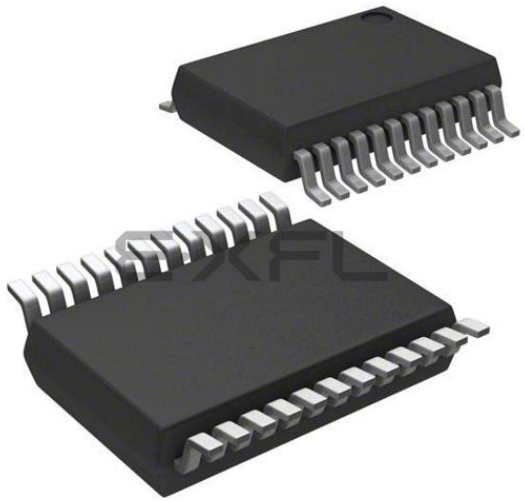


Welcome to E-XFL.COM

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	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 5x12b; D/A 1x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°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/atsaml10d14a-yf

Important: SRAM retention is not guaranteed after Power Supply Resets (POR, BOD12 and BOD33).

10.1.4 Boot ROM

SAM L10/L11 devices embed 8 KB of internal ROM mapped at address 0x0200 0000.

Note: Please refer to 14. Boot ROM for more details.

10.2 NVM Rows

SAM L10 and SAM L11 have different Non Volatile Memory (NVM) rows which contain device configuration data that can be used by the system:

Table 10-7. NVM Rows Mapping

NVM Rows	Address
User Row (UROW)	0x00804000
Software Calibration Row	0x00806020
Temperature Log Row	0x00806038
Boot Configuration Row (BOCOR)	0x0080C000

10.2.1 NVM User Row (UROW)

The Non Volatile Memory User Row (UROW) contains device configuration data that are automatically read at device power-on.

This row can be updated using the NVMCTRL peripheral.

When writing to the NVM User Row, the new values are not loaded by the other peripherals on the device until a device reset occurs.

The NVM User Row can be read at the address 0x00804000.

SAM L10 and SAM L11 have different NVM User Row mappings.

Related Links

30. NVMCTRL – Nonvolatile Memory Controller
25. SUPC – Supply Controller
25.8.5 BOD33
26. WDT – Watchdog Timer
26.8.1 CTRLA
26.8.2 CONFIG
26.8.3 EWCTRL

10.2.1.1 SAM L10 User Row

Table 10-8. SAM L10 UROW Bitfields Definition

Bit Pos.	Name	Usage	Factory Setting	Related Peripheral Register
2:0	Reserved	Reserved	Reserved	Reserved
5:3	NSULCK	NVM UnLock Bits	0x7	NVMCTRL.NSULCK

