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	25
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	1.62V ~ 3.63V
Data Converters	A/D 10x12b; D/A 1x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	32-UFQFN Exposed Pad
Supplier Device Package	32-UQFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsaml10e14a-af

Please refer to the Electrical Characteristics sections for more information.

10.1.2 Data Flash

SAM L10/L11 devices embed 2 KB of internal Data Flash with Write-While-Read (WWR) capability mapped at address 0x0040 0000.

The Data Flash can be programmed or erased while reading the Flash memory. It is not possible to read the Data Flash while writing or erasing the Flash.
Note: The Data Flash memory can be executable but requires more cycles to be read which may affect system performance.

The Data Flash cannot be cached.

The Data Flash is organized into rows, where each row contains four pages. The Data Flash has a row-erase and a page-write granularity.

Table 10-4. Data Flash Memory Parameters

Device	Memory Size [KB]	Number of Rows	Row size [Bytes]	Number of Pages	Page size [Bytes]
SAM L10/L11	2	8	256	32	64

The Data Flash is divided into one or two regions. Each region has a dedicated lock bit preventing from writing and erasing pages on it. Refer to the NVM Memory Organization figures in the *NVMCTRL* chapter to obtain the definitions of the different regions.

Note: The regions size is configured by the Boot ROM at device startup by reading the NVM Boot Configuration Row (BOCOR).

Table 10-5. Data Flash Lock Regions Parameters

Device	SAM L10	SAM L11
Number of Data FLASH Lock Regions	1	2
Regions Name	Data Flash	Data Flash Secure / Data Flash Non-Secure

10.1.3 SRAM

SAM L10/L11 devices embed 4 KB, 8 KB, or 16 KB of internal SRAM mapped at address 0x2000 0000.

Table 10-6. SRAM Memory Parameters

Device	Memory Size [KB]
SAM L11x16 / SAM L10x16 (1)	16
SAM L11x15 / SAM L10x15 (1)	8
SAM L11x14 (1)	8
SAM L10x14 (1)	4

Note:

1. x = E or D.

SRAM is composed of 4KB sub-blocks which can be retained or not in STANDBY Low-Power mode to optimize power consumption.

By default, all sub-blocks are retained, but it is possible to switch them off using the Power Manager (PM).

SRAM retention is guaranteed for Watchog, External and System Reset resets. However, the two first 2kB of SRAM are reset by the Boot ROM.

Module	Source	NVIC line
	MC 1	
	OVF	
TC1 – Timer Counter 1	ERR A	35
	MC 0	
	MC 1	
	OVF	
TC2 – Timer Counter 2	ERR A	36
	MC 0	
	MC 1	
	OVF	
ADC – Analog-to-Digital Converter	OVERRUN	37
	WINMON	
	RESRDY	38
AC – Analog Comparator	COMP 0	39
	COMP 1	
	WIN 0	
DAC – Digital-to-Analog Converter	UNDERRUN	40
	EMPTY	41
PTC – Peripheral Touch Controller	EOC	42
	WCOMP	
TRNG - True Random Number Generator	DATARDY	43
TRAM - Trust RAM	DRP	44
	ERR	

Note:

1. NSCHK interrupt sources will not generate any interrupts for SAM L10 devices.

11.3 High-Speed Bus System

11.3.1 Features

The High-Speed Bus Matrix has the following features:

- 32-bit data bus
- Allows concurrent accesses from different masters to different slaves
- Operation at a one-to-one clock frequency with the bus masters

11.3.2 Configuration

There are two master-to-slave connections to optimize system bandwidth:

- Multi-Slave Masters which are connected through the AHB bus matrix

Table 11-4. AHB Multi-Slave Masters

AHB Multi-Slave Masters
Cortex-M23 Processor
DSU - Device Service Unit
DMAC - Direct Memory Access Controller / Data Access

14.4.5.7 Random Session Key Generation (CMD_DCEK) - SAM L11 only

This command allows using a challenge-response scheme to prevent exposure of the keys in clear text on the debugger communication lines.

The different keys sent by the debugger during the Boot ROM for Chip Erase (CMD_CEx) and CRC (CMD_CRC) commands execution are:

- CRCKEY for CMD_CRC command
- CEKEYx for CMD_CEx commands

Note: The CMD_DCEK command has no effect on the SAM L10, the key derivation will not be enabled.

