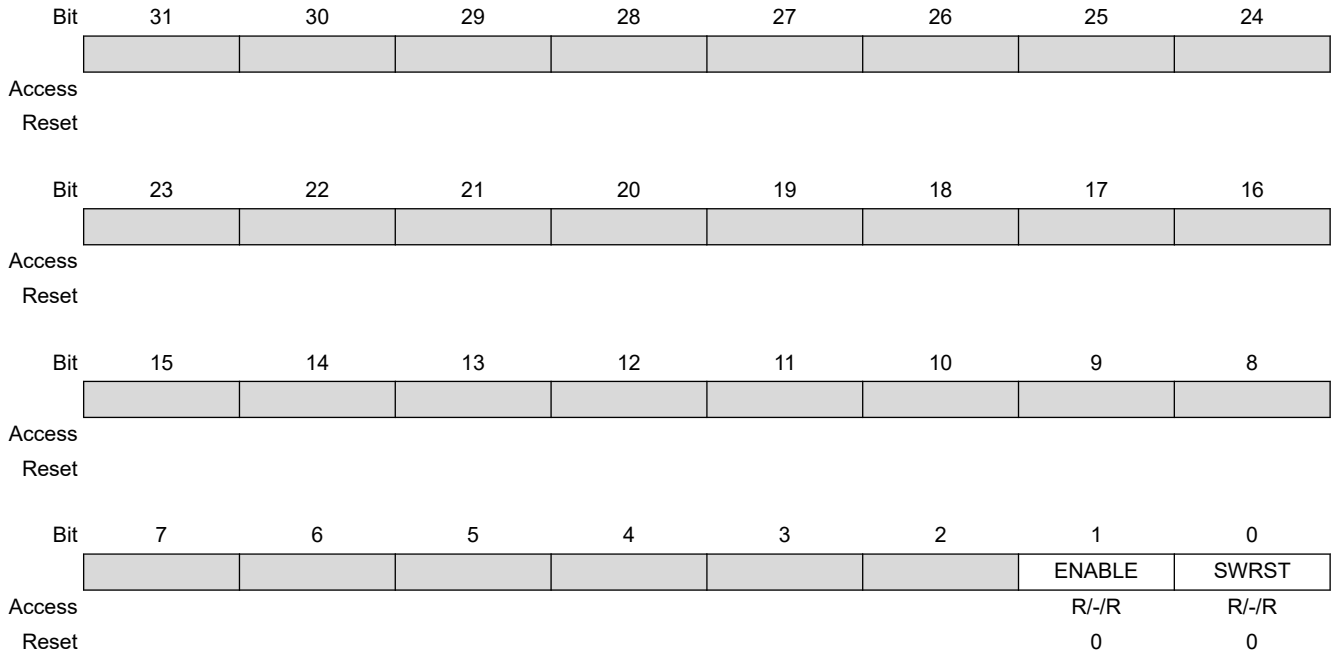
27.6.8.3 General Purpose Registers

The RTC includes four General Purpose registers (GPn). These registers are reset only when the RTC is reset or when tamper detection occurs while CTRLA.GPTRST=1, and remain powered while the RTC is powered. They can be used to store user-defined values while other parts of the system are powered off.

The general purpose registers 2*n and 2*n+1 are enabled by writing a '1' to the General Purpose Enable bit n in the Control B register (CTRLB.GPnEN).

29.8.4 Synchronization Busy

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



Bit 1 – ENABLE Enable Synchronization Busy Status

Value	Description
0	Write synchronization for CTRLA.ENABLE bit is complete.
1	Write synchronization for CTRLA.ENABLE bit is ongoing.

Bit 0 – SWRST Software Reset Synchronization Busy Status

Value	Description
0	Write synchronization for CTRLA.SWRST bit is complete.
1	Write synchronization for CTRLA.SWRST bit is ongoing.

29.8.7 Interrupt Enable Set

Name: INTENSET
Offset: 0x10
Reset: 0x00000000
Property: PAC Write-Protection, Mix-Secure



Important: For **SAM L11 Non-Secure** accesses, read and write accesses (RW*) are allowed only if the external interrupt x (EXTINTx) is set as Non-Secure in the NONSEC register (NONSEC.EXTINTx bit).

