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	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K 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-VFQFN Exposed Pad
Supplier Device Package	48-QFN (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsamd21g17a-mu

- • MASTER: Contains the main trace enable bit and other trace control fields,
- • FLOW: Contains the WATERMARK address and the AUTOSTOP and AUTOHALT control bits,
- • BASE: Indicates where the SRAM is located in the processor memory map. This register is provided to enable auto discovery of the MTB SRAM location, by a debug agent.

See the CoreSight MTB-M0+ Technical Reference Manual for a detailed description of these registers.

11.4 High-Speed Bus System

11.4.1 Features

High-Speed Bus Matrix has the following features:

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

11.4.2 Configuration
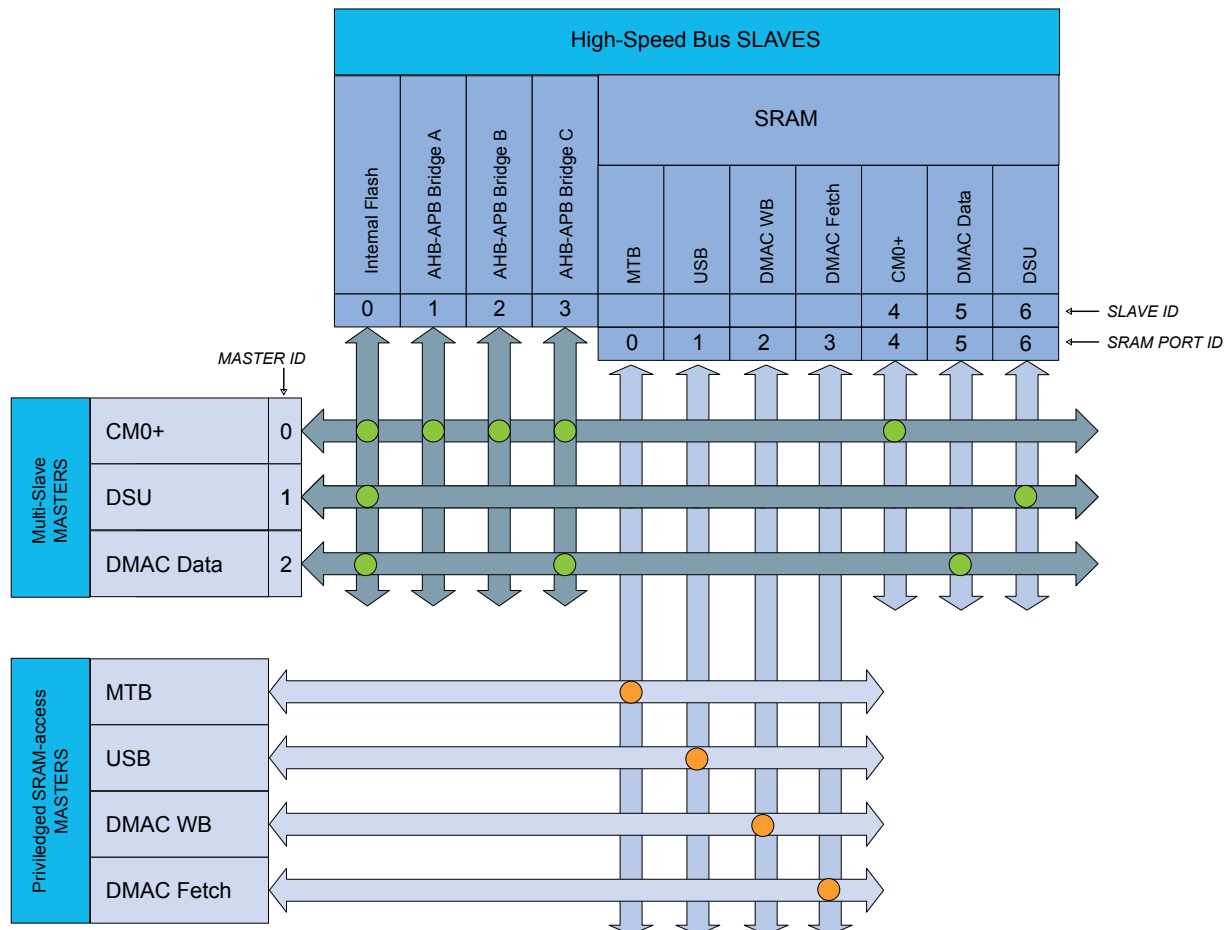
13.13 Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to Register Access Protection.

