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®-M0+
Core Size	32-Bit Single-Core
Speed	48MHz
Connectivity	I²C, LINbus, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, I²S, POR, PWM, WDT
Number of I/O	38
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 14x12b; D/A 1x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-TQFP
Supplier Device Package	48-TQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsamd21g16b-aut

Bit 20 – I2S
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 19 – PTC
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 18 – DAC:
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 17 – AC
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 16 – ADC
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.

Bits 11, 12, 13, 14, 15 – TC3, TC4, TC5, TC4, TC7
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.

Bits 8, 9, 10 – TCCn
Writing a zero to these bits has no effect.

The OSCULP32K is enabled by default after a power-on reset (POR) and will always run except during POR. The OSCULP32K has a 32.768kHz output and a 1.024kHz output that are always running.

The frequency of the OSCULP32K oscillator is controlled by the value in the 32kHz Ultra Low Power Internal Oscillator Calibration bits (OSCULP32K.CALIB) in the 32kHz Ultra Low Power Internal Oscillator Control register. OSCULP32K.CALIB is automatically loaded from Flash Factory Calibration during startup, and is used to compensate for process variation, as described in the *Electrical Characteristics*. The calibration value can be overridden by the user by writing to OSCULP32K.CALIB.

Related Links

Electrical Characteristics

GCLK - Generic Clock Controller

17.6.6 8MHz Internal Oscillator (OSC8M) Operation

OSC8M is an internal oscillator operating in open-loop mode and generating an 8MHz frequency. The OSC8M is factory-calibrated under typical voltage and temperature conditions.

OSC8M is the default clock source that is used after a power-on reset (POR). The OSC8M can be used as a source for the generic clock generators, as described in the *GCLK – Generic Clock Controller*.

In order to enable OSC8M, the Oscillator Enable bit in the OSC8M Control register (OSC8M.ENABLE) must be written to one. OSC8M will not be enabled until OSC8M.ENABLE is set. In order to disable OSC8M, OSC8M.ENABLE must be written to zero. OSC8M will not be disabled until OSC8M is cleared.

The frequency of the OSC8M oscillator is controlled by the value in the calibration bits (OSC8M.CALIB) in the OSC8M Control register. CALIB is automatically loaded from Flash Factory Calibration during start-up, and is used to compensate for process variation, as described in the *Electrical Characteristics*.

The user can control the oscillation frequency by writing to the Frequency Range (FRANGE) and Calibration (CALIB) bit groups in the 8MHz RC Oscillator Control register (OSC8M). It is not recommended to update the FRANGE and CALIB bits when the OSC8M is enabled. As this is in open-loop mode, the frequency will be voltage, temperature and process dependent. Refer to the *Electrical Characteristics* for details.

OSC8M is automatically switched off in certain sleep modes to reduce power consumption, as described in the *PM – Power Manager*.

Related Links

PM – Power Manager

Electrical Characteristics

GCLK - Generic Clock Controller

17.6.7 Digital Frequency Locked Loop (DFLL48M) Operation

The DFLL48M can operate in both open-loop mode and closed-loop mode. In closed-loop mode, a low-frequency clock with high accuracy can be used as the reference clock to get high accuracy on the output clock (CLK_DFLL48M).

The DFLL48M can be used as a source for the generic clock generators, as described in the *GCLK – Generic Clock Controller*.

Related Links

GCLK - Generic Clock Controller

GAIN[2:0]	Recommended Max Frequency
0x0	2MHz
0x1	4MHz
0x2	8MHz
0x3	16MHz
0x4	30MHz
0x5-0x7	Reserved

Bit 7 – ONDEMAND: On Demand Control
The On Demand operation mode allows an oscillator to be enabled or disabled, depending on peripheral clock requests.

In On Demand operation mode, i.e., if the XOSC.ONDEMAND bit has been previously written to one, the oscillator will be running only when requested by a peripheral. If there is no peripheral requesting the oscillator s clock source, the oscillator will be in a disabled state.

If On Demand is disabled, the oscillator will always be running when enabled.

In standby sleep mode, the On Demand operation is still active if the XOSC.RUNSTDBY bit is one. If XOSC.RUNSTDBY is zero, the oscillator is disabled.

Value	Description
0	The oscillator is always on, if enabled.
1	The oscillator is enabled when a peripheral is requesting the oscillator to be used as a clock source. The oscillator is disabled if no peripheral is requesting the clock source.