The random challenge value is generated using the TRNG of the device. It is generated once the CMD_DCEK is received and communicated to the debugger.

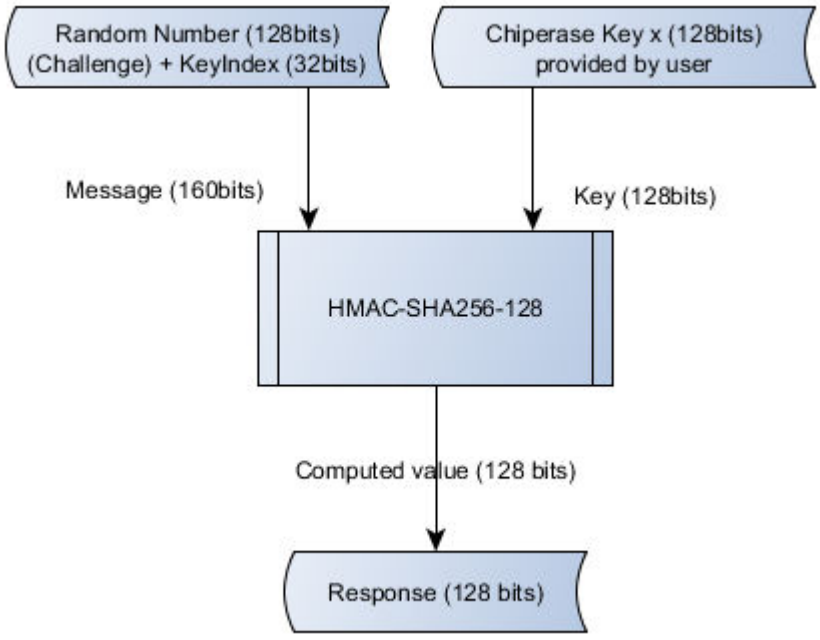
The next CMD_CEx or CMD_CRC commands will expect the key value to be replaced by the computed response corresponding to the challenge.

The challenge value is valid only for the next CMD_CEx/ CMD_CRC command.

Before sending a new CMD_CEx/ CMD_CRC command, a CMD_DCEK shall be used to re-enable the challenge-response scheme a get a new challenge value.

On the debugger side, the response shall be computed using the following algorithm:

Figure 14-13. Debugger Algorithm



Where KeyIndex is:

- 0 for ChipErase_NS
- 1 for ChipErase_S
- 2 for ChipErase_ALL
- 3 for CRC Command

Note:

Related Links

19.6.2.6 Peripheral Clock Masking

19.6.2.2 Enabling, Disabling, and Resetting

The MCLK module is always enabled and cannot be reset.

19.6.2.3 Selecting the Main Clock Source

Refer to the Generic Clock Controller description for details on how to configure the clock source of the GCLK_MAIN clock.

Refer to the Oscillators Controller (OSCCTRL) description for details on how to configure the clock source of the CLK_DFLLULP clock.

Related Links

18. GCLK - Generic Clock Controller

19.6.2.4 Selecting the Synchronous Clock Division Ratio

The main clock CLK_MAIN feeds an 8-bit prescaler, which can be used to generate the synchronous clocks. By default, the synchronous clocks run on the undivided main clock. The user can select a prescaler division for the CPU clock domain by writing the Division (DIV) bits in the CPU Clock Division register CPUDIV, resulting in a CPU clock domain frequency determined by this equation:

$$f_{CPU} = \frac{f_{main}}{CPUDIV}$$

If the application attempts to write forbidden values in CPUDIV register, registers are written but these bad values are not used and a violation is reported to the PAC module.

Division bits (DIV) can be written without halting or disabling peripheral modules. Writing DIV bits allows a new clock setting to be written to all synchronous clocks belonging to the corresponding clock domain at the same time.

19.8.6 AHB Mask

Name: AHBMASK
Offset: 0x10
Reset: 0x000001FFF
Property: PAC Write-Protection

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
				TRAM	Reserved	Reserved	Reserved	Reserved
Access				R/W	R/W	R/W	R/W	R/W
Reset				1	1	1	1	1

Bit	7	6	5	4	3	2	1	0
	NVMCTRL	PAC	Reserved	DSU	DMAC	APBC	APBB	APBA
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit 12 – TRAM TRAM AHB Clock Enable

Value	Description
0	The AHB clock for the TRAM is stopped
1	The AHB clock for the TRAM is enabled

Bit 11 – Reserved Must Be Set to 1
Bit 11 must always be set to ‘1’ when programming the AHBMASK register.