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.

	Bit	31	30	29	28	27	26	25	24
		NSCHK							
Access		RW/RW/RW							
Reset		0							
	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
		EXTINT[7:0]							
Access		RW/RW*/RW	RW/RW*/RW	RW/RW*/RW	RW/RW*/RW	RW/RW*/RW	RW/RW*/RW	RW/RW*/RW	RW/RW*/RW
Reset		0	0	0	0	0	0	0	0

Bit 31 – NSCHK Non-secure Check Interrupt Enable
 Writing a '0' to this bit has no effect.
 Writing a '1' to this bit will set the NSCHK Interrupt Enable bit.

Bits 7:0 – EXTINT[7:0] External Interrupt Enable
 The bit x of EXTINT enables the interrupt associated with the EXTINTx pin.
 Writing a '0' to bit x has no effect.

Writing a '1' to bit x will set the External Interrupt Enable bit x, which enables the external interrupt EXTINTx.

Value	Description
0	The external interrupt x is disabled.
1	The external interrupt x is enabled.

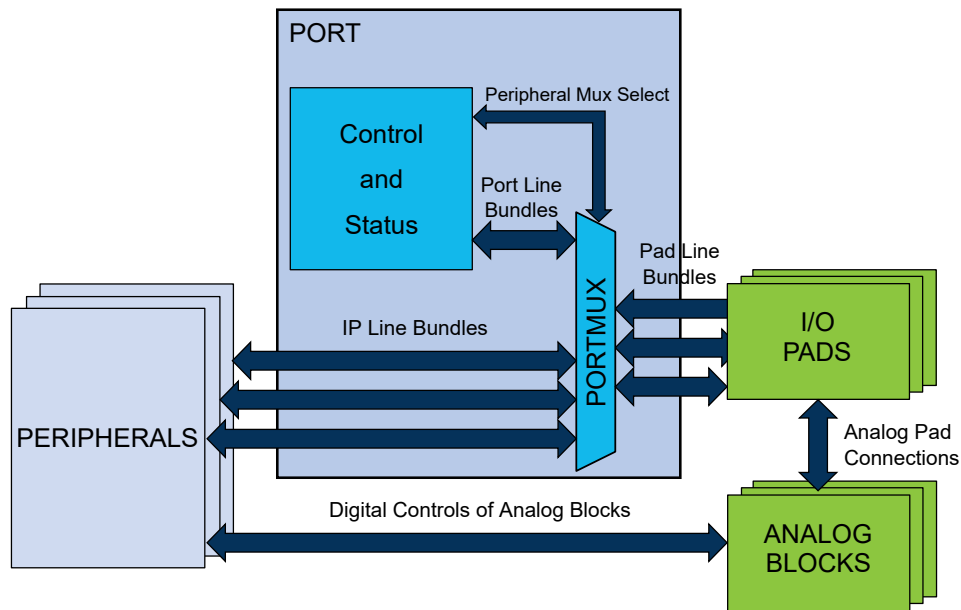
SAM L10/L11 Family

EIC – External Interrupt Controller

Value	Description
0	EXTINTx is secure.
1	EXTINTx is non-secure.

32.3 Block Diagram

Figure 32-1. PORT Block Diagram



32.4 Signal Description

Table 32-1. Signal description for PORT

Signal name	Type	Description
Pxy	Digital I/O	General-purpose I/O pin y in group x

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

32.5 Product Dependencies

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

32.5.1 I/O Lines

The I/O lines of the PORT are mapped to pins of the physical device. The following naming scheme is used:

Each line bundle with up to 32 lines is assigned an identifier 'xy', with letter x=A, B, C... and two-digit number y=00, 01, ...31. Examples: A24, C03.

PORT pins are labeled 'Pxy' accordingly, for example PA24, PC03. This identifies each pin in the device uniquely.