Bit 6 – RUNSTDBY: Run in Standby
This bit controls how the XOSC behaves during standby sleep mode:

Value	Description
0	The oscillator is disabled in standby sleep mode.
1	The oscillator is not stopped in standby sleep mode. If XOSC.ONDEMAND is one, the clock source will be running when a peripheral is requesting the clock. If XOSC.ONDEMAND is zero, the clock source will always be running in standby sleep mode.

Bit 2 – XTALEN: Crystal Oscillator Enable
This bit controls the connections between the I/O pads and the external clock or crystal oscillator:

Value	Description
0	External clock connected on XIN. XOUT can be used as general-purpose I/O.
1	Crystal connected to XIN/XOUT.

Bit 1 – ENABLE: Oscillator Enable

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

17.8.6 32kHz External Crystal Oscillator (XOSC32K) Control

Bit 7 – ONDEMAND: On Demand Control
The On Demand operation mode allows an oscillator to be enabled or disabled depending on peripheral clock requests.

In On Demand operation mode, i.e., if the ONDEMAND bit has been previously written to one, the oscillator will only be running when requested by a peripheral. If there is no peripheral requesting the oscillator s clock source, the oscillator will be in a disabled state.

If On Demand is disabled the oscillator will always be running when enabled.

In standby sleep mode, the On Demand operation is still active if the XOSC32K.RUNSTDBY bit is one. If XOSC32K.RUNSTDBY is zero, the oscillator is disabled.

Value	Description
0	The oscillator is always on, if enabled.
1	The oscillator is enabled when a peripheral is requesting the oscillator to be used as a clock source. The oscillator is disabled if no peripheral is requesting the clock source.

Bit 6 – RUNSTDBY: Run in Standby
This bit controls how the XOSC32K behaves during standby sleep mode:

Value	Description
0	The oscillator is disabled in standby sleep mode.
1	The oscillator is not stopped in standby sleep mode. If XOSC32K.ONDEMAND is one, the clock source will be running when a peripheral is requesting the clock. If XOSC32K.ONDEMAND is zero, the clock source will always be running in standby sleep mode.

Bit 5 – AAMPEN: Automatic Amplitude Control Enable

Value	Description
0	The automatic amplitude control for the crystal oscillator is disabled.
1	The automatic amplitude control for the crystal oscillator is enabled.

Bit 3 – EN32K: 32kHz Output Enable
This bit controls the connections between the I/O pads and the external clock or crystal oscillator:

Value	Description
0	The 32kHz output is disabled.
1	The 32kHz output is enabled.

Bit 2 – XTALEN: Crystal Oscillator Enable
This bit controls the connections between the I/O pads and the external clock or crystal oscillator:

Value	Description
0	External clock connected on XIN32. XOUT32 can be used as general-purpose I/O.
1	Crystal connected to XIN32/XOUT32.

Bit 1 – ENABLE: Oscillator Enable

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

20. DMAC – Direct Memory Access Controller

20.1 Overview

The Direct Memory Access Controller (DMAC) contains both a Direct Memory Access engine and a Cyclic Redundancy Check (CRC) engine. The DMAC can transfer data between memories and peripherals, and thus off-load these tasks from the CPU. It enables high data transfer rates with minimum CPU intervention, and frees up CPU time. With access to all peripherals, the DMAC can handle automatic transfer of data between communication modules.

The DMA part of the DMAC has several DMA channels which all can receive different types of transfer triggers to generate transfer requests from the DMA channels to the arbiter, see also the Block Diagram. The arbiter will grant one DMA channel at a time to act as the active channel. When an active channel has been granted, the fetch engine of the DMAC will fetch a transfer descriptor from the SRAM and store it in the internal memory of the active channel, which will execute the data transmission.

An ongoing data transfer of an active channel can be interrupted by a higher prioritized DMA channel. The DMAC will write back the updated transfer descriptor from the internal memory of the active channel to SRAM, and grant the higher prioritized channel to start transfer as the new active channel. Once a DMA channel is done with its transfer, interrupts and events can be generated optionally.

The DMAC has four bus interfaces:

- The *data transfer bus* is used for performing the actual DMA transfer.
- The *AHB/APB Bridge bus* is used when writing and reading the I/O registers of the DMAC.
- The *descriptor fetch bus* is used by the fetch engine to fetch transfer descriptors before data transfer can be started or continued.
- The *write-back bus* is used to write the transfer descriptor back to SRAM.

All buses are AHB master interfaces but the AHB/APB Bridge bus, which is an APB slave interface.