Bit 10 – Reserved Must Be Set to 1
Bit 10 must always be set to ‘1’ when programming the AHBMASK register.

Bit 9 – Reserved Must Be Set to 1
Bit 9 must always be set to ‘1’ when programming the AHBMASK register.

Bit 8 – Reserved Must Be Set to 1
Bit 8 must always be set to ‘1’ when programming the AHBMASK register.

Bit 7 – NVMCTRL NVMCTRL AHB Clock Enable

Static Power Domain Gating is a technique that allows to automatically turn off the PDSW power domain supply when not used while keeping PDAO powered up.

- SleepWalking extension to power gating (SleepWalking with dynamic power gating)
  SleepWalking is the capability for a device in Standby Sleep mode, to temporarily wake-up clocks for a peripheral to perform a task without waking-up the CPU. The SleepWalking feature has been expanded to control power gating in addition to clock gating. The power domain PDSW can be automatically controlled (active or retention state) depending on peripheral requirements (PDCFG bit from the STDBYCFG register).

The static and dynamic power gating features are fully transparent for the user.

Table 22-3. Sleep Modes versus Power Domain States Overview

	Power Domain State	
Sleep Mode	**PDSW**	**PDAO**
Active	active	active
Idle	active	active
Standby - At least one peripheral from PDSW with RUNSTDBY = 1 OR PDCFG = 1	active (1)	active
Standby - No peripheral from PDSW with RUNSTDBY = 1	retention	active
Off	off	off

Note:

1. PDSW can be switched automatically in retention mode if the dynamic power gating feature is enabled.

22.6.3.6 Regulators, RAMs, and NVM State in Sleep Mode

By default, in Standby Sleep mode, the RAMs, NVM, and regulators are automatically set in Low-Power mode to reduce power consumption:

- The RAM is in Low-Power mode if its power domain is in retention or off state.
- Non-Volatile Memory - the NVM is located in the power domain PDSW. By default, the NVM is automatically set in low power mode in these conditions:
  - When the power domain PDSW is in retention or off state.
  - When the device is in Standby Sleep mode and the NVM is not accessed. This behavior can be changed by software by configuring the SLEEPPRM bit group of the CTRLB register in the NVMCTRL peripheral.
  - When the device is in Idle Sleep mode and the NVM is not accessed. This behavior can be changed by software by configuring the SLEEPPRM bit group of the CTRLB register in the NVMCTRL peripheral.
- Regulators: by default, in Standby Sleep mode, the PM analyzes the device activity to use either the main or the low-power voltage regulator to supply the VDDCORE.

GCLK clocks, regulators and RAM are not affected in Idle Sleep mode and will operate as normal.

27.10.2 Control B in COUNT16 mode (CTRLA.MODE=1)

Name: CTRLB
Offset: 0x02
Reset: 0x0000
Property: PAC Write-Protection, Enable-Protected

Bit	15	14	13	12	11	10	9	8
	SEPTO	ACTF[2:0]				DEBF[2:0]		
Access	R/W	R/W	R/W	R/W		R/W	R/W	R/W
Reset	0	0	0	0		0	0	0

Bit	7	6	5	4	3	2	1	0
	DMAEN	RTCOUT	DEBASYNC	DEBMAJ				GP0EN
Access	R/W	R/W	R/W	R/W				R/W
Reset	0	0	0	0				0

Bit 15 – SEPTO Separate Tamper Outputs

Value	Description
0	IN[n] is compared to OUT[0] (backward-compatible).
1	IN[n] is compared to OUT[n].

Bits 14:12 – ACTF[2:0] Active Layer Frequency
These bits define the prescaling factor for the RTC clock output (OUT) used during active layer protection in terms of the CLK_RTC.