Bit 8 – WDT Peripheral WDT Write Protection Status

Bit 7 – GCLK Peripheral GCLK Write Protection Status

Bit 6 – SUPC Peripheral SUPC Write Protection Status

Bit 5 – OSC32KCTRL Peripheral OSC32KCTRL Write Protection Status

Bit 4 – OSCCTRL Peripheral OSCCTRL Write Protection Status

Bit 3 – RSTC Peripheral RSTC Write Protection Status

Bit 2 – MCLK Peripheral MCLK Write Protection Status

Bit 1 – PM Peripheral PM Write Protection Status

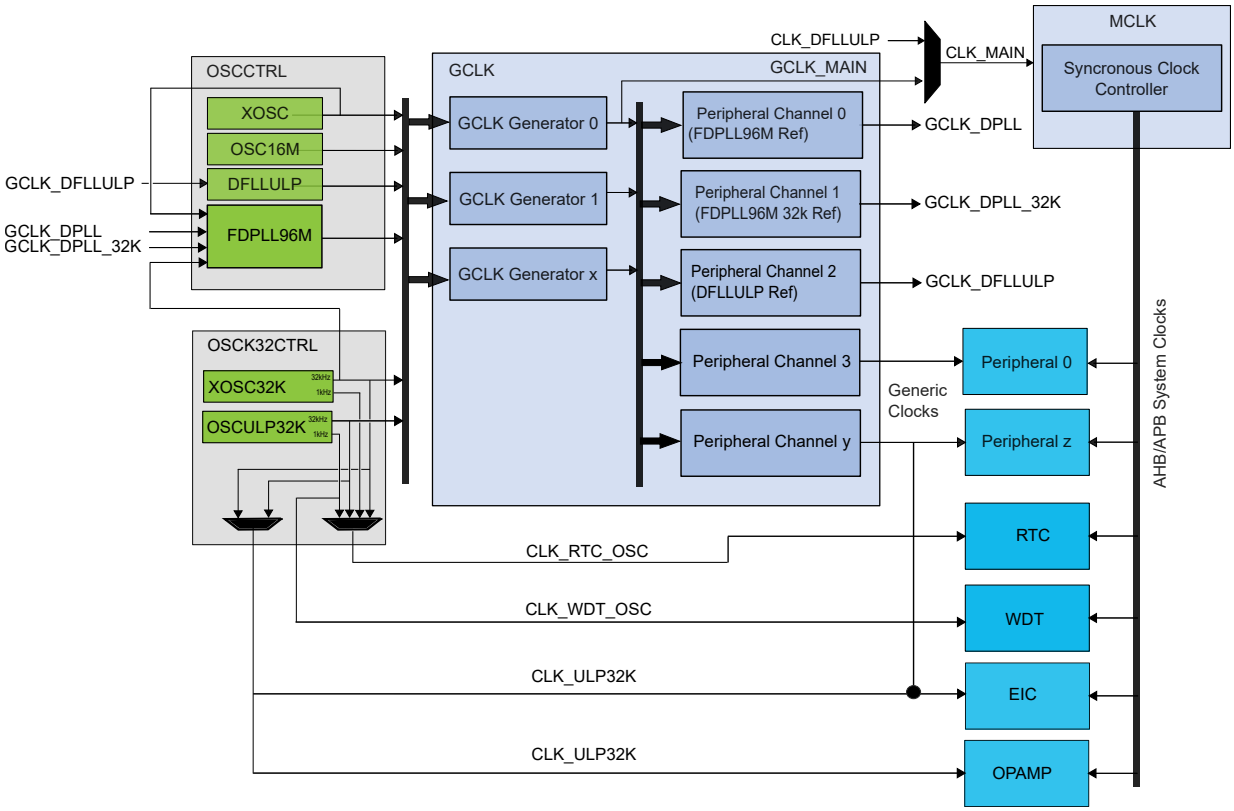
Bit 0 – PAC Peripheral PAC Write Protection Status

17. Clock System

This chapter summarizes the clock distribution and terminology in the SAM L10/L11 device. This document will not explain every detail of its configuration, hence for in-depth details, refer to the respective peripherals descriptions and the *Generic Clock* documentation.

17.1 Clock Distribution

Figure 17-1. Clock Distribution



The SAM L10/L11 clock system consists of these features:

- *Clock sources*, that is oscillators controlled by OSCCTRL and OSC32KCTRL
  - A clock source provides a time base that is used by other components, such as Generic Clock Generators. Example clock sources are the internal 16MHz oscillator (OSC16M), external crystal oscillator (XOSC) and the Fractional Digital Phase Locked Loop (FDPLL96M).
- *Generic Clock Controller (GCLK),* which generates, controls and distributes the asynchronous clock consisting of:
  - *Generic Clock Generators:* These are programmable prescalers that can use any of the system clock sources as a time base. The Generic Clock Generator 0 generates the clock signal GCLK_MAIN, which is used by the Power Manager and the Main Clock (MCLK) module, which in turn generates synchronous clocks.
  - *Generic Clocks:* These are clock signals generated by Generic Clock Generators and output by the Peripheral Channels, and serve as clocks for the peripherals of the system. Multiple instances of a peripheral will typically have a separate Generic Clock for each instance.

23.8.5 Status

Name: STATUS
Offset: 0x10
Reset: 0x00000100
Property: -

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
					DPLLLDRTO	DPLLLTO	DPLLLCKF	DPLLLCKR
Access					R	R	R	R
Reset					0	0	0	0

Bit	15	14	13	12	11	10	9	8
						DFLLULPNOLOCK	DFLLULPLOCK	DFLLULPRDY
Access						R	R	R
Reset						0	0	1

Bit	7	6	5	4	3	2	1	0
				OSC16MRDY		CLKSW	CLKFAIL	XOSCRDY
Access				R		R	R	R
Reset				0		0	0	0

Bit 19 – DPLLLDRTO DPLL Loop Divider Ratio Update Complete

Value	Description
0	DPLL Loop Divider Ratio Update Complete not detected.
1	DPLL Loop Divider Ratio Update Complete detected.

Bit 18 – DPLLLTO DPLL Lock Timeout

Value	Description
0	DPLL Lock time-out not detected.
1	DPLL Lock time-out detected.

