



Welcome to [E-XFL.COM](https://www.e-xfl.com)

What is "[Embedded - Microcontrollers](#)"?

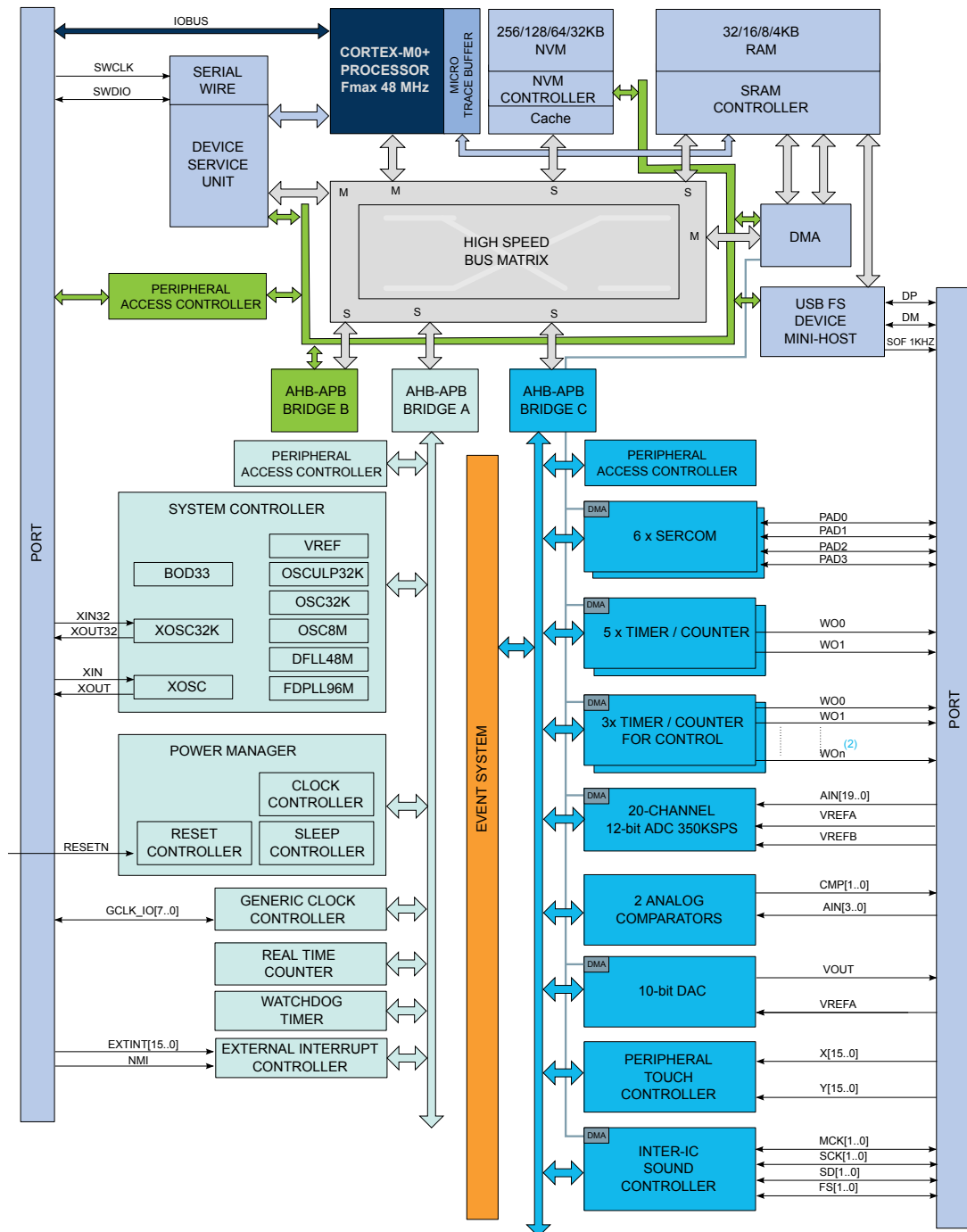
"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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®-M0+
Core Size	32-Bit Single-Core
Speed	48MHz
Connectivity	I ² C, LINbus, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, I ² S, POR, PWM, WDT
Number of I/O	26
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	1.62V ~ 3.6V
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-VFQFN Exposed Pad
Supplier Device Package	32-VQFN (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsamd21e16b-mf

4. Block Diagram



1. Some products have different number of SERCOM instances, Timer/Counter instances, PTC signals and ADC signals. Refer to the Configuration Summary for details.
2. The three TCC instances have different configurations, including the number of Waveform Output (WO) lines. Refer to the TCC Configuration for details.

Related Links

Configuration Summary

TCC Configurations

Bit 5 – USB

Writing a zero to these bits has no effect.

Writing a one to these bits will clear the Write Protect bit for the corresponding peripherals.

Value	Description
0	Write-protection is disabled.
1	Write-protection is enabled.

Bit 4 – DMAC:

Writing a zero to these bits has no effect.

Writing a one to these bits will clear the Write Protect bit for the corresponding peripherals.

Value	Description
0	Write-protection is disabled.
1	Write-protection is enabled.

Bit 3 – PORT

Writing a zero to these bits has no effect.

Writing a one to these bits will clear the Write Protect bit for the corresponding peripherals.

Value	Description
0	Write-protection is disabled.
1	Write-protection is enabled.

Bit 2 – NVMCTRL

Writing a zero to these bits has no effect.

Writing a one to these bits will clear the Write Protect bit for the corresponding peripherals.

Value	Description
0	Write-protection is disabled.
1	Write-protection is enabled.

Bit 1 – DSU

Writing a zero to these bits has no effect.

Writing a one to these bits will clear the Write Protect bit for the corresponding peripherals.

Value	Description
0	Write-protection is disabled.
1	Write-protection is enabled.