Value	Name	Description
0x0	DIV2	CLK_RTC_OUT = CLK_RTC / 2
0x1	DIV4	CLK_RTC_OUT = CLK_RTC / 4
0x2	DIV8	CLK_RTC_OUT = CLK_RTC / 8
0x3	DIV16	CLK_RTC_OUT = CLK_RTC / 16
0x4	DIV32	CLK_RTC_OUT = CLK_RTC / 32
0x5	DIV64	CLK_RTC_OUT = CLK_RTC / 64
0x6	DIV128	CLK_RTC_OUT = CLK_RTC / 128
0x7	DIV256	CLK_RTC_OUT = CLK_RTC / 256

Bits 10:8 – DEBF[2:0] Debounce Frequency
These bits define the prescaling factor for the input debouncers in terms of the CLK_RTC.

Value	Name	Description
0x0	DIV2	CLK_RTC_DEB = CLK_RTC / 2
0x1	DIV4	CLK_RTC_DEB = CLK_RTC / 4
0x2	DIV8	CLK_RTC_DEB = CLK_RTC / 8
0x3	DIV16	CLK_RTC_DEB = CLK_RTC / 16
0x4	DIV32	CLK_RTC_DEB = CLK_RTC / 32
0x5	DIV64	CLK_RTC_DEB = CLK_RTC / 64
0x6	DIV128	CLK_RTC_DEB = CLK_RTC / 128
0x7	DIV256	CLK_RTC_DEB = CLK_RTC / 256

28.8.13 Pending Channels

Name: PENDCH
Offset: 0x2C
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
	PENDCH7	PENDCH6	PENDCH5	PENDCH4	PENDCH3	PENDCH2	PENDCH1	PENDCH0
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 0, 1, 2, 3, 4, 5, 6, 7 – PENDCH Pending Channel n [n=7..0]
This bit is cleared when trigger execution defined by channel trigger action settings for DMA channel n is started, when a bus error for DMA channel n is detected or when DMA channel n is disabled. For details on trigger action settings, refer to TRIGACT bit in 28.8.19 CHCTRLB.

This bit is set when a transfer is pending on DMA channel n.

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

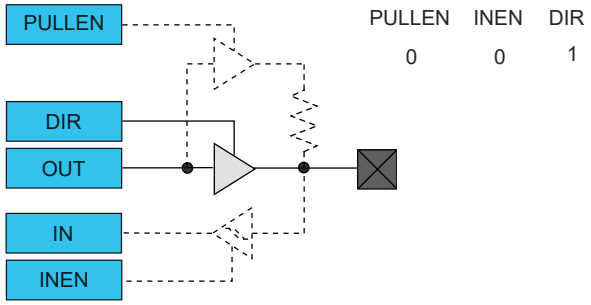


Figure 32-7. I/O Configuration - Totem-Pole Output with Enabled Input

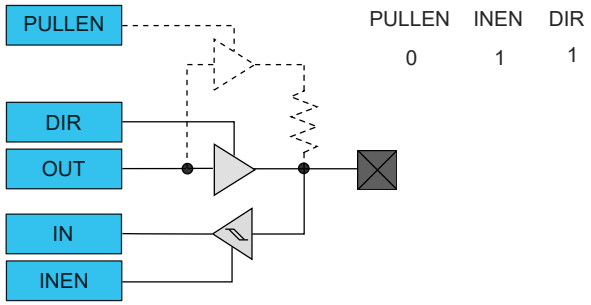
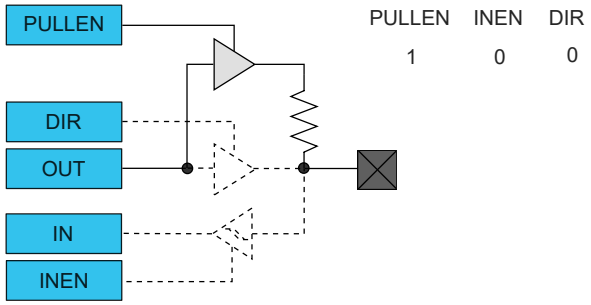


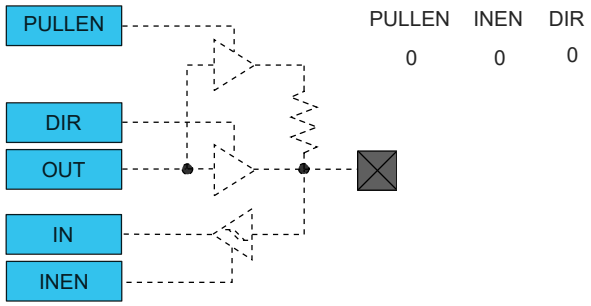
Figure 32-8. I/O Configuration - Output with Pull



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.