The CRC engine can be used by software to detect an accidental error in the transferred data and to take corrective action, such as requesting the data to be sent again or simply not using the incorrect data.

20.2 Features

- Data transfer from:
  - Peripheral to peripheral
  - Peripheral to memory
  - Memory to peripheral
  - Memory to memory
- Transfer trigger sources
  - Software
  - Events from Event System
  - Dedicated requests from peripherals
- SRAM based transfer descriptors
  - Single transfer using one descriptor
  - Multi-buffer or circular buffer modes by linking multiple descriptors
- Up to 12 channels

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

Bit 3 – NMIFILTEN: Non-Maskable Interrupt Filter Enable

Value	Description
0	NMI filter is disabled.
1	NMI filter is enabled.

Bits 2:0 – NMISENSE[2:0]: Non-Maskable Interrupt Sense
These bits define on which edge or level the NMI triggers.

NMISENSE[2:0]	Name	Description
0x0	NONE	No detection
0x1	RISE	Rising-edge detection
0x2	FALL	Falling-edge detection
0x3	BOTH	Both-edges detection
0x4	HIGH	High-level detection
0x5	LOW	Low-level detection
0x6-0x7		Reserved

21.8.4 Non-Maskable Interrupt Flag Status and Clear

Name: NMIFLAG
Offset: 0x03
Reset: 0x00
Property: -

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

Bit 0 – NMI: Non-Maskable Interrupt
This flag is cleared by writing a one to it.

This flag is set when the NMI pin matches the NMI sense configuration, and will generate an interrupt request.

Writing a zero to this bit has no effect.

Writing a one to this bit clears the non-maskable interrupt flag.

21.8.5 Event Control

Name: EVCTRL

Some event generators can generate an event when the system clock is stopped. The generic clock (GCLK_EVSYS_CHANNELx) for this channel will be restarted if the channel uses a synchronized path or a resynchronized path, without waking the system from sleep. The clock remains active only as long as necessary to handle the event. After the event has been handled, the clock will be turned off and the system will remain in the original sleep mode. This is known as SleepWalking. When an asynchronous path is used, there is no need for the clock to be activated for the event to be propagated to the user.

On a software reset, all registers are set to their reset values and any ongoing events are canceled.

24.7 Register Summary

Table 24-1. Event System Register Summary

Offset	Name	Bit Pos.								
0x00	CTRL	7:0				GCLKREQ				SWRST
0x01 ... 0x03	Reserved									
0x04	CHANNEL	7:0					CHANNEL[3:0]			
0x05		15:8								SWEVT
0x06		23:16		EVGEN[6:0]						
0x07		31:24					EDGSEL[1:0]		PATH[1:0]	
0x08	USER	7:0				USER[4:0]				
0x09		15:8				CHANNEL[4:0]				
0x0A	Reserved									
0x0B	Reserved									
0x0C	CHSTATUS	7:0	USRRDY7	USRRDY6	USRRDY5	USRRDY4	USRRDY3	USRRDY2	USRRDY1	USRRDY0
0x0D		15:8	CHBUSY7	CHBUSY6	CHBUSY5	CHBUSY4	CHBUSY3	CHBUSY2	CHBUSY1	CHBUSY0
0x0E		23:16					USRRDY11	USRRDY10	USRRDY9	USRRDY8
0x0F		31:24					CHBUSY11	CHBUSY10	CHBUSY9	CHBUSY8
0x10	INTENCLR	7:0	OVR7	OVR6	OVR5	OVR4	OVR3	OVR2	OVR1	OVR0
0x11		15:8	EVD7	EVD6	EVD5	EVD4	EVD3	EVD2	EVD1	EVD0
0x12		23:16					OVR11	OVR10	OVR9	OVR8
0x13		31:24					EVD11	EVD10	EVD9	EVD8
0x14	INTENSET	7:0	OVR7	OVR6	OVR5	OVR4	OVR3	OVR2	OVR1	OVR0
0x15		15:8	EVD7	EVD6	EVD5	EVD4	EVD3	EVD2	EVD1	EVD0
0x16		23:16					OVR11	OVR10	OVR9	OVR8
0x17		31:24					EVD11	EVD10	EVD9	EVD8
0x18	INTFLAG	7:0	OVR7	OVR6	OVR5	OVR4	OVR3	OVR2	OVR1	OVR0
0x19		15:8	EVD7	EVD6	EVD5	EVD4	EVD3	EVD2	EVD1	EVD0
0x1A		23:16					OVR11	OVR10	OVR9	OVR8
0x1B		31:24					EVD11	EVD10	EVD9	EVD8

24.8 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.