Write Protect Set

Name: WPSET

Offset: 0x04

Reset: 0x000002

Property: –

14. Clock System

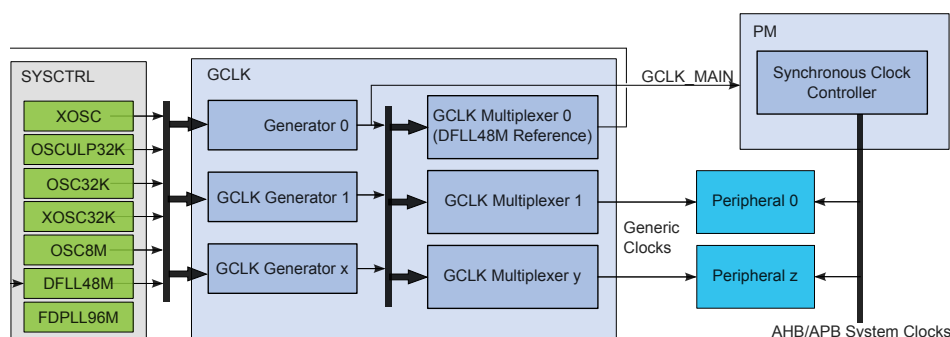
This chapter summarizes the clock distribution and terminology in the SAM D21 device. It will not explain every detail of its configuration. For in-depth documentation, see the respective peripherals descriptions and the *Generic Clock* documentation.

Related Links

[GCLK - Generic Clock Controller](#)

14.1 Clock Distribution

Figure 14-1. Clock distribution



The clock system on the SAM D21 consists of:

- **Clock sources**, controlled by SYSCTRL
 - A clock source provides a time base that is used by other components, such as Generic Clock Generators. Example clock sources are the internal 8MHz oscillator (OSC8M), External crystal oscillator (XOSC) and the Digital frequency locked loop (DFLL48M).
- **Generic Clock Controller (GCLK)** which controls the clock distribution system, made up 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, which in turn generates synchronous clocks.
 - **Generic Clocks:** These are clock signals generated by Generic Clock Generators and output by the Generic Clock Multiplexer, 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. Generic Clock 0 serves as the clock source for the DFLL48M clock input (when multiplying another clock source).
- **Power Manager (PM)**
 - The PM generates and controls the synchronous clocks on the system. This includes the CPU, bus clocks (APB, AHB) as well as the synchronous (to the CPU) user interfaces of the peripherals. It contains clock masks that can turn on/off the user interface of a peripheral as well as prescalers for the CPU and bus clocks.

The next figure shows an example where SERCOM0 is clocked by the DFLL48M in open loop mode. The DFLL48M is enabled, the Generic Clock Generator 1 uses the DFLL48M as its clock source and feeds into Peripheral Channel 20. The Generic Clock 20, also called GCLK_SERCOM0_CORE, is connected to SERCOM0. The SERCOM0 interface, clocked by CLK_SERCOM0_APB, has been unmasked in the APBC Mask register in the PM.

stall. APB registers can also be read while the synchronization is ongoing without causing the peripheral bus to stall.

14.3.1.3 Read-Synchronization

Reading a read-synchronized peripheral core register will cause the peripheral bus to stall immediately until the read-synchronization is complete. STATUS.SYNCBUSY will not be set. Refer to [Synchronization Delay](#) for details on the synchronization delay. Note that reading a read-synchronized peripheral core register while STATUS.SYNCBUSY is one will cause the peripheral bus to stall twice; first because of the ongoing synchronization, and then again because reading a read-synchronized core register will cause the peripheral bus to stall immediately.

14.3.1.4 Completion of synchronization

The user can either poll STATUS.SYNCBUSY or use the Synchronisation Ready interrupt (if available) to check when the synchronization is complete. It is also possible to perform the next read/write operation and wait, as this next operation will be started once the previous write/read operation is synchronized and/or complete.

14.3.1.5 Read Request

The read request functionality is only available to peripherals that have the Read Request register (READREQ) implemented. Refer to the register description of individual peripheral chapters for details.

To avoid forcing the peripheral bus to stall when reading read-synchronized peripheral core registers, the read request mechanism can be used.

Basic Read Request

Writing a '1' to the Read Request bit in the Read Request register (READREQ.RREQ) will request read-synchronization of the register specified in the Address bits in READREQ (READREQ.ADDR) and set STATUS.SYNCBUSY. When read-synchronization is complete, STATUS.SYNCBUSY is cleared. The read-synchronized value is then available for reading without delay until READREQ.RREQ is written to '1' again.

The address to use is the offset to the peripheral's base address of the register that should be synchronized.

Continuous Read Request

Writing a '1' to the Read Continuously bit in READREQ (READREQ.RCONT) will force continuous read-synchronization of the register specified in READREQ.ADDR. The latest value is always available for reading without stalling the bus, as the synchronization mechanism is continuously synchronizing the given value.

SYNCBUSY is set for the first synchronization, but not for the subsequent synchronizations. If another synchronization is attempted, i.e. by executing a write-operation of a write-synchronized register, the read request will be stopped, and will have to be manually restarted.

Note:

The continuous read-synchronization is paused in sleep modes where the generic clock is not running. This means that a new read request is required if the value is needed immediately after exiting sleep.

14.3.1.6 Enable Write-Synchronization

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

When the enable write-synchronization is ongoing (STATUS.SYNCBUSY is one), attempt to do any of the following will cause the peripheral bus to stall until the enable synchronization is complete:

15.6.5.2 Run in Standby Mode

In standby mode, the GCLK can continuously output the generator output to GCLK_IO.

When set, the GCLK can continuously output the generator output to GCLK_IO.

Refer to [Generic Clock Output on I/O Pins](#) for details.

15.6.6 Synchronization

Due to asynchronicity between the main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read.

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

If an operation that requires synchronization is executed while STATUS.SYNCBUSY=1, the bus will be stalled. All operations will complete successfully, but the CPU will be stalled and interrupts will be pending as long as the bus is stalled.