33.7.11 Channel n Interrupt Flag Status and Clear

Name: CHINTFLAG
Offset: 0x26 + n*0x08 [n=0..7]
Reset: 0x00
Property: Mix-Secure

Important: For **SAM L11 Non-Secure** accesses, read and write accesses (RW*) are allowed only if the security attribution for the corresponding channel (CHANNELx) is set as Non-Secured in the NONSECCHAN register.

Bit	7	6	5	4	3	2	1	0
							EVD	OVR
Access							RW/RW*/RW	RW/RW*/RW
Reset							0	0

Bit 1 – EVD Channel Event Detected
This flag is set on the next CLK_EVSYS_APB cycle when an event is being propagated through the channel, and an interrupt request will be generated if CHINTENCLR/SET.EVD is '1'.

When the event channel path is asynchronous, the EVD interrupt flag will not be set.

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

Writing a '1' to this bit will clear the Event Detected Channel interrupt flag.

Bit 0 – OVR Channel Overrun
This flag is set on the next CLK_EVSYS cycle after an overrun channel condition occurs, and an interrupt request will be generated if CHINTENCLR/SET.OVRx is '1'.

There are two possible overrun channel conditions:

- One or more of the event users on the channel are not ready when a new event occurs.
- An event happens when the previous event on channel has not yet been handled by all event users.

When the event channel path is asynchronous, the OVR interrupt flag will not be set.

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

Writing a '1' to this bit will clear the Overrun Channel interrupt flag.

Value	Description
0	NACK is transmitted when a parity error is received.
1	NACK is not transmitted when a parity error is received.

Bits 2:0 – GTIME[2:0] Guard Time
These bits define the guard time when using RS485 mode (CTRLA.FORM=0x0 or CTRLA.FORM=0x1, and CTRLA.TXPO=0x3) or ISO7816 mode (CTRLA.FORM=0x7).

For RS485 mode, the guard time is programmable from 0-7 bit times and defines the time that the transmit enable pin (TE) remains high after the last stop bit is transmitted and there is no remaining data to be transmitted.

For ISO7816 T=0 mode, the guard time is programmable from 2-9 bit times and defines the guard time between each transmitted byte.

35.8.11 Receive Error Count

Name: RXERRCNT
Offset: 0x20
Reset: 0x00
Property: Read-Synchronized

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

Bits 7:0 – RXERRCNT[7:0] Receive Error Count
This register records the total number of parity errors and NACK errors combined in ISO7816 mode (CTRLA.FORM=0x7).

This register is automatically cleared on read.

Note: The I²C standard *Fm+* (Fast-mode plus) requires a nominal high to low SCL ratio of 1:2, and BAUD should be set accordingly. At a minimum, BAUD.BAUD and/or BAUD.BAUDLOW must be non-zero.

Startup Timing The minimum time between SDA transition and SCL rising edge is 6 APB cycles when the DATA register is written in smart mode. If a greater startup time is required due to long rise times, the time between DATA write and IF clear must be controlled by software.
Note: When timing is controlled by user, the Smart Mode cannot be enabled.

Master Clock Generation (High-Speed Mode)
For I²C *Hs* transfers, there is no SCL synchronization. Instead, the SCL frequency is determined by the GCLK_SERCOMx_CORE frequency (f_{GCLK}) and the High-Speed Baud setting in the Baud register (BAUD.HSBAUD). When BAUD.HSBAUDLOW=0, the HSBAUD value will determine both SCL high and SCL low. In this case the following formula determines the SCL frequency.

$$f_{SCL} = \frac{f_{GCLK}}{2 + 2 \cdot HS\ BAUD}$$

When HSBAUDLOW is non-zero, the following formula determines the SCL frequency.

$$f_{SCL} = \frac{f_{GCLK}}{2 + HS\ BAUD + HSBAUDLOW}$$

Note: The I²C standard *Hs* (High-speed) requires a nominal high to low SCL ratio of 1:2, and HSBAUD should be set accordingly. At a minimum, BAUD.HSBAUD and/or BAUD.HSBAUDLOW must be non-zero.