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

Bit 0 – SWRST: Software Reset Synchronization Busy
Resetting the SERCOM (CTRLA.SWRST) requires synchronization. When written, the SYNCBUSY.SWRST bit will be set until synchronization is complete.

Writes to any register while synchronization is on-going will be discarded and an APB error will be generated.

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

26.8.10 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 register. The register should be read only when the Receive Complete Interrupt Flag bit in the Interrupt Flag Status and Clear register (INTFLAG.RXC) is set. The status bits in STATUS should be read before reading the DATA value in order to get any corresponding error.

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

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.

28.8.5 Interrupt Flag Status and Clear

Name: INTFLAG
Offset: 0x18
Reset: 0x00
Property: -

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

29.6.8 DMA, Interrupts and Events

Table 29-4. Module Request for I²S

Condition	DMA request	DMA request is cleared	Interrupt request	Event input/ output
Receive Ready	YES	When data is read	YES	
Transmit Ready (Buffer empty)	YES	When data is written	YES	
Receive Overrun			YES	
Transmit Underrun			YES	

29.6.8.1 DMA Operation

Each Serializer can be connected either to one single DMAC channel or to one DMAC channel per data slot in stereo mode. This is selected by writing the SERCTRLm.DMA bit:

Table 29-5. I²C DMA Request Generation

SERCTRLm.DMA	Mode	Slot Parity	DMA Request Trigger
0	Receiver	all	I2S_DMAC_ID_RX_m
	Transmitter	all	I2S_DMAC_ID_TX_m
1	Receiver	even	I2S_DMAC_ID_RX_m
		odd	I2S_DMAC_ID_TX_m
	Transmitter	even	I2S_DMAC_ID_TX_m
		odd	I2S_DMAC_ID_RX_m

The DMAC reads from the DATAm register and writes to the DATAm register for all data slots, successively.

The DMAC transfers may use 32-, 16- or or 8-bit transactions according to the value of the SERCTRLm.DATASIZE field. 8-bit compact stereo uses 16-bit and 16-bit compact stereo uses 32-bit transactions.

29.6.8.2 Interrupts

The I²S has the following interrupt sources:

- Receive Ready (RXRDYm): this is an asynchronous interrupt and can be used to wake-up the device from any sleep mode.
- Receive Overrun (RXORm): this is an asynchronous interrupt and can be used to wake-up the device from any sleep mode.
- Transmit Ready (TXRDYm): this is an asynchronous interrupt and can be used to wake-up the device from any sleep mode.
- Transmit Underrun (TXORm): this is an asynchronous interrupt and can be used to wake-up the device from any sleep mode.

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

Bit 11 – DFS: Non-Recoverable Debug Fault Interrupt Enable
Writing a '0' to this bit has no effect.

Writing a '1' to this bit will clear the Debug Fault State Interrupt Disable/Enable bit, which disables the Debug Fault State interrupt.

Value	Description
0	The Debug Fault State interrupt is disabled.
1	The Debug Fault State interrupt is enabled.

Bit 3 – ERR: Error Interrupt Enable
Writing a '0' to this bit has no effect.

Writing a '1' to this bit will clear the Error Interrupt Disable/Enable bit, which disables the Compare interrupt.

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

Bit 2 – CNT: Counter Interrupt Enable
Writing a '0' to this bit has no effect.

Writing a '1' to this bit will clear the Counter Interrupt Disable/Enable bit, which disables the Counter interrupt.

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

Bit 1 – TRG: Retrigger Interrupt Enable
Writing a '0' to this bit has no effect.

Writing a '1' to this bit will clear the Retrigger Interrupt Disable/Enable bit, which disables the Retrigger interrupt.

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

Bit 0 – OVF: Overflow Interrupt Enable
Writing a '0' to this bit has no effect.

Writing a '1' to this bit will clear the Overflow Interrupt Disable/Enable bit, which disables the Overflow interrupt request.