The following registers are synchronized when written:

- Generic Clock Generator Control register (GENCTRL)
- Generic Clock Generator Division register (GENDIV)
- Control register (CTRL)

Required write-synchronization is denoted by the "Write-Synchronized" property in the register description.

Related Links

[Register Synchronization](#)

15.7 Register Summary

Table 15-2. Register Summary

Offset	Name	Bit Pos.								
0x0	CTRL	7:0								SWRST
0x1	STATUS	7:0	SYNCBUSY							
0x2	CLKCTRL	7:0					ID[5:0]			
0x3		15:8	WRTLOCK	CLKEN			GEN[3:0]			
0x4	GENCTRL	7:0					ID[3:0]			
0x5		15:8					SRC[4:0]			
0x6		23:16			RUNSTDBY	DIVSEL	OE	OOV	IDC	GENEN
0x7		31:24								
0x8	GENDIV	7:0					ID[3:0]			
0x9		15:8					DIV[7:0]			
0xA		23:16					DIV[15:8]			
0xB		31:24								

32-bit ARM-Based Microcontrollers

Reset: 0x00

Property: Write-Protected

Bit	7	6	5	4	3	2	1	0
	OVF	SYNCRDY						ALARM0
Access	R/W	R/W						R/W
Reset	0	0						0

Bit 7 – OVF: Overflow Interrupt Enable

Writing a zero to this bit has no effect.

Writing a one to this bit will set the Overflow Interrupt Enable bit and enable the Overflow interrupt.

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

Bit 6 – SYNCRDY: Synchronization Ready Interrupt Enable

Writing a zero to this bit has no effect.

Writing a one to this bit will set the Synchronization Ready Interrupt bit and enable the Synchronization Ready interrupt.

Value	Description
0	The synchronization ready interrupt is disabled.
1	The synchronization ready interrupt is enabled.

Bit 0 – ALARM0: Alarm 0 Interrupt Enable

Writing a zero to this bit has no effect.

Writing a one to this bit will set the Alarm 0 Interrupt Enable bit and enable the Alarm 0 interrupt.

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

19.8.14 Interrupt Flag Status and Clear - MODE0

Name: INTFLAG

Offset: 0x08

Reset: 0x00

Property: -

Bit	7	6	5	4	3	2	1	0
	OVF	SYNCRDY						CMP0
Access	R/W	R/W						R/W
Reset	0	0						0

Bit 7 – OVF: Overflow

This flag is cleared by writing a one to the flag.

Bit 1 – ENABLE: Enable

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

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

Bit 0 – SWRST: Software Reset

Writing a zero to this bit has no effect.

Writing a one to this bit resets all registers in the EIC to their initial state, and the EIC will be disabled.

Writing a one to CTRL.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 CTRL.SWRST until the reset is complete. CTRL.SWRST and STATUS.SYNCBUSY will both be cleared when the reset is complete.

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

21.8.2 Status

Name: STATUS

Offset: 0x01

Reset: 0x00

Property: -

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

Bit 7 – SYNCBUSY: Synchronization Busy

This bit is cleared when the synchronization of registers between the clock domains is complete.

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

21.8.3 Non-Maskable Interrupt Control

Name: NMICTRL

Offset: 0x02

Reset: 0x00

Property: Write-Protected

26.6.4 DMA, Interrupts and Events

Table 26-4. Module Request for SERCOM USART

Condition	Request		
	DMA	Interrupt	Event
Data Register Empty (DRE)	Yes (request cleared when data is written)	Yes	NA
Receive Complete (RXC)	Yes (request cleared when data is read)	Yes	
Transmit Complete (TXC)	NA	Yes	
Receive Start (RXS)	NA	Yes	
Clear to Send Input Change (CTSIC)	NA	Yes	
Receive Break (RXBRK)	NA	Yes	
Error (ERROR)	NA	Yes	

26.6.4.1 DMA Operation

The USART generates the following DMA requests:

- Data received (RX): The request is set when data is available in the receive FIFO. The request is cleared when DATA is read.
- Data transmit (TX): The request is set when the transmit buffer (TX DATA) is empty. The request is cleared when DATA is written.

26.6.4.2 Interrupts

The USART has the following interrupt sources. These are asynchronous interrupts, and can wake up the device from any sleep mode:

- Data Register Empty (DRE)
- Receive Complete (RXC)
- Transmit Complete (TXC)
- Receive Start (RXS)
- Clear to Send Input Change (CTSIC)
- Received Break (RXBRK)
- Error (ERROR)

Each interrupt source has its own interrupt flag. The interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG) will be set when the interrupt condition is met. Each interrupt can be individually enabled by writing '1' to the corresponding bit in the Interrupt Enable Set register (INTENSET), and disabled by writing '1' to the corresponding bit in the Interrupt Enable Clear register (INTENCLR).

An interrupt request is generated when the interrupt flag is set and if the corresponding interrupt is enabled. The interrupt request remains active until either the interrupt flag is cleared, the interrupt is disabled, or the USART is reset. For details on clearing interrupt flags, refer to the INTFLAG register description.

The USART has one common interrupt request line for all the interrupt sources. The value of INTFLAG indicates which interrupt is executed. Note that interrupts must be globally enabled for interrupt requests. Refer to *Nested Vector Interrupt Controller* for details.

32-bit ARM-Based Microcontrollers

Bit	23	22	21	20	19	18	17	16
	ADDRMASK[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
Access								
Reset								
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 – ADDRMASK[7:0]: Address Mask

These bits hold the address mask when the transaction format with address is used (CTRLA.FORM, CTRLB.AMODE).

Bits 7:0 – ADDR[7:0]: Address

These bits hold the address when the transaction format with address is used (CTRLA.FORM, CTRLB.AMODE).

27.8.10 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 buffer. 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.

Writing these bits will write the transmit data buffer. 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.