37.6.2.4.2 Transmitting Address Packets

The I²C master starts a bus transaction by writing the I²C slave address to ADDR.ADDR and the direction bit, as described in 37.6.1 Principle of Operation. If the bus is busy, the I²C master will wait until the bus becomes idle before continuing the operation. When the bus is idle, the I²C master will issue a start condition on the bus. The I²C master will then transmit an address packet using the address written to ADDR.ADDR. After the address packet has been transmitted by the I²C master, one of four cases will arise according to arbitration and transfer direction.

Case 1: Arbitration lost or bus error during address packet transmission

If arbitration was lost during transmission of the address packet, the Master on Bus bit in the Interrupt Flag Status and Clear register (INTFLAG.MB) and the Arbitration Lost bit in the Status register (STATUS.ARBLOST) are both set. Serial data output to SDA is disabled, and the SCL is released, which disables clock stretching. In effect the I²C master is no longer allowed to execute any operation on the bus until the bus is idle again. A bus error will behave similarly to the arbitration lost condition. In this case, the MB interrupt flag and Master Bus Error bit in the Status register (STATUS.BUSERR) are both set in addition to STATUS.ARBLOST.

The Master Received Not Acknowledge bit in the Status register (STATUS.RXNACK) will always contain the last successfully received acknowledge or not acknowledge indication.

In this case, software will typically inform the application code of the condition and then clear the interrupt flag before exiting the interrupt routine. No other flags have to be cleared at this moment, because all flags will be cleared automatically the next time the ADDR.ADDR register is written.

Case 2: Address packet transmit complete – No ACK received

If there is no I²C slave device responding to the address packet, then the INTFLAG.MB interrupt flag and STATUS.RXNACK will be set. The clock hold is active at this point, preventing further activity on the bus.

37.8.4 Interrupt Enable Set

Name: INTENSET
Offset: 0x16
Reset: 0x00
Property: PAC Write-Protection

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 register (INTENCLR).

Bit	7	6	5	4	3	2	1	0
	ERROR					DRDY	AMATCH	PREC
Access	R/W					R/W	R/W	R/W
Reset	0					0	0	0

Bit 7 – ERROR Error Interrupt Enable
Writing '0' to this bit has no effect.

Writing '1' to this bit will set the Error Interrupt Enable bit, which enables the Error interrupt.

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

Bit 2 – DRDY Data Ready Interrupt Enable
Writing '0' to this bit has no effect.

Writing '1' to this bit will set the Data Ready bit, which enables the Data Ready interrupt.

Value	Description
0	The Data Ready interrupt is disabled.
1	The Data Ready interrupt is enabled.

Bit 1 – AMATCH Address Match Interrupt Enable
Writing '0' to this bit has no effect.

Writing '1' to this bit will set the Address Match Interrupt Enable bit, which enables the Address Match interrupt.

Value	Description
0	The Address Match interrupt is disabled.
1	The Address Match interrupt is enabled.

Bit 0 – PREC Stop Received Interrupt Enable
Writing '0' to this bit has no effect.