Bit 17 – DPLLLCKF DPLL Lock Fall

Value	Description
0	DPLL Lock fall edge not detected.
1	DPLL Lock fall edge detected.

Bit 16 – DPLLLCKR DPLL Lock Rise

23.8.20 DPLL Status

Name: DPLLSTATUS
Offset: 0x40
Reset: 0x00
Property: –

Bit	7	6	5	4	3	2	1	0
							CLKRDY	LOCK
Access							R	R
Reset							0	0

Bit 1 – CLKRDY DPLL Clock Ready

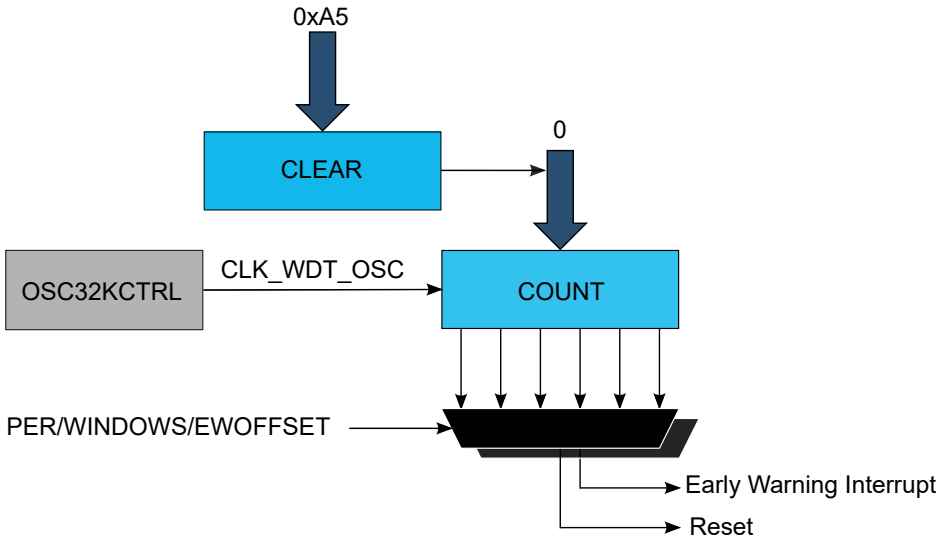
Value	Description
0	The DPLL output clock is off.
1	The DPLL output clock in on.

Bit 0 – LOCK DPLL Lock

Value	Description
0	The DPLL Lock signal is cleared, when the DPLL is disabled or when the DPLL is trying to reach the target frequency.
1	The DPLL Lock signal is asserted when the desired frequency is reached.

26.3 Block Diagram

Figure 26-1. WDT Block Diagram



26.4 Signal Description

Not applicable.

26.5 Product Dependencies

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

26.5.1 I/O Lines

Not applicable.

26.5.2 Power Management

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

26.5.3 Clocks

The WDT bus clock (CLK_WDT_APB) can be enabled and disabled (masked) in the Main Clock module (MCLK).

A 1.024 kHz oscillator clock (CLK_WDT_OSC) is required to clock the WDT internal counter.

The CLK_WDT_OSC CLOCK is sourced from the clock of the internal Ultra Low-Power Oscillator (OSCULP32K). Due to ultra low-power design, the oscillator is not accurate, hence the exact time-out period may vary from device-to-device. This variation must be considered when designing software that uses the WDT to ensure that the time-out periods used are valid for all devices.

27.12.5 Interrupt Enable Set in Clock/Calendar mode (CTRLA.MODE=2)

Name: INTENSET
Offset: 0x0A
Reset: 0x0000
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 (INTENCLR) 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 set the Overflow Interrupt Enable bit, which enables the Overflow interrupt.

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

Bit 14 – TAMPER Tamper Interrupt Enable
Writing a '0' to this bit has no effect. Writing a '1' to this bit will set the Tamper Interrupt Enable bit, which enables the Tamper interrupt.

Value	Description
0	The Tamper interrupt it disabled.
1	The Tamper interrupt is enabled.

Bit 8 – ALARM0 Alarm 0 Interrupt Enable
Writing a '0' to this bit has no effect. Writing a '1' to this bit will set the Alarm 0 Interrupt Enable bit, which enables the Alarm 0 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 set the Periodic Interval n Interrupt Enable bit, which enables the Periodic Interval n interrupt.