13.13.1 Control

Name: CTRL
Offset: 0x0000
Reset: 0x00
Property: PAC Write-Protection

Bit	7	6	5	4	3	2	1	0
				CE	MBIST	CRC		SWRST
Access				W	W	W		W
Reset				0	0	0		0

Bit 4 – CE: Chip-Erase
Writing a '0' to this bit has no effect.

Writing a '1' to this bit starts the Chip-Erase operation.

Bit 3 – MBIST: Memory Built-In Self-Test
Writing a '0' to this bit has no effect.

Writing a '1' to this bit starts the memory BIST algorithm.

Bit 2 – CRC: 32-bit Cyclic Redundancy Check
Writing a '0' to this bit has no effect.

Writing a '1' to this bit starts the cyclic redundancy check algorithm.

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

Writing a '1' to this bit resets the module.

13.13.2 Status A

Name: STATUSA
Offset: 0x0001
Reset: 0x00
Property: PAC Write-Protection

Reset: 0x00000010
Property: -

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

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

Bits 7:4 – CCLASS[3:0]: Component Class
These bits will always return 0x1 when read indicating that this ARM CoreSight component is ROM table (refer to the ARM Debug Interface v5 Architecture Specification at http://www.arm.com).

Bits 3:0 – PREAMBLE[3:0]: Preamble
These bits will always return 0x0 when read.

13.13.21 Component Identification 2

Name: CID2
Offset: 0x1FF8
Reset: 0x00000005
Property: -

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

16.5.8 Register Access Protection

Registers with write-access can be optionally write-protected by the Peripheral Access Controller (PAC), except the following:

- Interrupt Flag register (INTFLAG).
- Reset Cause register (RCAUSE).

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

Write-protection does not apply for accesses through an external debugger. Refer to *PAC – Peripheral Access Controller* for details.

Related Links

PAC - Peripheral Access Controller

16.5.9 Analog Connections

Not applicable.

16.6 Functional Description

16.6.1 Principle of Operation

16.6.1.1 Synchronous Clocks

The GCLK_MAIN clock from GCLK module provides the source for the main clock, which is the common root for the synchronous clocks for the CPU and APBx modules. The main clock is divided by an 8-bit prescaler, and each of the derived clocks can run from any tapping off this prescaler or the undivided main clock, as long as $f_{CPU} \geq f_{APBx}$. The synchronous clock source can be changed on the fly to respond to varying load in the application. The clocks for each module in each synchronous clock domain can be individually masked to avoid power consumption in inactive modules. Depending on the sleep mode, some clock domains can be turned off (see Table 16-4).

16.6.1.2 Reset Controller

The Reset Controller collects the various reset sources and generates reset for the device. The device contains a power-on-reset (POR) detector, which keeps the system reset until power is stable. This eliminates the need for external reset circuitry to guarantee stable operation when powering up the device.

16.6.1.3 Sleep Mode Controller

In ACTIVE mode, all clock domains are active, allowing software execution and peripheral operation. The PM Sleep Mode Controller allows the user to choose between different sleep modes depending on application requirements, to save power (see Table 16-4).

16.6.2 Basic Operation

16.6.2.1 Initialization

After a power-on reset, the PM is enabled and the Reset Cause register indicates the POR source (RCAUSE.POR). The default clock source of the GCLK_MAIN clock is started and calibrated before the CPU starts running. The GCLK_MAIN clock is selected as the main clock without any division on the prescaler. The device is in the ACTIVE mode.
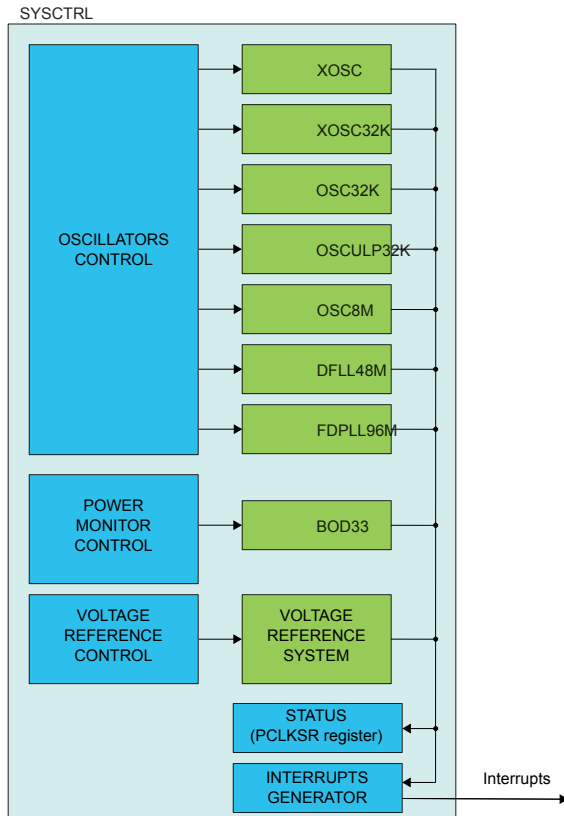
By default, only the necessary clocks are enabled (see Table 16-1).