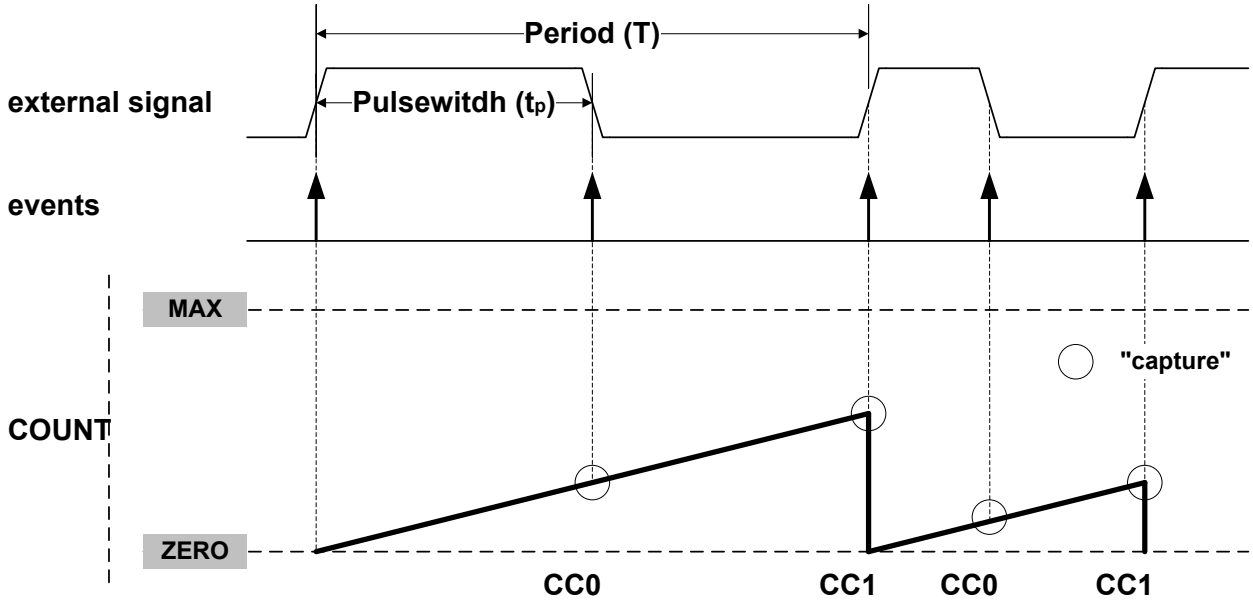
Writing '1' to this bit will set the Stop Received Interrupt Enable bit, which enables the Stop Received interrupt.

Value	Description
0	The Stop Received interrupt is disabled.
1	The Stop Received interrupt is enabled.

$$f = \frac{1}{T}$$

$$\text{dutyCycle} = \frac{t_p}{T}$$

Figure 38-13. PWP Capture



Selecting PWP in the Event Action bit group in the Event Control register (EVCTRL.EVACT) enables the TC to perform one capture action on the rising edge and the other one on the falling edge. The period *T* will be captured into CC1 and the pulse width t_p in CC0. EVCTRL.EVACT=PPW (period and pulse-width) offers identical functionality, but will capture *T* into CC0 and t_p into CC1.

The TC Event Input Invert Enable bit in the Event Control register (EVCTRL.TCINV) is used to select whether the wraparound should occur on the rising edge or the falling edge. If EVCTRL.TCINV=1, the wraparound will happen on the falling edge. In case pin capture is enabled, this can also be achieved by modifying the value of the DRVCTRL.INVENx bit.

The TC can detect capture overflow of the input capture channels: When a new capture event is detected while the Capture Interrupt flag (INTFLAG.MCx) is still set, the new timestamp will not be stored and INTFLAG.ERR will be set.

Note: The corresponding capture is working only if the channel is enabled in capture mode (CTRLA.CAPTENx=1). If not, the capture action is ignored and the channel is enabled in compare mode of operation. Consequently, both channels must be enabled in order to fully characterize the input.

38.6.2.8.3 Pulse-Width Capture Action

The TC performs the input capture on the falling edge of the input signal. When the edge is detected, the counter value is cleared and the TC stops counting. When a rising edge is detected on the input signal, the counter restarts the counting operation. To enable the operation on opposite edges, the input signal to capture must be inverted (refer to DRVCTRL.INVEN or EVCTRL.TCEINV).

39. TRNG – True Random Number Generator

39.1 Overview

The True Random Number Generator (TRNG) generates unpredictable random numbers that are not generated by an algorithm. It passes the American NIST Special Publication 800-22 and Diehard Random Tests Suites.

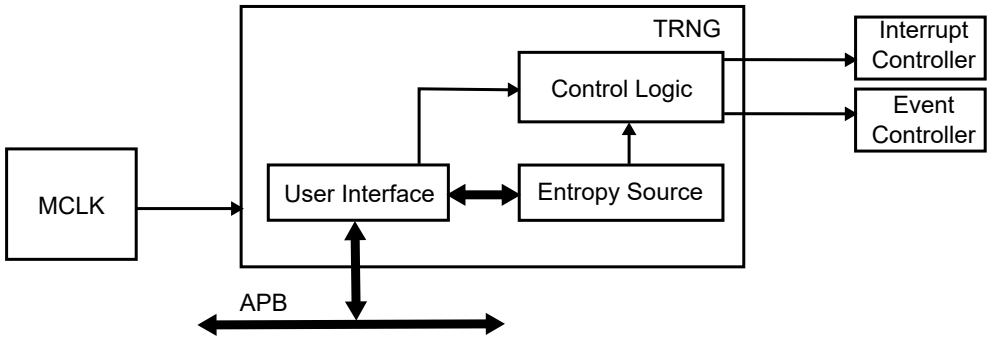
The TRNG may be used as an entropy source for seeding an NIST approved DRNG (Deterministic RNG) as required by FIPS PUB 140-2 and 140-3.