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

Memory Region	Secure Access	Non-Secure Access	Limitations
FLASH Application non-secure	Y	Y	No if NSULCK.ANS=0
Data FLASH secure	Y	N	No if SULCK.DS=0
Data FLASH non-secure	Y	Y	No if NSULCK.DNS=0
AUX FLASH User Row (UROW)	Y	N	No if BOCOR.URWEN=0
AUX FLASH Boot Configuration (BOCOR)	Y	N	No if BOCOR.BCWEN=0

The NSULCK SULCK bitfields in the user row define respectively the NSULCK and SULCK register default value after a reset.

Special care must be taken when sharing the NVMCTRL between the secure and non-secure domains. When the secure code modifies the NVM it is highly recommended that it disables all write accesses to the APB non-secure alias and writes to AHB non-secure regions by writing a 0 to NONSEC.WRITE. This avoids any interference with non-secure modify operations. Note that in this case even a secure application cannot write the page buffer at a non-secure location since the IDAU changes security attributions of Non-Secure transactions to Non-Secure regions to Non-Secure.

The NONSEC.WRITE reset value is '1', meaning that it is always possible to program a Non-Secure FLASH or Data FLASH region after a debugger probe cold-plugging. But if the debugger connects with the hot-plugging procedure then NONSEC.WRITE must be '1' to let the debugger program Non-Secure regions otherwise the transaction will cause a hardfault (seen as a DAP fault at DAP level).

For applications that don't require Non-Secure regions programming other than from a secure code, it is recommended to always disable Non-Secure writes by disabling NONSEC.WRITE. When disabled secure code needs to enable it to be able to modify Non-Secure regions following this procedure:

- disable interrupts
- write a one to NONSEC.WRITE to allow writes to the non-secure region
- write the page buffer
- write a zero to NONSEC.WRITE
- enable again the interrupts

If the NSCHK interrupt is enabled, a NONSEC.WRITE modification will generate an interrupt so that the non-secure world is aware of this change. Depending on NSCHK.WRITE INTFLAG.NSCHK will rise upon a rising or falling NONSEC.WRITE transition. The interrupt can be configured as secure or non-secure in the NVIC. If secure then a software mechanism can be implemented to call a non-secure NVMCTRL IRQ handler from the secure world.

The NVMCTRL monitors the Page Buffer write accesses and accepts only writes to non-secure regions when the transaction is non-secure. Moreover it checks that any write to the page buffer is in the same page as the previous write when the Page Buffer is not empty. When this check fails, an error is returned to the bus master that initiated the transaction. This ensures that it is not possible to mix different page writes into the Page Buffer. Therefore, any Page Buffer write access must at some point be followed by a manual or automatic Write Page (WP) that automatically clears the page buffer or a Clear Page Buffer (PBC) command.

33.7.5 Interrupt Status

Name: INTSTATUS
Offset: 0x14
Reset: 0x00000000
Property: Mix-Secure

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

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

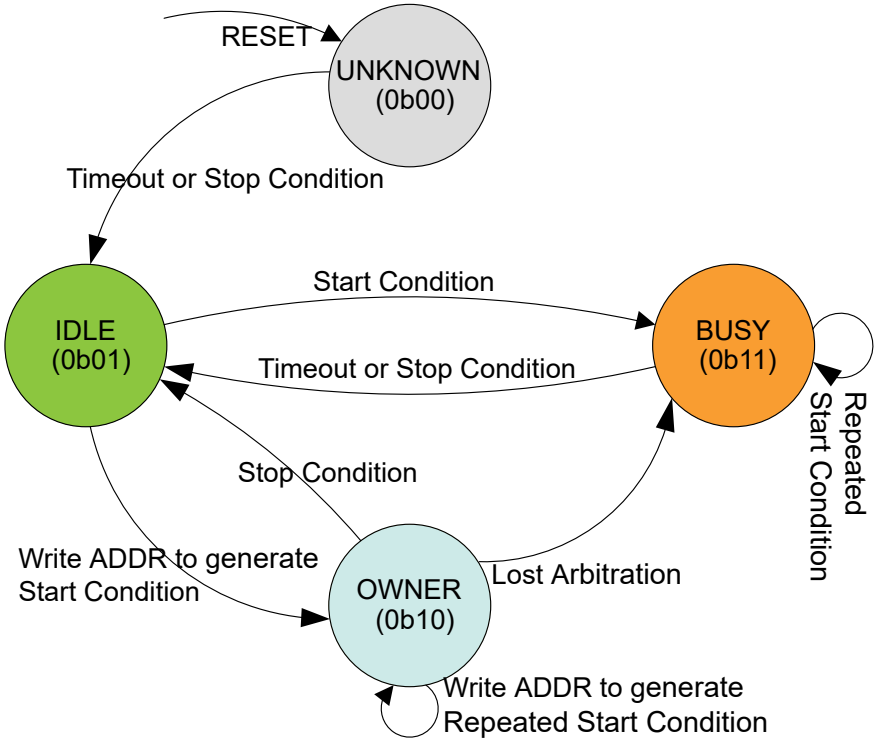
Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
					CHINT3	CHINT2	CHINT1	CHINT0
Access					R/R*/R	R/R*/R	R/R*/R	R/R*/R
Reset					0	0	0	0