Each pin may be controlled by one or more peripheral multiplexer settings, which allow the pad to be routed internally to a dedicated peripheral function. When the setting is enabled, the selected peripheral has control over the output state of the pad, as well as the ability to read the current physical pad state. Refer to *I/O Multiplexing and Considerations* for details.

Each pin may be secured or non-secured, with secured pins only accessible by secure accesses.

Device-specific configurations may cause some lines (and the corresponding Pxy pin) not to be implemented.

32.5.2 Power Management

During Reset, all PORT lines are configured as inputs with input buffers, output buffers and pull disabled.

The PORT peripheral will continue operating in any sleep mode where its source clock is running.

32.5.3 Clocks

The PORT bus clock (CLK_PORT_APB) can be enabled and disabled in the Main Clock module, and the default state of CLK_PORT_APB can be found in the *Peripheral Clock Masking* section in *MCLK – Main Clock*.

The PORT requires an APB clock, which may be divided from the CPU main clock and allows the CPU to access the registers of PORT through the high-speed matrix and the AHB/APB bridge.

The priority of IOBUS accesses is higher than APB accesses. One clock cycle latency can be observed on the APB access in case of concurrent PORT accesses.

Related Links

[19. MCLK – Main Clock](#)

32.5.4 DMA

Not applicable.

32.5.5 Interrupts

Not applicable.

32.5.6 Events

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

Related Links

[33. EVSYS – Event System](#)

32.5.7 Debug Operation

When the CPU is halted in debug mode, this peripheral will continue normal operation.

32.5.8 Register Access Protection

All registers with write-access can be optionally write-protected by the Peripheral Access Controller (PAC).

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

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

Related Links

[15. PAC - Peripheral Access Controller](#)

32.5.9 Analog Connections

Analog functions are connected directly between the analog blocks and the I/O pads using analog buses. However, selecting an analog peripheral function for a given pin will disable the corresponding digital features of the pad.

Non-secure write to a bit of DIR, DIRCLR, DIRSET, DIRTGL, OUT, OUTCLR, OUTSET, or OUTTGL registers is prohibited if the corresponding bit in NONSEC is zero.

Non-secure write to a PINCFGn register or to a PMUXn register field, either directly or through a write to the WRCONFIG register, is prohibited if the corresponding bit in NONSEC is zero.

DIR, DIRCLR, DIRSET, DIRTGL, IN, OUT, OUTCLR, OUTSET, or OUTTGL bits, PINCFGn registers, or PMUXn register fields relating to secure I/O pins (i.e. the corresponding bits in NONSEC are zero), read as zero in non-secure mode, with no error resulting.

INTFLAG.NSCHK is set to 1 when NSCHK and NONSEC register values are different. Writing a 1 to INTFLAG.NSCHK will clear it and clear the PORT interrupt if enabled.

Secure code should initially write a 1 for all non-secure pins into both NONSEC and NSCHK registers. Then, whenever secure code writes a different value into NONSEC, INTFLAG.NSCHK will be set to 1 and a PORT interrupt will occur, if enabled. The non-secure code can then compare the values of the NONSEC and NSCHK registers to determine which PORT pins have just changed to secure or to non-secure. It should then copy the current NONSEC register value into NSCHK.

32.6.5 Events

The PORT allows input events to control individual I/O pins. These input events are generated by the EVSYS module and can originate from a different clock domain than the PORT module.

The PORT can perform the following actions:

- Output (OUT): I/O pin will be set when the incoming event has a high level ('1') and cleared when the incoming event has a low-level ('0').
- Set (SET): I/O pin will be set when an incoming event is detected.
- Clear (CLR): I/O pin will be cleared when an incoming event is detected.
- Toggle (TGL): I/O pin will toggle when an incoming event is detected.

The event is output to pin without any internal latency. For SET, CLEAR and TOGGLE event actions, the action will be executed up to three clock cycles after a rising edge.