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

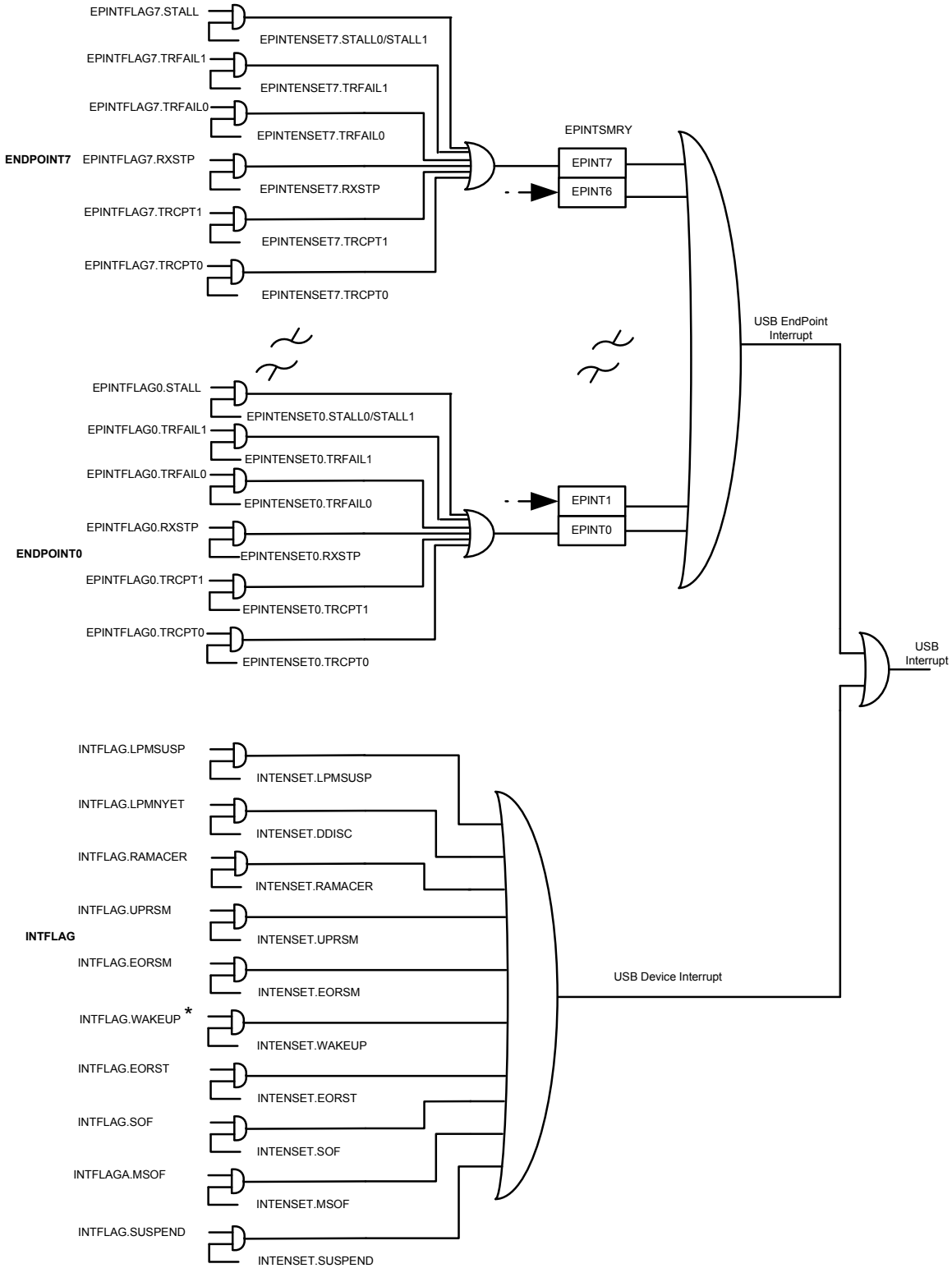
31.8.11 Interrupt Enable Set

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.

Name: INTENSET

32.6.2.16 USB Device Interrupt

Figure 32-9. Device Interrupt



* Asynchronous interrupt

The WAKEUP is an asynchronous interrupt and can be used to wake-up the device from any sleep mode.

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units
I_{DFLL}	Power consumption on V_{DDIN}	f_{REF} = XTAL, 32 .768kHz, 100ppm DFLLMUL = 1464	-	425	482	µA
t_{LOCK}	Lock time	f_{REF} = XTAL, 32 .768kHz, 100ppm DFLLMUL = 1464 DFLLVAL.COARSE = DFLL48M COARSE CAL DFLLVAL.FINE = 512 DFLLCTRL.BPLCKC = 1 DFLLCTRL.QLDIS = 0 DFLLCTRL.CCDIS = 1 DFLLMUL.FSTEP = 10	100	200	500	µs

Table 37-52. DFLL48M Characteristics - Closed Loop Mode(1) (Device Variant B and C)