16.6.2.2 Enabling, Disabling and Resetting

The PM module is always enabled and can not be reset.

17.3   Block Diagram

Figure 17-1. SYSCTRL Block Diagram

17.4   Signal Description

Signal Name	Types	Description
XIN	Analog Input	Multipurpose Crystal Oscillator or external clock generator input
XOUT	Analog Output	External Multipurpose Crystal Oscillator output
XIN32	Analog Input	32kHz Crystal Oscillator or external clock generator input
XOUT32	Analog Output	32kHz Crystal Oscillator output

The I/O lines are automatically selected when XOSC or XOSC32K are enabled. Refer to *Oscillator Pinout*.

Related Links

I/O Multiplexing and Considerations

17.5   Product Dependencies

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

Bit	23	22	21	20	19	18	17	16
							DPLLLTO	DPLLLCKF
Access	R	R	R	R	R	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	DPLLLCKR				B33SRDY	BOD33DET	BOD33RDY	DFLLRCS
Access	R/W	R	R	R	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
	DFLLLCKC	DFLLLCKF	DFLLOOB	DFLLRDY	OSC8MRDY	OSC32KRDY	XOSC32KRDY	XOSCRDY
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 17 – DPLLLTO: DPLL Lock Timeout Interrupt Enable
Writing a zero to this bit has no effect.

Writing a one to this bit will set the DPLL Lock Timeout Interrupt Enable bit, which enables the DPLL Lock Timeout interrupt.

Value	Description
0	The DPLL Lock Timeout interrupt is disabled.
1	The DPLL Lock Timeout interrupt is enabled, and an interrupt request will be generated when the DPLL Lock Timeout Interrupt flag is set.

Bit 16 – DPLLLCKF: DPLL Lock Fall Interrupt Enable
Writing a zero to this bit has no effect.

Writing a one to this bit will set the DPLL Lock Fall Interrupt Enable bit, which enables the DPLL Lock Fall interrupt.

Value	Description
0	The DPLL Lock Fall interrupt is disabled.
1	The DPLL Lock Fall interrupt is enabled, and an interrupt request will be generated when the DPLL Lock Fall Interrupt flag is set.

Bit 15 – DPLLLCKR: DPLL Lock Rise Interrupt Enable
Writing a zero to this bit has no effect.

Writing a one to this bit will set the DPLL Lock Rise Interrupt Enable bit, which enables the DPLL Lock Rise interrupt.

Value	Description
0	The DPLL Lock Rise interrupt is disabled.
1	The DPLL Lock Rise interrupt is enabled, and an interrupt request will be generated when the DPLL Lock Rise Interrupt flag is set.

Bit 11 – B33SRDY: BOD33 Synchronization Ready Interrupt Enable
Writing a zero to this bit has no effect.

Writing a one to this bit will set the BOD33 Synchronization Ready Interrupt Enable bit, which enables the BOD33 Synchronization Ready interrupt.

Bit	7	6	5	4	3	2	1	0
				FINE[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 31:16 – DIFF[15:0]: Multiplication Ratio Difference
In closed-loop mode (DFLLCTRL.MODE is written to one), this bit group indicates the difference between the ideal number of DFLL cycles and the counted number of cycles. This value is not updated in open-loop mode, and should be considered invalid in that case.

Bits 15:10 – COARSE[5:0]: Coarse Value
Set the value of the Coarse Calibration register. In closed-loop mode, this field is read-only.

Bits 9:0 – FINE[9:0]: Fine Value
Set the value of the Fine Calibration register. In closed-loop mode, this field is read-only.

17.8.12 DFLL48M Multiplier

Name: DFLLMUL
Offset: 0x2C
Reset: 0x00000000
Property: Write-Protected

Bit	31	30	29	28	27	26	25	24
			CSTEP[5:0]				FSTEP[9:8]	
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	23	22	21	20	19	18	17	16
				FSTEP[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
				MUL[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
				MUL[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 31:26 – CSTEP[5:0]: Coarse Maximum Step
This bit group indicates the maximum step size allowed during coarse adjustment in closed-loop mode. When adjusting to a new frequency, the expected output frequency overshoot depends on this step size.