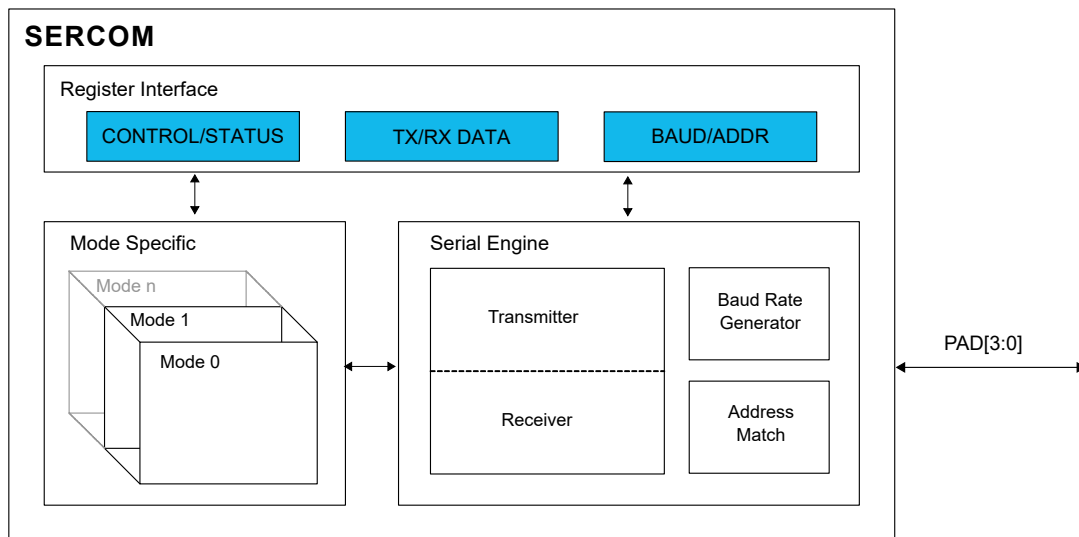
The event actions can be configured with the Event Action m bit group in the Event Input Control register (EVCTRL.EVACTm). Writing a '1' to a PORT Event Enable Input m of the Event Control register (EVCTRL.PORTEIm) enables the corresponding action on input event. Writing '0' to this bit disables the corresponding action on input event. Note that several actions can be enabled for incoming events. If several events are connected to the peripheral, any enabled action will be taken for any of the incoming events. Refer to *EVSYS – Event System*. for details on configuring the Event System.

Each event input can address one and only one I/O pin at a time. The selection of the pin is indicated by the PORT Event Pin Identifier of the Event Input Control register (EVCTR.PIDn). On the other hand, one I/O pin can be addressed by up to four different input events. To avoid action conflict on the output value of the register (OUT) of this particular I/O pin, only one action is performed according to the table below.

Note that this truth table can be applied to any SET/CLR/TGL configuration from two to four active input events.

34.3 Block Diagram

Figure 34-1. SERCOM Block Diagram



34.4 Signal Description

See the respective SERCOM mode chapters for details.

Related Links

- [35. SERCOM USART - SERCOM Synchronous and Asynchronous Receiver and Transmitter](#)
- [36. SERCOM SPI – SERCOM Serial Peripheral Interface](#)
- [37. SERCOM I2C – SERCOM Inter-Integrated Circuit](#)

34.5 Product Dependencies

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

34.5.1 I/O Lines

Using the SERCOM I/O lines requires the I/O pins to be configured using port configuration (PORT).

The SERCOM has four internal pads, PAD[3:0], and the signals from I2C, SPI and USART are routed through these SERCOM pads via a multiplexer. The configuration of the multiplexer is available from the different SERCOM modes. Refer to the mode specific chapters for details.

Related Links

- [35. SERCOM USART - SERCOM Synchronous and Asynchronous Receiver and Transmitter](#)
- [36. SERCOM SPI – SERCOM Serial Peripheral Interface](#)
- [37. SERCOM I2C – SERCOM Inter-Integrated Circuit](#)
- [32. PORT - I/O Pin Controller](#)
- [35.3 Block Diagram](#)

34.5.2 Power Management

The SERCOM can operate in any Sleep mode provided the selected clock source is running. SERCOM interrupts can be configured to wake the device from Sleep modes.