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units
f_{OUT}	Average Output frequency	f_{REF} = 32 .768kHz	47.963	47.972	47.981	MHz
f_{REF}	Reference frequency		0.732	32.768	33	kHz
Jitter	Cycle to Cycle jitter	f_{REF} = 32 .768kHz	-	-	0.42	ns
I_{DFLL}	Power consumption on V_{DDIN}	f_{REF} =32 .768kHz	-	403	453	µA
t_{LOCK}	Lock time	f_{REF} = 32 .768kHz DFLLVAL.COARSE = DFLL48M COARSE CAL DFLLVAL.FINE = 512 DFLLCTRL.BPLCKC = 1 DFLLCTRL.QLDIS = 0 DFLLCTRL.CCDIS = 1 DFLLMUL.FSTEP = 10	-	200	500	µs

1. To insure that the device stays within the maximum allowed clock frequency, any reference clock for DFLL in close loop must be within a 2% error accuracy.

39.5 Clocks and Crystal Oscillators

The SAM D21 can be run from internal or external clock sources, or a mix of internal and external sources. An example of usage will be to use the internal 8MHz oscillator as source for the system clock, and an external 32.768kHz watch crystal as clock source for the Real-Time counter (RTC).

39.5.1 External Clock Source

Figure 39-5. External Clock Source Example Schematic

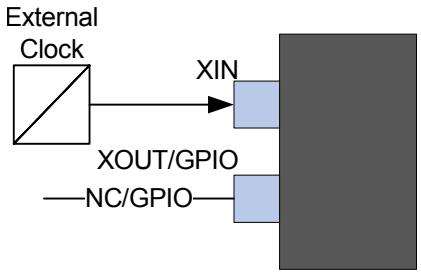
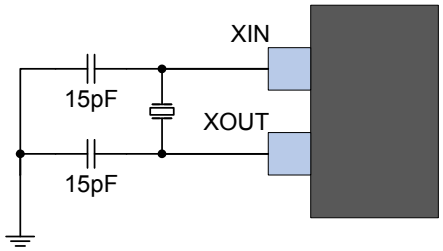


Table 39-4. External Clock Source Connections

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	

39.5.2 Crystal Oscillator

Figure 39-6. Crystal Oscillator Example 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 39-5. Crystal Oscillator Checklist

Signal Name	Recommended Pin Connection	Description
XIN	Load capacitor 15pF(1)(2)	External crystal between 0.4 to 30MHz
XOUT	Load capacitor 15pF(1)(2)	

1. These values are given only as typical example.
2. Decoupling capacitor should be placed close to the device for each supply pin pair in the signal group.

39.5.3 External Real Time Oscillator

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

40.1.1.10 SERCOM

1 – The I2C Slave SCL Low Extend Time-out (CTRLA.SEXTTOEN) and Master SCL Low Extend Time-out (CTRLA.MEXTTOEN) cannot be used if SCL Low Time-out (CTRLA.LOWTOUT) is disabled. When SCTRLA.LOWTOUT=0, the GCLK_SERCOM_SLOW is not requested.
Errata reference: 12003
Fix/Workaround:
To use the Master or Slave SCL low extend time-outs, enable the SCL Low Time-out (CTRLA.LOWTOUT=1).

2 – In USART autobaud mode, missing stop bits are not recognized as inconsistent sync (ISF) or framing (FERR) errors.
Errata reference: 13852
Fix/Workaround:
None

3 – If the SERCOM is enabled in SPI mode with SSL detection enabled (CTRLB.SSDE) and CTRLB.RXEN=1, an erroneous slave select low interrupt (INTFLAG.SSL) can be generated.
Errata reference: 13369
Fix/Workaround:
Enable the SERCOM first with CTRLB.RXEN=0. In a subsequent write, set CTRLB.RXEN=1.

4 – In TWI master mode, an ongoing transaction should be stalled immediately when DBGCTRL.DBGSTOP is set and the CPU enters debug mode. Instead, it is stopped when the current byte transaction is completed and the corresponding interrupt is triggered if enabled.
Errata reference: 12499
Fix/Workaround:
In TWI master mode, keep DBGCTRL.DBGSTOP=0 when in debug mode.

40.1.1.11 TC

1 – Spurious TC overflow and Match/Capture events may occur.
Errata reference: 13268
Fix/Workaround:
Do not use the TC overflow and Match/Capture events. Use the corresponding Interrupts instead.