Bits 25:16 – FSTEP[9:0]: Fine Maximum Step
This bit group indicates the maximum step size allowed during fine adjustment in closed-loop mode. When adjusting to a new frequency, the expected output frequency overshoot depends on this step size.

22.   NVMCTRL – Non-Volatile Memory Controller

22.1   Overview

Non-Volatile Memory (NVM) is a reprogrammable Flash memory that retains program and data storage even with power off. It embeds a main array and a separate smaller array intended for EEPROM emulation (RWWEE) that can be programmed while reading the main array. The NVM Controller (NVMCTRL) connects to the AHB and APB bus interfaces for system access to the NVM block. The AHB interface is used for reads and writes to the NVM block, while the APB interface is used for commands and configuration.

22.2   Features

- 32-bit AHB interface for reads and writes
- Read While Write EEPROM emulation area
- All NVM sections are memory mapped to the AHB, including calibration and system configuration
- 32-bit APB interface for commands and control
- Programmable wait states for read optimization
- 16 regions can be individually protected or unprotected
- Additional protection for boot loader
- Supports device protection through a security bit
- Interface to Power Manager for power-down of Flash blocks in sleep modes
- Can optionally wake up on exit from sleep or on first access
- Direct-mapped cache

Note: A register with property "Enable-Protected" may contain bits that are *not* enable-protected.

22.3   Block Diagram

Figure 22-1. Block Diagram
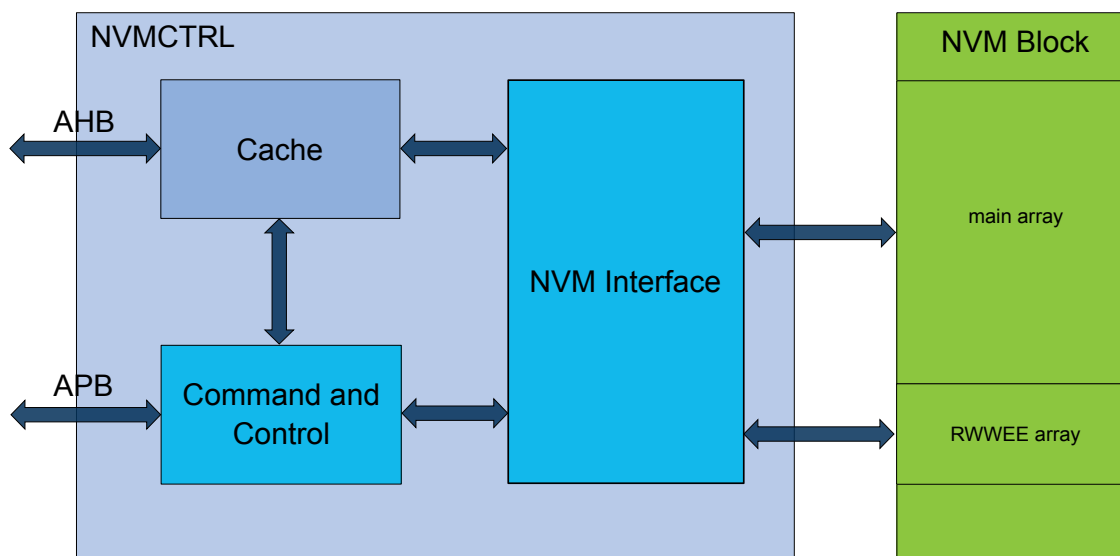
24.5.8   Register Access Protection

Registers with write-access can be optionally write-protected by the Peripheral Access Controller (PAC), except the following:

- Interrupt Flag Status and Clear register (INTFLAG)

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

When the CPU is halted in debug mode, all write-protection is automatically disabled. Write-protection does not apply for accesses through an external debugger.

24.5.9   Analog Connections

Not applicable.

24.6   Functional Description

24.6.1   Principle of Operation

The Event System consists of several channels which route the internal events from peripherals (generators) to other internal peripherals or IO pins (users). Each event generator can be selected as source for multiple channels, but a channel cannot be set to use multiple event generators at the same time.

24.6.2   Basic Operation

24.6.2.1   Initialization

Before enabling events routing within the system, the User Multiplexer (USER) and Channel (CHANNEL) register must be configured. The User Multiplexer (USER) must be configured first.