Bit 7 – CLKHOLD Clock Hold

The slave Clock Hold bit (STATUS.CLKHOLD) is set when the slave is holding the SCL line low, stretching the I2C clock. Software should consider this bit a read-only status flag that is set when INTFLAG.DRDY or INTFLAG.AMATCH is set.

This bit is automatically cleared when the corresponding interrupt is also cleared.

Bit 6 – LOWTOUT SCL Low Time-out

This bit is set if an SCL low time-out occurs.

This bit is cleared automatically if responding to a new start condition with ACK or NACK (write 3 to CTRLB.CMD) or when INTFLAG.AMATCH is cleared.

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

Writing a '1' to this bit will clear the status.

Value	Description
0	No SCL low time-out has occurred.
1	SCL low time-out has occurred.

Bit 4 – SR Repeated Start

When INTFLAG.AMATCH is raised due to an address match, SR indicates a repeated start or start condition.

This flag is only valid while the INTFLAG.AMATCH flag is one.

Value	Description
0	Start condition on last address match
1	Repeated start condition on last address match

Bit 3 – DIR Read / Write Direction

The Read/Write Direction (STATUS.DIR) bit stores the direction of the last address packet received from a master.

Value	Description
0	Master write operation is in progress.
1	Master read operation is in progress.

Bit 2 – RXNACK Received Not Acknowledge

This bit indicates whether the last data packet sent was acknowledged or not.

Value	Description
0	Master responded with ACK.
1	Master responded with NACK.

Bit 1 – COLL Transmit Collision

If set, the I2C slave was not able to transmit a high data or NACK bit, the I2C slave will immediately release the SDA and SCL lines and wait for the next packet addressed to it.

This flag is intended for the SMBus address resolution protocol (ARP). A detected collision in non-ARP situations indicates that there has been a protocol violation, and should be treated as a bus error.

38.7.2.17 Channel x Compare Buffer Value, 16-bit Mode

Name: CCBUFx
Offset: 0x30 + x*0x02 [x=0..1]
Reset: 0x0000
Property: Write-Synchronized

Bit	15	14	13	12	11	10	9	8
	CCBUF[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
	CCBUF[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 15:0 – CCBUF[15:0] Channel x Compare Buffer Value

These bits hold the value of the Channel x Compare Buffer Value. When the buffer valid flag is '1' and double buffering is enabled (CTRLBCLR.LUPD=1), the data from buffer registers will be copied into the corresponding CCx register under UPDATE condition (CTRLBSET.CMD=0x3), including the software update command.

41.8.2 Control B

Name: CTRLB
Offset: 0x01
Reset: 0x00
Property: PAC Write-Protection, Enable-Protected

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

Bits 2:0 – PRESCALER[2:0] Prescaler Configuration

This field defines the ADC clock relative to the peripheral clock.

Value	Name	Description
0x0	DIV2	Peripheral clock divided by 2
0x1	DIV4	Peripheral clock divided by 4
0x2	DIV8	Peripheral clock divided by 8
0x3	DIV16	Peripheral clock divided by 16
0x4	DIV32	Peripheral clock divided by 32
0x5	DIV64	Peripheral clock divided by 64
0x6	DIV128	Peripheral clock divided by 128
0x7	DIV256	Peripheral clock divided by 256

41.8.15 Gain Correction

Name: GAINCORR
Offset: 0x12
Reset: 0x0000
Property: PAC Write-Protection, Write-Synchronized

Bit	15	14	13	12	11	10	9	8
	GAINCORR[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
	GAINCORR[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 11:0 – GAINCORR[11:0] Gain Correction Value

If CTRL.CORREN=1, these bits define how the ADC conversion result is compensated for gain error before being written to the result register. The gain correction is a fractional value, a 1-bit integer plus an 11-bit fraction, and therefore $\frac{1}{2} \leq \text{GAINCORR} < 2$. GAINCORR values range from 0.1000000000 to 1.1111111111.