27.8.11 Debug Control

28.8.8 Address

Name: ADDR

Offset: 0x24

Reset: 0x00000000

Property: PAC Write-Protection, Enable-Protected

Bit	31	30	29	28	27	26	25	24
						ADDRMASK[9:7]		
Access						R/W	R/W	R/W
Reset						0	0	0
Bit	23	22	21	20	19	18	17	16
	ADDRMASK[6:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reset	0	0	0	0	0	0	0	
Bit	15	14	13	12	11	10	9	8
	TENBITEN					ADDR[9:7]		
Access	R/W					R/W	R/W	R/W
Reset	0					0	0	0
Bit	7	6	5	4	3	2	1	0
	ADDR[6:0]							GENCEN
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 26:17 – ADDRMASK[9:0]: Address Mask

These bits act as a second address match register, an address mask register or the lower limit of an address range, depending on the CTRLB.AMODE setting.

Bit 15 – TENBITEN: Ten Bit Addressing Enable

Value	Description
0	10-bit address recognition disabled.
1	10-bit address recognition enabled.

Bits 10:1 – ADDR[9:0]: Address

These bits contain the I²C slave address used by the slave address match logic to determine if a master has addressed the slave.

When using 7-bit addressing, the slave address is represented by ADDR[6:0].

When using 10-bit addressing (ADDR.TENBITEN=1), the slave address is represented by ADDR[9:0].

When the address match logic detects a match, INTFLAG.AMATCH is set and STATUS.DIR is updated to indicate whether it is a read or a write transaction.

Bit 0 – GENCEN: General Call Address Enable

A general call address is an address consisting of all-zeroes, including the direction bit (master write).

Writing '0' to this bit has no effect.

Writing '1' to this bit will clear it.

This bit is not write-synchronized.

28.10.8 Synchronization Busy

Name: SYNCBUSY

Offset: 0x1C

Reset: 0x00000000

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
						SYSOP	ENABLE	SWRST
Access						R	R	R
Reset						0	0	0

Bit 2 – SYSOP: System Operation Synchronization Busy

Writing CTRLB.CMD, STATUS.BUSSTATE, ADDR, or DATA when the SERCOM is enabled requires synchronization. When written, the SYNCBUSY.SYSOP bit will be set until synchronization is complete.

Value	Description
0	System operation synchronization is not busy.
1	System operation synchronization is busy.

Bit 1 – ENABLE: SERCOM Enable Synchronization Busy

Enabling and disabling the SERCOM (CTRLA.ENABLE) requires synchronization. When written, the SYNCBUSY.ENABLE bit will be set until synchronization is complete.

Writes to any register (except for CTRLA.SWRST) while enable synchronization is on-going will be discarded and an APB error will be generated.

Value	Description
0	Enable synchronization is not busy.
1	Enable synchronization is busy.

32-bit ARM-Based Microcontrollers

Bit	15	14	13	12	11	10	9	8
	DATA[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
	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 31:0 – DATA[31:0]: Sample Data

This register is used to transfer data from the Rx Serializer.

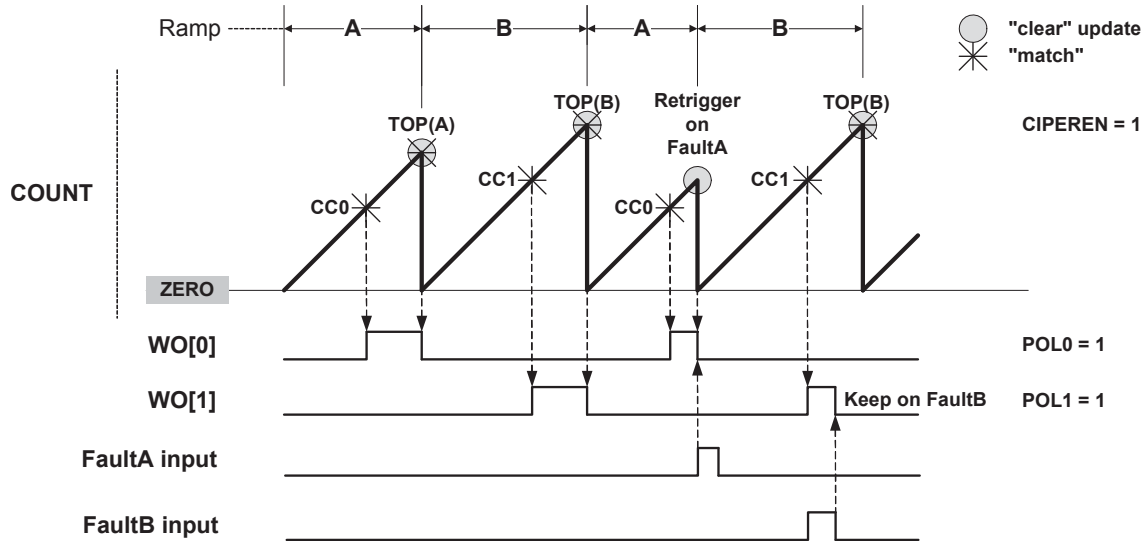
Data samples received by Rx Serializer will be available for reading from RXDATA register, through the Receive Formatting Unit, according to formatting information for Rx Serializer in the RXCTRL register.

A, odd channel output is disabled, and in cycle B, even channel output is disabled. The ramp index changes after each update, but can be software modified using the Ramp index command bits in Control B Set register (CTRLBSET.IDXCMD).

Standard RAMP2 (RAMP2) Operation

Ramp A and B periods are controlled by the PER register value. The PER value can be different on each ramp by the Circular Period buffer option in the Wave register (WAVE.CIPEREN=1). This mode uses a two-channel TCC to generate two output signals, or one output signal with another CC channel enabled in capture mode.