Configure the User Multiplexer (USER) register:
1. The channel to be connected to a user is written to the Channel bit group (USER.CHANNEL)
2. The user to connect the channel is written to the User bit group (USER.USER)

Configure the Channel (CHANNEL) register:
1. The channel to be configured is written to the Channel Selection bit group (CHANNEL.CHANNEL)
2. The path to be used is written to the Path Selection bit group (CHANNEL.PATH)
3. The type of edge detection to use on the channel is written to the Edge Selection bit group (CHANNEL.EDGSEL)
4. The event generator to be used is written to the Event Generator bit group (CHANNEL.EVGEN)

24.6.2.2   Enabling, Disabling and Resetting

The EVSYS is always enabled.

The EVSYS is reset by writing a '1' to the Software Reset bit in the Control register (CTRL.SWRST). All registers in the EVSYS will be reset to their initial state and all ongoing events will be canceled. Refer to CTRL.SWRST register for details.

24.6.2.3   User Multiplexer Setup

The user multiplexer defines the channel to be connected to which event user. Each user multiplexer is dedicated to one event user. A user multiplexer receives all event channels output and must be configured to select one of these channels, as shown in the next figure. The channel is selected with the Channel bit group in the USER register (USER.CHANNEL). The user multiplexer must always be configured before the channel. A full list of selectable users can be found in the User Multiplexer register (USER) description. Refer to UserList for details.

Table 25-1. SERCOM Modes

CTRLA.MODE	Description
0x0	USART with external clock
0x1	USART with internal clock
0x2	SPI in slave operation
0x3	SPI in master operation
0x4	I^{2}C slave operation
0x5	I^{2}C master operation
0x6-0x7	Reserved

For further initialization information, see the respective SERCOM mode chapters:

Related Links

SERCOM USART – SERCOM Universal Synchronous and Asynchronous Receiver and Transmitter

SERCOM SPI – SERCOM Serial Peripheral Interface

SERCOM I2C – SERCOM Inter-Integrated Circuit

25.6.2.2 Enabling, Disabling, and Resetting

This peripheral is enabled by writing '1' to the Enable bit in the Control A register (CTRLA.ENABLE), and disabled by writing '0' to it.

Writing '1' to the Software Reset bit in the Control A register (CTRLA.SWRST) will reset all registers of this peripheral to their initial states, except the DBGCTRL register, and the peripheral is disabled.

Refer to the CTRLA register description for details.

25.6.2.3 Clock Generation – Baud-Rate Generator

The baud-rate generator, as shown in Figure 25-3, generates internal clocks for asynchronous and synchronous communication. The output frequency (f_{BAUD}) is determined by the Baud register (BAUD) setting and the baud reference frequency (f_{ref}). The baud reference clock is the serial engine clock, and it can be internal or external.

For asynchronous communication, the /16 (divide-by-16) output is used when transmitting, whereas the /1 (divide-by-1) output is used while receiving.

For synchronous communication, the /2 (divide-by-2) output is used.

This functionality is automatically configured, depending on the selected operating mode.

- *D* represent the data bits per frame
- *S* represent the sum of start and first stop bits, if present.

Table 25-3 shows the BAUD register value versus baud frequency f_{BAUD} at a serial engine frequency of 48MHz. This assumes a *D* value of 8 bits and an *S* value of 2 bits (10 bits, including start and stop bits).

Table 25-3. BAUD Register Value vs. Baud Frequency

| BAUD Register Value	Serial Engine CPF	f_{BAUD} at 48MHz Serial Engine Frequency (f_{REF})
0 – 406	160	3MHz
407 – 808	161	2.981MHz
809 – 1205	162	2.963MHz
...	...	...
65206	31775	15.11kHz
65207	31871	15.06kHz
65208	31969	15.01kHz

25.6.3   Additional Features

25.6.3.1   Address Match and Mask

The SERCOM address match and mask feature is capable of matching either one address, two unique addresses, or a range of addresses with a mask, based on the mode selected. The match uses seven or eight bits, depending on the mode.

Address With Mask

An address written to the Address bits in the Address register (ADDR.ADDR), and a mask written to the Address Mask bits in the Address register (ADDR.ADDRMASK) will yield an address match. All bits that are masked are not included in the match. Note that writing the ADDR.ADDRMASK to 'all zeros' will match a single unique address, while writing ADDR.ADDRMASK to 'all ones' will result in all addresses being accepted.