Bits 0, 1, 2, 3 – CHINT Channel x Pending Interrupt
This bit is set when Channel x has a pending interrupt.

This bit is cleared when the corresponding Channel x interrupts are disabled, or the source interrupt sources are cleared.

Figure 37-4. Bus State Diagram



The bus state machine is active when the I²C master is enabled.

After the I²C master has been enabled, the bus state is UNKNOWN (0b00). From the UNKNOWN state, the bus will transition to IDLE (0b01) by either:

- Forcing by writing 0b01 to STATUS.BUSSTATE
- A stop condition is detected on the bus
- If the inactive bus time-out is configured for SMBus compatibility (CTRLA.INACTOUT) and a time-out occurs.

Note: Once a known bus state is established, the bus state logic will not re-enter the UNKNOWN state.

When the bus is IDLE it is ready for a new transaction. If a start condition is issued on the bus by another I²C master in a multi-master setup, the bus becomes BUSY (0b11). The bus will re-enter IDLE either when a stop condition is detected, or when a time-out occurs (inactive bus time-out needs to be configured).

If a start condition is generated internally by writing the Address bit group in the Address register (ADDR.ADDR) while IDLE, the OWNER state (0b10) is entered. If the complete transaction was performed without interference, i.e., arbitration was not lost, the I²C master can issue a stop condition, which will change the bus state back to IDLE.

However, if a packet collision is detected while in OWNER state, the arbitration is assumed lost and the bus state becomes BUSY until a stop condition is detected. A repeated start condition will change the bus state only if arbitration is lost while issuing a repeated start.

Note: Violating the protocol may cause the I²C to hang. If this happens it is possible to recover from this state by a software reset (CTRLA.SWRST='1').

Related Links

37.10.1 CTRLA

37.6.2.4.4 Receiving Data Packets (SCLSM=0)

When INTFLAG.SB is set, the I²C master will already have received one data packet. The I²C master must respond by sending either an ACK or NACK. Sending a NACK may be unsuccessful when arbitration is lost during the transmission. In this case, a lost arbitration will prevent setting INTFLAG.SB. Instead, INTFLAG.MB will indicate a change in arbitration. Handling of lost arbitration is the same as for data bit transmission.

37.6.2.4.5 Receiving Data Packets (SCLSM=1)

When INTFLAG.SB is set, the I²C master will already have received one data packet and transmitted an ACK or NACK, depending on CTRLB.ACKACT. At this point, CTRLB.ACKACT must be set to the correct value for the next ACK bit, and the transaction can continue by reading DATA and issuing a command if not in the smart mode.

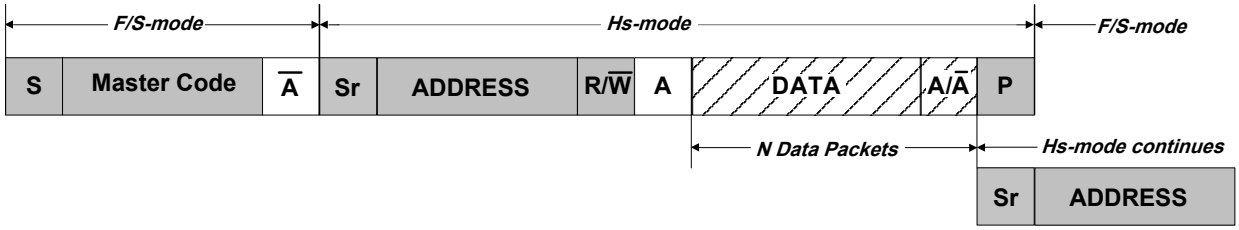
37.6.2.4.6 High-Speed Mode

High-speed transfers are a multi-step process, see High Speed Transfer.

First, a master code (0b00001nnn, where 'nnn' is a unique master code) is transmitted in Full-speed mode, followed by a NACK since no slaveshould acknowledge. Arbitration is performed only during the Full-speed Master Code phase. The master code is transmitted by writing the master code to the address register (ADDR.ADDR) and writing the high-speed bit (ADDR.HS) to '0'.