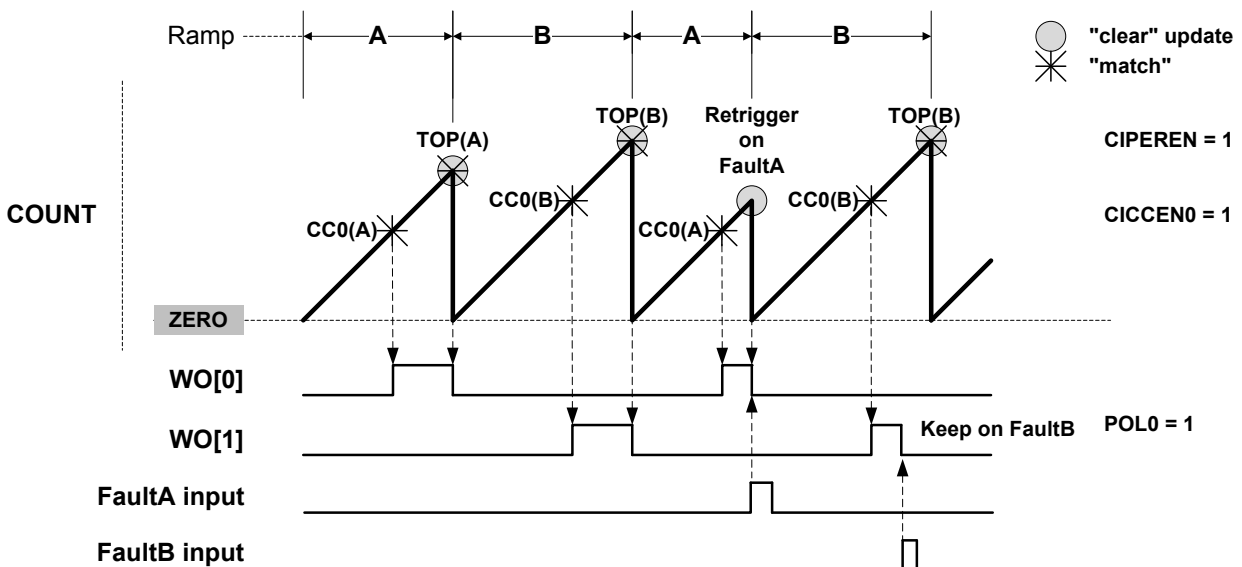
Figure 31-18. RAMP2 Standard Operation



Alternate RAMP2 (RAMP2A) Operation

Alternate RAMP2 operation is similar to RAMP2, but CC0 controls both WO[0] and WO[1] waveforms when the corresponding circular buffer option is enabled (CIPEREN=1). The waveform polarity is the same on both outputs. Channel 1 can be used in capture mode.

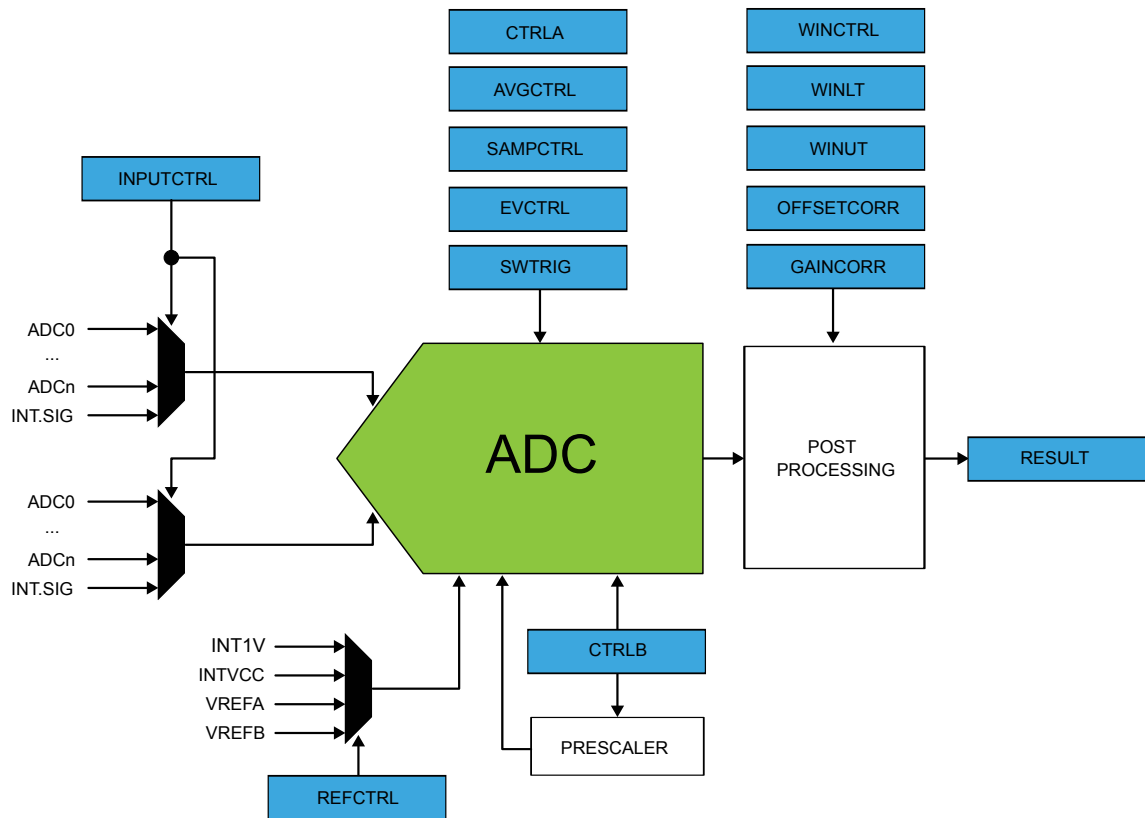
Figure 31-19. RAMP2 Alternate Operation



- Event-triggered conversion for accurate timing (one event input)
- Optional DMA transfer of conversion result
- Hardware gain and offset compensation
- Averaging and oversampling with decimation to support, up to 16-bit result
- Selectable sampling time

33.3 Block Diagram

Figure 33-1. ADC Block Diagram



33.4 Signal Description

Signal Name	Type	Description
VREFA	Analog input	External reference voltage A
VREFB	Analog input	External reference voltage B
ADC[19..0] ⁽¹⁾	Analog input	Analog input channels

Note: Refer to *Configuration Summary* for details on exact number of analog input channels.