39.2 Features

- Passed NIST Special Publication 800-22 Tests Suite
- Passed Diehard Random Tests Suite
- May be used as Entropy Source for seeding an NIST approved DRNG (Deterministic RNG) as required by FIPS PUB 140-2 and 140-3
- Provides a 32-bit random number every 84 clock cycles

39.3 Block Diagram

Figure 39-1. TRNG Block Diagram.



39.4 Signal Description

Not applicable.

39.5 Product Dependencies

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

39.5.1 I/O Lines

Not applicable.

39.5.2 Power Management

The functioning of TRNG depends on the sleep mode of device.

43.7 Register Summary

Offset	Name	Bit Pos.								
0x00	CTRLA	7:0		RUNSTDBY					ENABLE	SWRST
0x01	CTRLB	7:0	REFSEL[1:0]		DITHER		VPD	LEFTADJ	IOEN	EOEN
0x02	EVCTRL	7:0						INVEI	EMPTYEO	STARTEI
0x03	Reserved									
0x04	INTENCLR	7:0							EMPTY	UNDERRUN
0x05	INTENSET	7:0							EMPTY	UNDERRUN
0x06	INTFLAG	7:0							EMPTY	UNDERRUN
0x07	STATUS	7:0								READY
0x08	DATA	7:0	DATA[7:0]							
		15:8	DATA[15:8]							
0x0A ... 0x0B	Reserved									
0x0C	DATABUF	7:0	DATABUF[7:0]							
		15:8	DATABUF[15:8]							
0x0E ... 0x0F	Reserved									
0x10	SYNCBUSY	7:0					DATABUF	DATA	ENABLE	SWRST
		15:8								
		23:16								
		31:24								
0x14 ... 0x17	Reserved									
0x18	DBGCTRL	7:0								DBGRUN

43.8 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 43.5.8 Register Access Protection.

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 43.6.7 Synchronization.

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.

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

46.7 Power Consumption

The values in this section are measured values of power consumption under the following conditions, except where noted:

- **Operating Conditions**
  - V_{DDIO} = 3.3V or 1.8V
  - CPU is running on Flash with required Wait states, as recommended in the NVM Characteristics section
  - Low-power cache is enabled
  - BOD33 is disabled
  - I/Os are configured with digital input trigger disabled (default Reset configuration)
- **Oscillators**
  - XOSC (crystal oscillator) stopped
  - XOSC32K (32.768 kHz crystal oscillator) running with external 32.768 kHz crystal
  - When in Active mode with Performance Level 2 (PL2), DPLL is running at 32 MHz and using XOSC32K as reference
  - When in Active mode on DFLLULP, the DFLLULP is configured in Closed Loop mode using XOSC32K as reference clock and MCLK.CTRLA.CKSEL = 1

Table 46-8. Active Current Consumption

Mode	Conditions	Regulator	PL	CPU Clock	Vcc	Ta	Typ.	Max	Units
ACTIVE	COREMARK/ FIBONACCI	LDO	PL0	DFLLULP at 8MHz	1.8V	Max at 85°C Typ at 25°	64.1	82	µA/Mhz
					3.3V		64.4	84	
				OSC 8MHz	1.8V		66.6	81	
					3.3V		70.3	83	
				OSC 4MHz	1.8V		74.1	102	
					3.3V		77.8	106	
			PL2	FDPLL96M at 32MHz	1.8V		82.0	89	
					3.3V		82.5	89	
				DFLLULP at 32MHz	1.8V		75.8	99	
					3.3V		75.8	96	
		BUCK	PL0	DFLLULP at 8MHz	1.8V		40.0	53	
					3.3V		25.3	34	
				OSC 8MHz	1.8V		43.8	53	
					3.3V		32.1	39	
				OSC 4MHz	1.8V		50.3	68	
					3.3V		38.9	52	
			PL2	FDPLL96M at 32MHz	1.8V		59.9	66	