After the master code and NACK have been transmitted, the master write interrupt will be asserted. In the meanwhile, the slave address can be written to the ADDR.ADDR register together with ADDR.HS=1. Now in High-speed mode, the master will generate a repeated start, followed by the slave address with RW-direction. The bus will remain in High-speed mode until a stop is generated. If a repeated start is desired, the ADDR.HS bit must again be written to '1', along with the new address ADDR.ADDR to be transmitted.

Figure 37-8. High Speed Transfer



Transmitting in High-speed mode requires the I²C master to be configured in High-speed mode (CTRLA.SPEED=0x2) and the SCL clock stretch mode (CTRLA.SCLSM) bit set to '1'.

37.6.2.4.7 10-Bit Addressing

When 10-bit addressing is enabled by the Ten Bit Addressing Enable bit in the Address register (ADDR.TENBITEN=1) and the Address bit field ADDR.ADDR is written, the two address bytes will be transmitted, see 10-bit Address Transmission for a Read Transaction. The addressed slave acknowledges the two address bytes, and the transaction continues. Regardless of whether the transaction is a read or write, the master must start by sending the 10-bit address with the direction bit (ADDR.ADDR[0]) being zero.

If the master receives a NACK after the first byte, the write interrupt flag will be raised and the STATUS.RXNACK bit will be set. If the first byte is acknowledged by one or more slaves, then the master will proceed to transmit the second address byte and the master will first see the write interrupt flag after the second byte is transmitted. If the transaction direction is read-from-slave, the 10-bit address transmission must be followed by a repeated start and the first 7 bits of the address with the read/write bit equal to '1'.

37.10.9 Address

Name: ADDR
Offset: 0x24
Reset: 0x0000
Property: Write-Synchronized

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	LEN[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	TENBITEN	HS	LENEN			ADDR[10:8]		
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
	ADDR[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 23:16 – LEN[7:0] Transaction Length
These bits define the transaction length of a DMA transaction from 0 to 255 bytes. The Transfer Length Enable (LENEN) bit must be written to '1' in order to use DMA.

Bit 15 – TENBITEN Ten Bit Addressing Enable
This bit enables 10-bit addressing. This bit can be written simultaneously with ADDR to indicate a 10-bit or 7-bit address transmission.

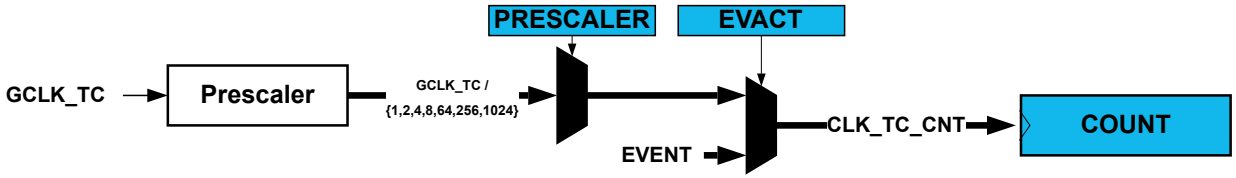
Value	Description
0	10-bit addressing disabled.
1	10-bit addressing enabled.

Bit 14 – HS High Speed
This bit enables High-speed mode for the current transfer from repeated START to STOP. This bit can be written simultaneously with ADDR for a high speed transfer.

Value	Description
0	High-speed transfer disabled.
1	High-speed transfer enabled.

Bit 13 – LENEN Transfer Length Enable

Figure 38-2. Prescaler



38.6.2.4 Counter Mode

The counter mode is selected by the Mode bit group in the Control A register (CTRLA.MODE). By default, the counter is enabled in the 16-bit counter resolution. Three counter resolutions are available:

- COUNT8: The 8-bit TC has its own Period Value and Period Buffer Value registers (PER and PERBUF).
- COUNT16: 16-bit is the default counter mode. There is no dedicated period register in this mode.
- COUNT32: This mode is achieved by pairing two 16-bit TC peripherals. TCn is paired with TCn+1. TC2 does not support 32-bit resolution.
  When paired, the TC peripherals are configured using the registers of the even-numbered TC. The odd-numbered partner will act as a slave, and the Slave bit in the Status register (STATUS.SLAVE) will be set. The register values of a slave will not reflect the registers of the 32-bit counter. Writing to any of the slave registers will not affect the 32-bit counter. Normal access to the slave COUNT and CCx registers is not allowed.