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

Related Links

- Comparator (COMP0, COMP1): Indicates a change in comparator status.
- Window (WIN0): Indicates a change in the window status.

Comparator interrupts are generated based on the conditions selected by the Interrupt Selection bit group in the Comparator Control registers (COMPCTRLx.INTSEL). Window interrupts are generated based on the conditions selected by the Window Interrupt Selection bit group in the Window Control register (WINCTRL.WINTSEL[1:0]).

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear (INTFLAG) register is set when the interrupt condition occurs. Each interrupt can be individually enabled by writing a one to the corresponding bit in the Interrupt Enable Set (INTENSET) register, and disabled by writing a one to the corresponding bit in the Interrupt Enable Clear (INTENCLR) 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 AC is reset. See INFLAG register for details on how to clear interrupt flags. All interrupt requests from the peripheral are ORed together on system level to generate one combined interrupt request to the NVIC. The user must read the INTFLAG register to determine which interrupt condition is present.

Note that interrupts must be globally enabled for interrupt requests to be generated.

Related Links

[Nested Vector Interrupt Controller](#)

34.6.13 Events

The AC can generate the following output events:

- Comparator (COMP0, COMP1): Generated as a copy of the comparator status
- Window (WIN0): Generated as a copy of the window inside/outside status

Output events must be enabled to be generated. Writing a one to an Event Output bit in the Event Control register (EVCTRL.COMPEOx) enables the corresponding output event. Writing a zero to this bit disables the corresponding output event. The events must be correctly routed in the Event System.

The AC can take the following action on an input event:

- Single-shot measurement
- Single-shot measurement in window mode

Writing a one to an Event Input bit into the Event Control register (EVCTRL.COMPEIx) enables the corresponding action on input event. Writing a zero to this bit disables the corresponding action on input event. Note that if several events are connected to the AC, the enabled action will be taken on any of the incoming events. Refer to the Event System chapter for details on configuring the event system.

When EVCTRL.COMPEIx is one, the event will start a comparison on COMPx after the start-up time delay. In normal mode, each comparator responds to its corresponding input event independently. For a pair of comparators in window mode, either comparator event will trigger a comparison on both comparators simultaneously.

34.6.14 Sleep Mode Operation

The Run in Standby bits in the Comparator x Control registers (COMPCTRLx.RUNSTDBY) control the behavior of the AC during standby sleep mode. Each RUNSTDBY bit controls one comparator. When the bit is zero, the comparator is disabled during sleep, but maintains its current configuration. When the bit is one, the comparator continues to operate during sleep. Note that when RUNSTDBY is zero, the analog blocks are powered off for the lowest power consumption. This necessitates a start-up time delay when the system returns from sleep.

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

If an operation that requires synchronization is executed while its busy bit is one, the operation is discarded and an error is generated.

The following bits need synchronization when written:

- Software Reset bit in the Control A register (CTRLA.SWRST)
- Enable bit in the Control A register (CTRLA.ENABLE)
- All bits in the Data register (DATA)
- All bits in the Data Buffer register (DATABUF)

Write-synchronization is denoted by the Write-Synchronized property in the register description.

The following bits need synchronization when read:

- All bits in the Data register (DATA)

35.6.8 Additional Features

35.6.8.1 DAC as an Internal Reference

The DAC output can be internally enabled as input to the analog comparator. This is enabled by writing a one to the Internal Output Enable bit in the Control B register (CTRLB.IOEN). It is possible to have the internal and external output enabled simultaneously.

The DAC output can also be enabled as input to the Analog-to-Digital Converter. In this case, the output buffer must be enabled.

35.6.8.2 Data Buffer

The Data Buffer register (DATABUF) and the Data register (DATA) are linked together to form a two-stage FIFO. The DAC uses the Start Conversion event to load data from DATABUF into DATA and start a new conversion. The Start Conversion event is enabled by writing a one to the Start Event Input bit in the Event Control register (EVCTRL.STARTEI). If a Start Conversion event occurs when DATABUF is empty, an Underrun interrupt request is generated if the Underrun interrupt is enabled.

The DAC can generate a Data Buffer Empty event when DATABUF becomes empty and new data can be loaded to the buffer. The Data Buffer Empty event is enabled by writing a one to the Empty Event Output bit in the Event Control register (EVCTRL.EMPTYEO). A Data Buffer Empty interrupt request is generated if the Data Buffer Empty interrupt is enabled.

35.6.8.3 Voltage Pump

When the DAC is used at operating voltages lower than 2.5V, the voltage pump must be enabled. This enabling is done automatically, depending on operating voltage.

The voltage pump can be disabled by writing a one to the Voltage Pump Disable bit in the Control B register (CTRLB.VPD). This can be used to reduce power consumption when the operating voltage is above 2.5V.

The voltage pump uses the asynchronous GCLK_DAC clock, and requires that the clock frequency be at least four times higher than the sampling period.

Notes: 1. When using DFLL48M in USB recovery mode, the Fine Step value must be Ah to guarantee a USB clock at +/-0.25% before 11ms after a resume.

2. Very high signal quality and crystal less. It is the best setup for USB Device mode.

3. FDPLL lock time is short when the clock frequency source is high (> 1MHz). Thus, FDPLL and external OSC can be stopped during USB suspend mode to reduce consumption and guarantee a USB wake-up time (See TDRSMDN in USB specification).