42.8.7 Status A

Name: STATUSA
Offset: 0x07
Reset: 0x00
Property: Read-Only

Bit	7	6	5	4	3	2	1	0
			WSTATE0[1:0]				STATE _x	STATE _x
Access			R	R			R	R
Reset			0	0			0	0

Bits 5:4 – WSTATE0[1:0] Window 0 Current State

These bits show the current state of the signal if the window 0 mode is enabled.

Value	Name	Description
0x0	ABOVE	Signal is above window
0x1	INSIDE	Signal is inside window
0x2	BELOW	Signal is below window
0x3		Reserved

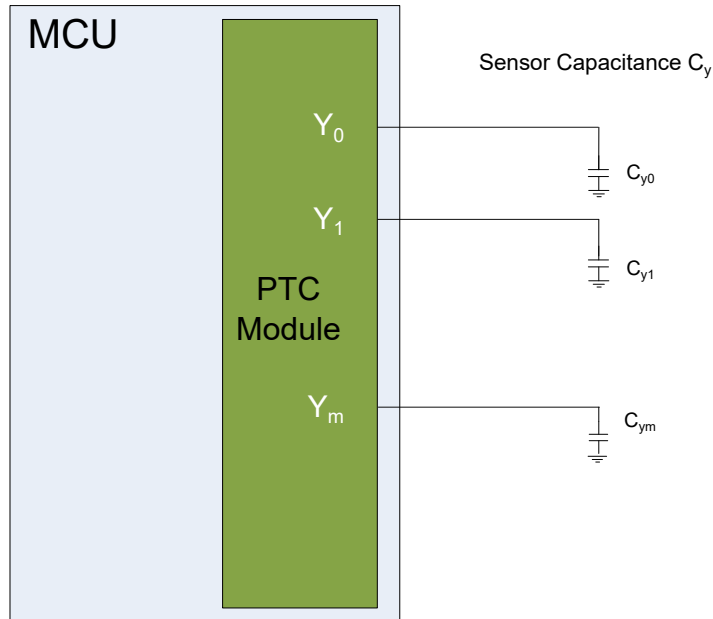
Bits 1,0 – STATE_x Comparator x Current State

This bit shows the current state of the output signal from COMP_x. STATE_x is valid only when STATUSB.READY_x is one.

45.5.1.2 Self-Capacitance Sensor Arrangement

A self-capacitance sensor is connected to a single pin on the Peripheral Touch Controller through the Y electrode for sensing the signal. The sense electrode capacitance is measured by the Peripheral Touch Controller.

Figure 45-4. Self-capacitance Sensor Arrangement



For more information about designing the touch sensor, refer to [Buttons, Sliders and Wheels Touch Sensor Design Guide](#).

45.5.2 Clocks

The PTC is clocked by the GCLK_PTC clock. The PTC operates from an asynchronous clock source and the operation is independent of the main system clock and its derivative clocks, such as the peripheral bus clock (CLK_APB). A number of clock sources can be selected as the source for the asynchronous GCLK_PTC. The clock source is selected by configuring the Generic Clock Selection ID in the Generic Clock Control register. For more information about selecting the clock sources, refer to *GCLK - Generic Clock Controller*.

The selected clock must be enabled in the Power Manager, before it can be used by the PTC. By default these clocks are disabled. The frequency range of GCLK_PTC is 400kHz to 4MHz.

Related Links

- 18. [GCLK - Generic Clock Controller](#)
- 22. [PM – Power Manager](#)

45.5.3 SAM L11 TrustZone Specific Register Access Protection

On **SAM L11** devices, this peripheral has different access permissions depending on PAC Security Attribution (Secure or Non-Secure):

- If the peripheral is configured as Non-Secure in the PAC:
 - Secure access and Non-Secure access are granted
- If the peripheral is configured as Secure in the PAC:
 - Secure access is granted
 - Non-Secure access is discarded (Write is ignored, read 0x0) and a PAC error is triggered

The Microchip Web Site

Microchip provides online support via our web site at <http://www.microchip.com/>. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Customer Change Notification Service

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at <http://www.microchip.com/>. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: <http://www.microchip.com/support>