38.6.2.5 Counter Operations

Depending on the mode of operation, the counter is cleared, reloaded, incremented, or decremented at each TC clock input (CLK_TC_CNT). A counter clear or reload marks the end of the current counter cycle and the start of a new one.

The counting direction is set by the Direction bit in the Control B register (CTRLB.DIR). If this bit is zero the counter is counting up, and counting down if CTRLB.DIR=1. The counter will count up or down for each tick (clock or event) until it reaches TOP or ZERO. When it is counting up and TOP is reached, the counter will be set to zero at the next tick (overflow) and the Overflow Interrupt Flag in the Interrupt Flag Status and Clear register (INTFLAG.OVF) will be set. When it is counting down, the counter is reloaded with the TOP value when ZERO is reached (underflow), and INTFLAG.OVF is set.

INTFLAG.OVF can be used to trigger an interrupt, a DMA request, or an event. An overflow/underflow occurrence (i.e., a compare match with TOP/ZERO) will stop counting if the One-Shot bit in the Control B register is set (CTRLBSET.ONESHOT).

It is possible to change the counter value (by writing directly in the COUNT register) even when the counter is running. When starting the TC, the COUNT value will be either ZERO or TOP (depending on the counting direction set by CTRLBSET.DIR or CTRLBCLR.DIR), unless a different value has been written to it, or the TC has been stopped at a value other than ZERO. The write access has higher priority than count, clear, or reload. The direction of the counter can also be changed when the counter is running. See also the following figure.

39.8.5 Interrupt Flag Status and Clear

Name: INTFLAG
Offset: 0x0A
Reset: 0x00
Property: -

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

Bit 0 – DATARDY Data Ready
This flag is set when a new random value is generated, and an interrupt will be generated if INTENCLR/SET.DATARDY=1.

This flag is cleared by writing a '1' to the flag or by reading the DATA register.

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

42.8.13 Synchronization Busy

Name: SYNCBUSY
Offset: 0x20
Reset: 0x00000000
Property: Read-Only

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
				COMPCTRLx	COMPCTRLx	WINCTRL	ENABLE	SWRST
Access				R	R	R	R	R
Reset				0	0	0	0	0

Bits 4,3 – COMPCTRLx COMPCTRLx Synchronization Busy
This bit is cleared when the synchronization of the COMPCTRLx register between the clock domains is complete.

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

Bit 2 – WINCTRL WINCTRL Synchronization Busy
This bit is cleared when the synchronization of the WINCTRL register between the clock domains is complete.

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

Bit 1 – ENABLE Enable Synchronization Busy
This bit is cleared when the synchronization of the CTRLA.ENABLE bit between the clock domains is complete.

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

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

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

43.8.1 Control A

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

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

Bit 6 – RUNSTDBY Run in Standby
This bit is not synchronized

Value	Description
0	The DAC output buffer is disabled in standby sleep mode.
1	The DAC output buffer can be enabled in standby sleep mode.

Bit 1 – ENABLE Enable DAC Controller
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 corresponding bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.

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

Bit 0 – SWRST Software Reset
Writing '0' to this bit has no effect.

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

Writing a '1' to CTRLA.SWRST 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.

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

43.8.11 Debug Control

Name: DBGCTRL
Offset: 0x18
Reset: 0x00
Property: PAC Write-Protection

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

Bit 0 – DBGRUN Debug Run
This bit is not reset by a software reset.

This bits controls the functionality when the CPU is halted by an external debugger.

Value	Description
0	The DAC is halted when the CPU is halted by an external debugger. Any ongoing conversion will complete.
1	The DAC continues normal operation when the CPU is halted by an external debugger.