40.1.1.12 TCC

1 – Using TCC in dithering mode with external retrigger events can lead to unexpected stretch of right aligned pulses, or shrink of left aligned pulses.
Errata reference: 15625
Fix/Workaround:
Do not use retrigger events/actions when TCC is configured in dithering mode.

2 – The TCC interrupts FAULT1, FAULT0, FAULTB, FAULTA, DFS, ERR,and CNT cannot wake up the chip from standby mode.
Errata reference: 11951
Fix/Workaround:

11 – When the Peripheral Access Controller (PAC) protection is enabled, writing to WAVE or WAVEB registers will not cause a hardware exception.
Errata reference: 11468
Fix/Workaround:
None

12 – If the MCx flag in the INTFLAG register is set when enabling the DMA, this will trigger an immediate DMA transfer and overwrite the current buffered value in the TCC register.
Errata reference: 12155
Fix/Workaround:
None

40.1.2.13 PTC

1 – WCOMP interrupt flag is not stable. The WCOMP interrupt flag will not always be set as described in the datasheet.
Errata reference: 12860
Fix/Workaround:
Do not use the WCOMP interrupt. Use the WCOMP event.

40.1.3 Die Revision C

40.1.3.1 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 – In single shot mode and at 125°C, the ADC conversions have linearity errors.
Errata reference: 13277
Fix/Workaround:
- Workaround 1: At 125°C, do not use the ADC in single shot mode; use the ADC in free running mode only.
- Workaround 2: At 125°C, use the ADC in single shot mode only with VDDANA > 3V.

3 – In the table ""NVM User Row Mapping"", the WDT Window bitfield default value on silicon is not as specified in the datasheet. The datasheet defines the default value as 0x5, while it is 0xB on silicon.
Errata reference: 13951
Fix/Workaround:
None.

4 – On pin PA24 and PA25 the pull-up and pull-down configuration is not disabled automatically when alternative pin function is enabled except for USB.

MTB renamed from “Memory Trace Buffer” to “Micro Trace Buffer”.
DSU - Device Service Unit
Updated description of Starting CRC32 Calculation. Updated title of Table 13-6. Added Device Selection table to Device Selection bit description the Device Identification register (DID.DEVSEL).
GCLK - Generic Clock Controller
Signal names updated in Device Clocking Diagram, Block Diagram.
PM – Power Manager
Added figure Figure 16-2. Register Summary: Removed CFD bit from INTENCLR, INTENSET and INTFLAG. Added PTC bit to APBCMASK register. Register Description: AHB Mask register (AHBMASK): Full bit names updated. APBC Mask register (APBCMASK.PTC): Added PTC to bit 19. CFD bit removed from INTENCLR, INTENSET and INTFLAG.
SYSCTRL – System Controller
Updated description of 8MHz Internal Oscillator (OSC8M) Operation. FDPLL96M section reorganized and more integrated in the SYSCTRL chapter: Features, Signal Description and Product Dependencies sub sections removed and integrated with the corresponding sections in SYSCTRL. Register Summary: Added VREG register on address 0x3C - 0x3D. Register Description: Updated reset values in OSC8M. Updated CALIB[11:0] bit description in OSC8M. Updated LBYPASS bit description in DPLLCTRLB.
WDT – Watchdog Timer
Updated description in Principle of Operation: Introducing the bits used in Table 18-1. Updated description in Initialization. Updated description in Normal Mode. Updated description in Window Mode. Updated description in Interrupts. WEN bit description in the Control register (CTRL.WEN) updated with information on enable-protection.
RTC – Real-Time Counter

43.16 Rev. A - 02/2014

	Initial revision

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units
SFDR	Spurious Free Dynamic Range	1x Gain F_{CLK_ADC} = 2.1MHz F_{IN} = 40kHz A_{IN} = 95%FSR	63.1	65.0	66.5	dB
SINAD	Signal-to-Noise and Distortion		50.7	59.5	61.0	dB
SNR	Signal-to-Noise Ratio		49.9	60.0	64.0	dB
THD	Total Harmonic Distortion		-65.4	-63.0	-62.1	dB
	Noise RMS	T = 25°C	-	1.0	-	mV

Table 44-18. Single-Ended Mode (Device Variant B)