Figure 25-4. Address With Mask
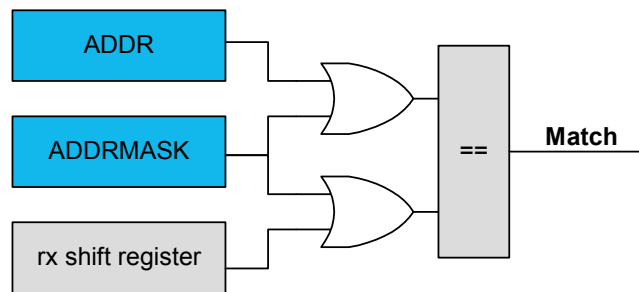
Two Unique Addresses

The two addresses written to ADDR and ADDRMASK will cause a match.

26.6.2.3 Clock Generation and Selection

For both synchronous and asynchronous modes, the clock used for shifting and sampling data can be generated internally by the SERCOM baud-rate generator or supplied externally through the XCK line.

The synchronous mode is selected by writing a '1' to the Communication Mode bit in the Control A register (CTRLA.CMODE), the asynchronous mode is selected by writing a zero to CTRLA.CMODE.

The internal clock source is selected by writing 0x1 to the Operation Mode bit field in the Control A register (CTRLA.MODE), the external clock source is selected by writing 0x0 to CTRLA.MODE.

The SERCOM baud-rate generator is configured as in the figure below.

In asynchronous mode (CTRLA.CMODE=0), the 16-bit Baud register value is used.

In synchronous mode (CTRLA.CMODE=1), the eight LSBs of the Baud register are used. Refer to *Clock Generation – Baud-Rate Generator* for details on configuring the baud rate.

Figure 26-3. Clock Generation
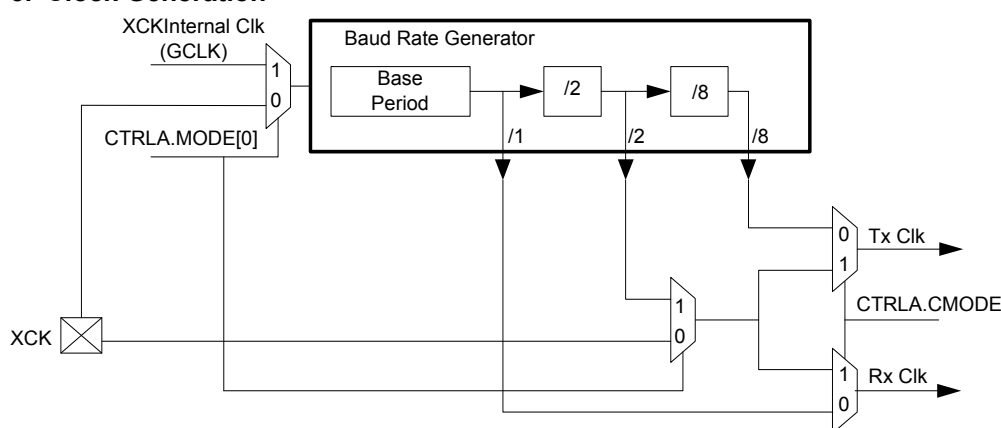
Related Links

Clock Generation – Baud-Rate Generator

Asynchronous Arithmetic Mode BAUD Value Selection

Synchronous Clock Operation

In synchronous mode, the CTRLA.MODE bit field determines whether the transmission clock line (XCK) serves either as input or output. The dependency between clock edges, data sampling, and data change is the same for internal and external clocks. Data input on the RxD pin is sampled at the opposite XCK clock edge when data is driven on the TxD pin.

The Clock Polarity bit in the Control A register (CTRLA.CPOL) selects which XCK clock edge is used for RxD sampling, and which is used for TxD change:

When CTRLA.CPOL is '0', the data will be changed on the rising edge of XCK, and sampled on the falling edge of XCK.

When CTRLA.CPOL is '1', the data will be changed on the falling edge of XCK, and sampled on the rising edge of XCK.

Name: CCx
Offset: 0x18+i*0x2 [i=0..1]
Reset: 0x0000
Property: Write-Synchronized

Bit	15	14	13	12	11	10	9	8
				CC[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
				CC[7:0]				
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 15:0 – CC[15:0]: Channel x Compare/Capture Value
These bits contain the compare/capture value in 16-bit TC mode. In Match frequency (MFRQ) or Match PWM (MPWM) waveform operation (CTRLA.WAVEGEN), the CC0 register is used as a period register.

30.8.14.3 Channel x Compare/Capture Value, 32-bit Mode

Name: CCx
Offset: 0x18+i*0x4 [i=0..1]
Reset: 0x00000000
Property: Write-Synchronized

Bit	31	30	29	28	27	26	25	24
				CC[31:24]				
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	23	22	21	20	19	18	17	16
				CC[23:16]				
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
				CC[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
				CC[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 – CC[31:0]: Channel x Compare/Capture Value
These bits contain the compare/capture value in 32-bit TC mode. In Match frequency (MFRQ) or Match PWM (MPWM) waveform operation (CTRLA.WAVEGEN), the CC0 register is used as a period register.