Table 47-2. Active Current Consumption

Mode	Conditions	Regulator	PL	CPU Clock	Vcc	Ta	Typ.	Max.	Units
ACTIVE	COREMARK / FIBONACCI	LDO	PL0	DFLLUP at 8 MHz	1.8V	Max at 125°C Typ at 25°C	64.1	129	uA/MHz
					3.3V		64.4	131	
				OSC 8 MHz	1.8V		66.6	130	
					3.3V		70.3	132	
				OSC 4 MHz	1.8V		74.1	203	
					3.3V		77.8	206	
			PL2	FDPLL96 at 32 MHz	1.8V		82.0	98	
					3.3V		82.5	99	
				DFLLULP at 32 MHz	1.8V		75.8	109	
					3.3V		75.8	107	
		BUCK	PL0	DFLLUP at 8 MHz	1.8V		40.0	84	
					3.3V		25.3	54	
				OSC 8 MHz	1.8V		43.8	84	
					3.3V		32.1	58	
				OSC 4 MHz	1.8V		50.3	131	
					3.3V		38.9	92	
			PL2	FDPLL96 at 32 MHz	1.8V		59.9	70	
					3.3V		35.3	43	
				DFLLULP at 32 MHz	1.8V		55.3	78	
					3.3V		32.6	46	
	WHILE1	LDO	PL0	DFLLUP at 8 MHz	1.8V		44.3	110	
					3.3V		44.4	111	
				OSC 8 MHz	1.8V		47.6	111	
					3.3V		50.1	113	
				OSC 4 MHz	1.8V		54.6	184	
					3.3V		57.7	187	
			PL2	FDPLL96 at 32 MHz	1.8V		56.9	79	
					3.3V		57.2	80	
				DFLLULP at 32 MHz	1.8V		50.8	72	
					3.3V		51.0	72	

- CPU is running on Flash with 0 wait states, at 4MHz
- PTC running at 4MHz
- Voltage Regulator mode: LPEFF enabled

PTC Configuration

- Mutual-Capacitance mode
- One touch channel

System Configuration

- Standby Sleep mode enabled
- RTC running on OSCULP32K: used to define the PTC scan rate, through the event system
- Drift Calibration disabled: no interrupts, PTC scans are performed in Standby mode
- Drift Calibration enabled: RTC interrupts (wake-up) the CPU to perform PTC scans. PTC drift calibration is performed every 1.5 sec.

Table 47-17. Power Consumption (1)

Symbol	Parameters	Drift Calibration	PTC scan rate (msec)	Oversamples	Ta	Typ.	Max.	Units
IDD	Current Consumption	Disabled	10	4	Max 125°C Typ 25°C	6.2	292.0	µA
				16		12.7	300.5	
			50	4		2.3	286.1	
				16		3.7	290.3	
			100	4		1.7	286.1	
				16		2.4	286.2	
			200	4		1.4	285.5	
				16		1.8	286.2	
		Enabled	10	4		8.3	293.9	
				16		14.2	304.9	
			50	4		3.0	289.2	
				16		4.8	290.5	
			100	4		2.3	289.2	
				16		2.8	289.5	
			200	4		1.9	287.9	
				16		2.4	289.0	

Note:

1. These are based on characterization.

50.5 Unused or Unconnected Pins

For unused pins the default state of the pins will give the lowest current leakage. Thus there is no need to do any configuration of the unused pins in order to lower the power consumption.

50.6 Clocks and Crystal Oscillators

The SAM L10/L11 can be run from internal or external clock sources, or a mix of internal and external sources. An example of usage can be to use the internal 16MHz oscillator as source for the system clock and an external 32.768kHz watch crystal as clock source for the Real-Time counter (RTC).

50.6.1 External Clock Source

Figure 50-6. External Clock Source Schematic

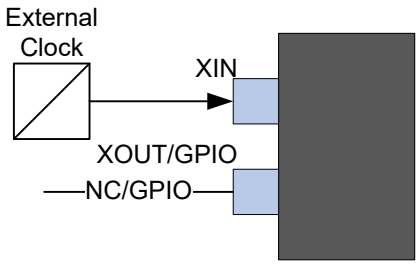
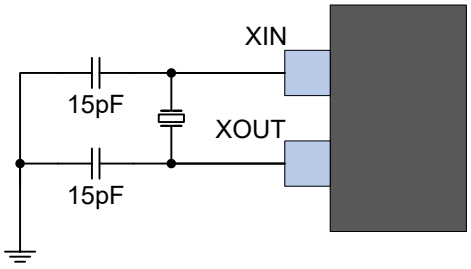


Table 50-4. External Clock Source Connections

Signal Name	Recommended Pin Connection	Description
XIN	XIN is used as input for an external clock signal	Input for inverting oscillator pin
XOUT/GPIO	Can be left unconnected or used as normal GPIO	NC/GPIO

50.6.2 Crystal Oscillator

Figure 50-7. Crystal Oscillator Schematic



The crystal should be located as close to the device as possible. Long signal lines may cause too high load to operate the crystal, and cause crosstalk to other parts of the system.

Table 50-5. Crystal Oscillator Checklist

Signal Name	Recommended Pin Connection	Description
XIN	Load capacitor 15pF(1)(2)	External crystal between 0.4 to 32MHz
XOUT	Load capacitor 15pF(1)(2)	