37.15 Timing Characteristics

37.15.1 External Reset

Table 37-60. External Reset Characteristics

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
t_{EXT}	Minimum reset pulse width		10	-	-	ns

37.15.2 SERCOM in SPI Mode Timing

Figure 37-20. SPI Timing Requirements in Master Mode

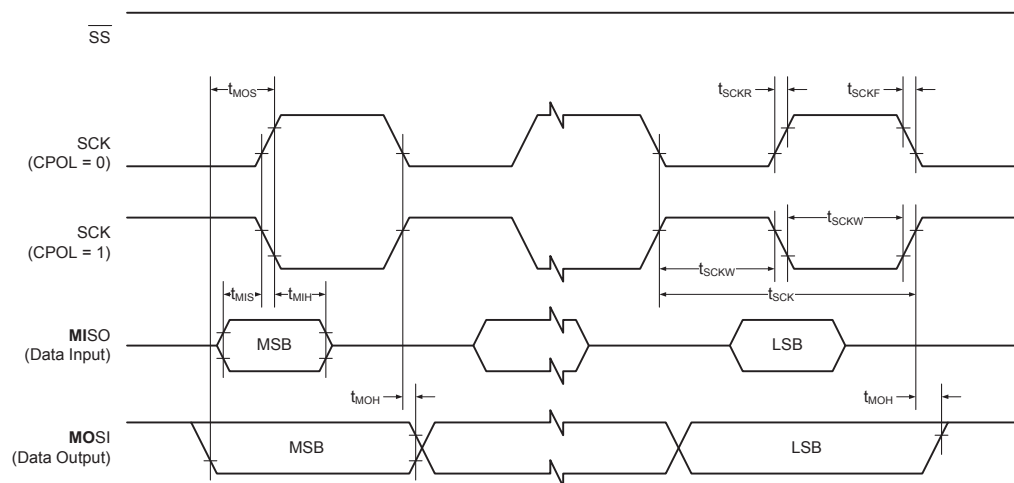


Figure 37-21. SPI Timing Requirements in Slave Mode

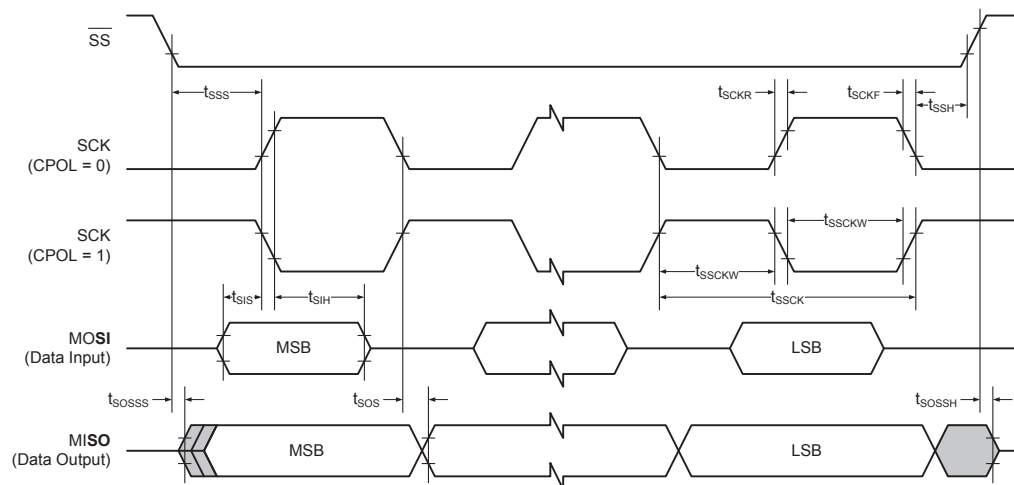


Figure 39-13. 20-pin IDC JTAG Connector

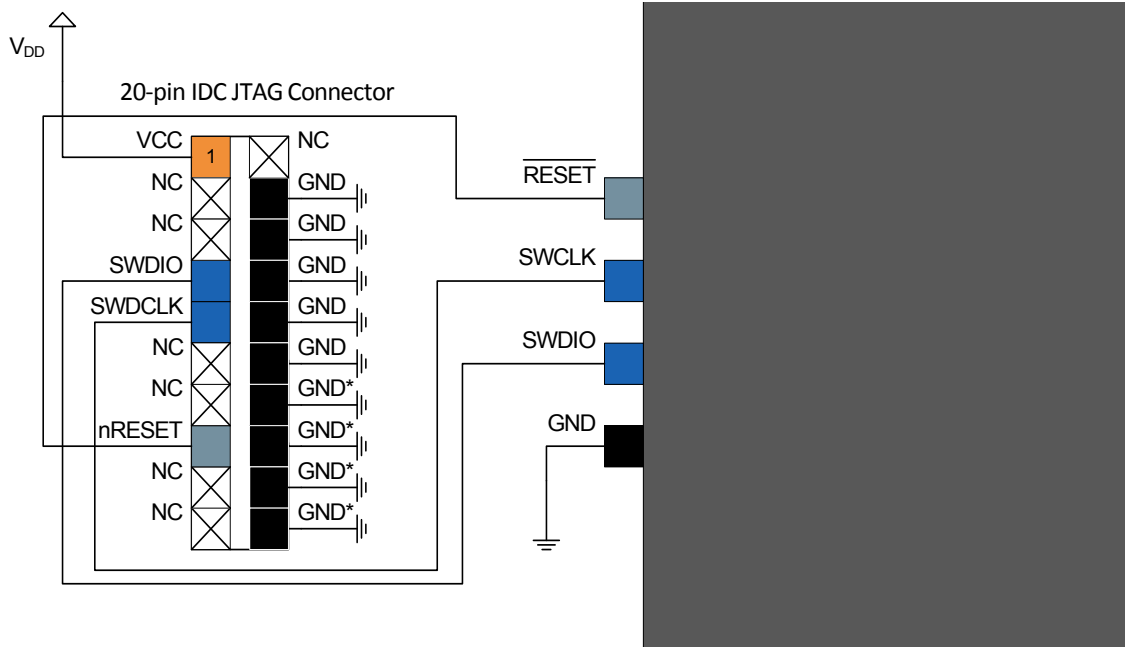


Table 39-12. 20-pin IDC JTAG Connector

Header Signal Name	Description
SWDCLK	Serial wire clock pin
SWDIO	Serial wire bidirectional data pin
RESET	Target device reset pin, active low
VCC	Target voltage sense, should be connected to the device V_{DD}
GND	Ground
GND*	These pins are reserved for firmware extension purposes. They can be left open or connected to GND in normal debug environment. They are not essential for SWD in general.