31. TCC – Timer/Counter for Control Applications

31.1 Overview

The device provides three instances of the Timer/Counter for Control applications (TCC) peripheral, TCC[2:0].

Each TCC instance consists of a counter, a prescaler, compare/capture channels and control logic. The counter can be set to count events or clock pulses. The counter together with the compare/capture channels can be configured to time stamp input events, allowing capture of frequency and pulse-width. It can also perform waveform generation such as frequency generation and pulse-width modulation.

Waveform extensions are intended for motor control, ballast, LED, H-bridge, power converters, and other types of power control applications. They allow for low- and high-side output with optional dead-time insertion. Waveform extensions can also generate a synchronized bit pattern across the waveform output pins. The fault options enable fault protection for safe and deterministic handling, disabling and/or shut down of external drivers.

Figure 31-1 shows all features in TCC.

Related Links

TCC Configurations

31.2 Features

- Up to four compare/capture channels (CC) with:
    - Double buffered period setting
    - Double buffered compare or capture channel
    - Circular buffer on period and compare channel registers
- Waveform generation:
    - Frequency generation
    - Single-slope pulse-width modulation (PWM)
    - Dual-slope pulse-width modulation with half-cycle reload capability
- Input capture:
    - Event capture
    - Frequency capture
    - Pulse-width capture
- Waveform extensions:
    - Configurable distribution of compare channels outputs across port pins
    - Low- and high-side output with programmable dead-time insertion
    - Waveform swap option with double buffer support
    - Pattern generation with double buffer support
    - Dithering support
- Fault protection for safe disabling of drivers:
    - Two recoverable fault sources
    - Two non-recoverable fault sources
    - Debugger can be source of non-recoverable fault

31.8.14 Counter Value

Note: Prior to any read access, this register must be synchronized by user by writing the according TCC Command value to the Control B Set register (CTRLBSET.CMD=READSYNC).

Name: COUNT
Offset: 0x34
Reset: 0x00000000
Property: PAC Write-Protection, Write-Synchronized, Read-Synchronized

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
				COUNT[23:16]				
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
				COUNT[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
				COUNT[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 23:0 – COUNT[23:0]: Counter Value
These bits hold the value of the counter register.

Note: When the TCC is configured as 16-bit timer/counter, the excess bits are read zero.

Note: This bit field occupies the MSB of the register, [23:m]. m is dependent on the Resolution bit in the Control A register (CTRLA.RESOLUTION):

CTRLA.RESOLUTION	Bits [23:m]
0x0 - NONE	23:0 (depicted)
0x1 - DITH4	23:4
0x2 - DITH5	23:5
0x3 - DITH6	23:6

31.8.15 Pattern

Name: PATT
Offset: 0x38
Reset: 0x0000

least 20 ms. It is required to first write a one to the Send USB Resume bit in CTRLB (CTRLB.RESUME) to respond to the upstream resume with a downstream resume. Alternatively, the host can resume from a suspend state by sending a Downstream Resume on the USB bus (CTRLB.RESUME set to 1). In both cases, when the downstream resume is completed, the CTRLB.SOFE bit is automatically set and the host enters again the active state.

32.6.3.8 Phase-locked SOFs

To support the Synchronous Endpoints capability, the period of the emitted Start-of-Frame is maintained while the USB connection is not in the active state. This does not apply for the disconnected/connected/ reset states. It applies for active/idle/suspend/resume states. The period of Start-of-Frame will be 1ms when the USB connection is in active state and an integer number of milli-seconds across idle/suspend/ resume states.

To ensure the Synchronous Endpoints capability, the GCLK_USB clock must be kept running. If the GCLK_USB is interrupted, the period of the emitted Start-of-Frame will be erratic.

32.6.3.9 Management of Control Pipes

A control transaction is composed of three stages:

- SETUP
- Data (IN or OUT)
- Status (IN or OUT)

The user has to change the pipe token according to each stage using the Pipe Token field in PCFG (PCFG.PTOKEN).

For control pipes only, the token is assigned a specific initial data toggle sequence:

- SETUP: Data0
- IN: Data1
- OUT: Data1

32.6.3.10 Management of IN Pipes

IN packets are sent by the USB device controller upon IN request reception from the host. All the received data from the device to the host will be stored in the bank provided the bank is empty. The pipe and its descriptor in RAM must be configured.