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units
ENOB	Effective Number of Bits	With gain compensation	-	9.7	10.1	Bits
TUE	Total Unadjusted Error	1x gain	-	7.9	40.0	LSB
INL	Integral Non-Linearity	1x gain	1.4	2.6	6.0	LSB
DNL	Differential Non-Linearity	1x gain	+/-0.6	+/-0.7	+/-0.95	LSB
	Gain Error	Ext. Ref. 1x	-5.0	0.6	+5.0	mV
	Gain Accuracy(4)	Ext. Ref. 0.5x	+/-0.1	+/-0.37	+/-0.55	%
		Ext. Ref. 2x to 16X	+/-0.01	+/-0.1	+/-0.2	%
	Offset Error	Ext. Ref. 1x	-5.0	0.6	+10.0	mV
SFDR	Spurious Free Dynamic Range	1x Gain F_{CLK_ADC} = 2.1MHz F_{IN} = 40kHz A_{IN} = 95%FSR	63.0	68.0	68.7	dB
SINAD	Signal-to-Noise and Distortion		55.0	60.1	62.5	dB
SNR	Signal-to-Noise Ratio		54.0	61.0	64.0	dB
THD	Total Harmonic Distortion		-69.0	-68.0	-65.0	dB
	Noise RMS	T = 25°C	-	1.0	-	mV

Note:

1. Maximum numbers are based on characterization and not tested in production, and for 5% to 95% of the input voltage range.
2. Respect the input common mode voltage through the following equations (where VCM_IN is the Input channel common mode voltage) for all VIN:
   - VCM_IN < 0.7*VDDANA + VREF/4 – 0.75V
   - VCM_IN > VREF/4 – 0.3*VDDANA - 0.1V
3. The ADC channels on pins PA08, PA09, PA10, PA11 are powered from the V_{DDIO} power supply. The ADC performance of these pins will not be the same as all the other ADC channels on pins powered from the VDDANA power supply.
4. The gain accuracy represents the gain error expressed in percent. Gain accuracy (%) = (Gain Error in V x 100) / (V_{ref}/GAIN)

44.6.4 Inputs and Sample and Hold Acquisition Times

The analog voltage source must be able to charge the sample and hold (S/H) capacitor in the ADC in order to achieve maximum accuracy. Seen externally the ADC input consists of a resistor (R_{SAMPLE}) and a

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units
	Current Consumption	f = 2MHz, C_L = 20pF, AGC off	27	65	90	µA
		f = 2MHz, C_L = 20pF, AGC on	14	52	79	
		f = 4MHz, C_L = 20pF, AGC off	61	117	161	
		f = 4MHz, C_L = 20pF, AGC on	23	74	110	
		f = 8MHz, C_L = 20pF, AGC off	131	226	319	
		f = 8MHz, C_L = 20pF, AGC on	56	128	193	
		f = 16MHz, C_L = 20pF, AGC off	305	502	742	
		f = 16MHz, C_L = 20pF, AGC on	116	307	627	
		f = 32MHz, C_L = 18pF, AGC off	1031	1622	2344	
		f = 32MHz, C_L = 18pF, AGC on	278	615	1422	
$t_{STARTUP}$	Start-up time	f = 2MHz, C_L = 20pF, XOSC.GAIN = 0, ESR = 600Ω	-	14K	48K	cycles
		f = 4MHz, C_L = 20pF, XOSC.GAIN = 1, ESR = 100Ω	-	6800	19.5K	
		f = 8 MHz, C_L = 20pF, XOSC.GAIN = 2, ESR = 35Ω	-	5550	13K	
		f = 16 MHz, C_L = 20pF, XOSC.GAIN = 3, ESR = 25Ω	-	6750	14.5K	
		f = 32MHz, C_L = 18pF, XOSC.GAIN = 4, ESR = 40Ω	-	5.3K	9.6K	

Table 44-34. Crystal Oscillator Characteristics (Device Variant B)

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units
f_{OUT}	Crystal oscillator frequency		0.4	-	32	MHz
ESR	Crystal Equivalent Series Resistance Safety Factor = 3 The AGC doesn't have any noticeable impact on these measurements.	f = 0.455 MHz, C_L = 100pF XOSC.GAIN = 0	-	-	5.6K	Ω
		f = 2MHz, C_L = 20pF XOSC.GAIN = 0	-	-	416	
		f = 4MHz, C_L = 20pF XOSC.GAIN = 1	-	-	243	
		f = 8 MHz, C_L = 20pF XOSC.GAIN = 2	-	-	138	
		f = 16 MHz, C_L = 20pF XOSC.GAIN = 3	-	-	66	
		f = 32MHz, C_L = 18pF XOSC.GAIN = 4	-	-	56	
C_{XIN}	Parasitic capacitor load		-	5.9	-	pF