39.8 USB Interface

The USB interface consists of a differential data pair (D+/D-) and a power supply (VBUS, GND). Refer to the Electrical Characteristics for operating voltages which will allow USB operation.

Table 39-13. USB Interface Checklist

Signal Name	Recommended Pin Connection	Description
D+	<ul style="list-style-type: none"> The impedance of the pair should be matched on the PCB to minimize reflections. USB differential tracks should be routed with the same characteristics (length, width, number of vias, etc.) Signals should be routed as parallel as possible, with a minimum number of angles and vias 	USB full speed / low speed positive data upstream pin
D-		USB full speed / low speed negative data upstream pin

40.1.2 Die Revision B

40.1.2.1 Device

1 – When VDDIN is lower than the POR threshold during power rise or fall, an internal pull-up resistor is enabled on pins with PTC functionality (see PORT Function Multiplexing). Note that this behavior will be present even if PTC functionality is not enabled on the pin. The POR level is defined in the “Power-On Reset (POR) Characteristics” chapter.

Errata reference: 12117

Fix/Workaround:

Use a pin without PTC functionality if the pull-up could damage your application during power up.

2 – The I2S is non-functional in the slave mode (i.e. when (FSSEL=1, SCKSEL=1).

Errata reference: 13407

Fix/Workaround:

None. FSSEL and SCKSEL must be 0.

3 – If APB clock is stopped and GCLK clock is running, APB read access to read-synchronized registers will freeze the system. The CPU and the DAP AHB-AP are stalled, as a consequence debug operation is impossible.

Errata reference: 10416

Fix/Workaround:

Do not make read access to read-synchronized registers when APB clock is stopped and GCLK is running. To recover from this situation, power cycle the device or reset the device using the RESETN pin.

4 – The software reset SWRST does not properly propagate inside the I2S module. As a consequence, the slave mode may not be reconfigured correctly and may result in unexpected behavior of the SYNCBUSY register.

Errata reference: 12848

Fix/workaround:

None.

5 – PA24 and PA25 cannot be used as input when configured as GPIO with continuous sampling (cannot be read by PORT).

Errata reference: 12005

Fix/Workaround:

- Use PA24 and PA25 for peripherals or only as output pins.
- Or configure PA31 to PA24 for on-demand sampling (CTRL[31:24] all zeroes) and access the IN register through the APB (not the IOBUS), to allow waiting for on-demand sampling.

6 – Rx serializer in the RIGHT Data Slot Formatting Adjust mode (SERCTRL.SLOTADJ clear) does not work when the slot size is not 32 bits.

Errata reference: 13411

Fix/Workaround:

In SERCTRL.SERMODE RX, SERCTRL.SLOTADJ RIGHT must be used with CLKCTRL.SLOTSIZE 32.

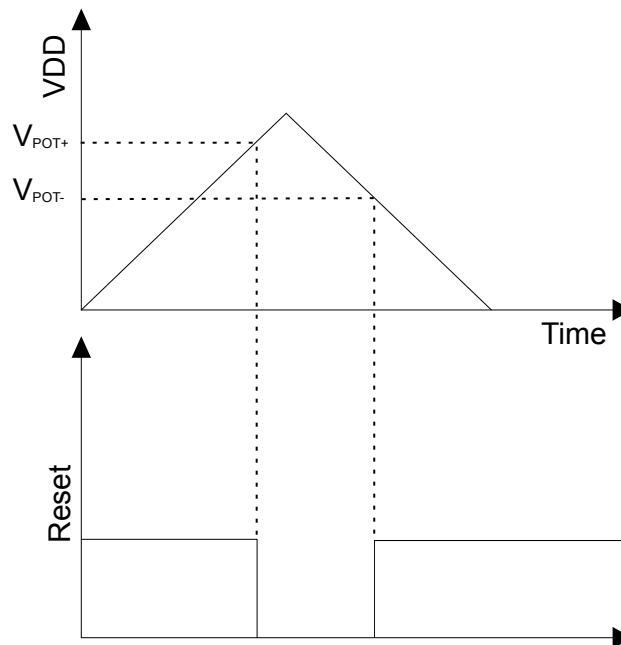
44.6 Analog Characteristics

44.6.1 Power-On Reset (POR) Characteristics

Table 44-10. POR Characteristics

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units
V_{POT+}	Voltage threshold on V_{DD} rising	V_{DD} falls at 1V/ms or slower	1.27	1.45	1.58	V
V_{POT-}	Voltage threshold on V_{DD} falling		0.72	0.99	1.32	V

Figure 44-2. POR Operating Principle



44.6.2 Brown-Out Detectors Characteristics

44.6.2.1 BOD33

Table 44-11. BOD33 Characteristics (Device Variant A)

Symbol	Parameter	Conditions	Temp.	Min.	Typ.	Max.	Units
	Step size, between adjacent values in BOD33.LEVEL			-	34	-	mV
V_{HYST}	$V_{BOD+} - V_{BOD-}$	Hysteresis ON		35	-	170	mV
t_{DET}	Detection time	Time with $V_{DDANA} < V_{TH}$ necessary to generate a reset signal		-	0.9 ⁽¹⁾	-	μ s