The host indicates it is able to receive data from the device by clearing the Bank 0/1 Ready bit in PSTATUS (PSTATUS.BK0/1RDY), which means that the memory for the bank is available for new USB transfer.

The USB will perform IN requests as long as the pipe is not frozen by the user.

The generation of IN requests starts when the pipe is unfrozen (PSTATUS.PFREEZE is set to zero).

When the current bank is full, the Transmit Complete 0/1 bit in PINTFLAG (PINTFLAG.TRCPT0/1) will be set and trigger an interrupt if enabled and the PSTATUS.BK0/1RDY bit will be set.

PINTFLAG.TRCPT0/1 must be cleared by software to acknowledge the interrupt. This is done by writing a one to the PINTFLAG.TRCPT0/1 of the addressed pipe.

The user reads the PCKSIZE.BYTE_COUNT to know how many bytes should be read.

To free the bank the user must read the IN data from the address ADDR in the pipe descriptor and clear the PKSTATUS.BK0/1RDY bit. When the IN pipe is composed of multiple banks, a successful IN transaction will switch to the next bank. Another IN request will be performed by the host as long as the PSTATUS.BK0/1RDY bit for that bank is set. The PINTFLAG.TRCPT0/1 and PSTATUS.BK0/1RDY will be updated accordingly.

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

Bit 1 – ENABLE: Synchronization Enable status bit
This bit is cleared when the synchronization of ENABLE register between the clock domains is complete.

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

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

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

32.8.1.3 QOS Control

Name: QOSCTRL
Offset: 0x03
Reset: 0x05
Property: PAC Write-Protection

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

Bits 3:2 – DQOS[1:0]: Data Quality of Service
These bits define the memory priority access during the endpoint or pipe read/write data operation. Refer to *SRAM Quality of Service*.

Bits 1:0 – CQOS[1:0]: Configuration Quality of Service
These bits define the memory priority access during the endpoint or pipe read/write configuration operation. Refer to *SRAM Quality of Service*.

32.8.1.4 Finite State Machine Status

Name: FSMSTATUS
Offset: 0x0D
Reset: 0xXXXX
Property: Read only

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

Bits 6:0 – FSMSTATE[6:0]: Fine State Machine Status
These bits indicate the state of the finite state machine of the USB controller.

Bit	7	6	5	4	3	2	1	0
	BK1RDY	BK0RDY	STALLRQ1	STALLRQ0		CURBK	DTGLIN	DTGLOUT
Access	W	W	W	W		W	W	W
Reset	0	0	0	0		0	0	0

Bit 7 – BK1RDY: Bank 1 Ready Clear
Writing a zero to this bit has no effect.

Writing a one to this bit will clear EPSTATUS.BK1RDY bit.

Bit 6 – BK0RDY: Bank 0 Ready Clear
Writing a zero to this bit has no effect.

Writing a one to this bit will clear EPSTATUS.BK0RDY bit.

Bit 5 – STALLRQ1: STALL bank 1 Request Clear
Writing a zero to this bit has no effect.

Writing a one to this bit will clear EPSTATUS.STALLRQ1 bit.

Bit 4 – STALLRQ0: STALL bank 0 Request Clear
Writing a zero to this bit has no effect.

Writing a one to this bit will clear EPSTATUS.STALLRQ0 bit.

Bit 2 – CURBK: Current Bank Clear
Writing a zero to this bit has no effect.

Writing a one to this bit will clear EPSTATUS.CURBK bit.

Bit 1 – DTGLIN: Data Toggle IN Clear
Writing a zero to this bit has no effect.

Writing a one to this bit will clear EPSTATUS.DTGLIN bit.

Bit 0 – DTGLOUT: Data Toggle OUT Clear
Writing a zero to this bit has no effect.

Writing a one to this bit will clear the EPSTATUS.DTGLOUT bit.

32.8.3.3 EndPoint Status Set n

Name: EPSTATUSSETn
Offset: 0x105 + (n x 0x20)
Reset: 0x00
Property: PAC Write-Protection

Bit	7	6	5	4	3	2	1	0
	BK1RDY	BK0RDY	STALLRQ1	STALLRQ0		CURBK	DTGLIN	DTGLOUT
Access	W	W	W	W		W	W	W
Reset	0	0	0	0		0	0	0

Bit 7 – BK1RDY: Bank 1 Ready Set
Writing a zero to this bit has no effect.

Property: PAC Write-Protection

Bit	7	6	5	4	3	2	1	0
			STALL	RXSTP		TRFAIL		TRCPT
Access			R/W	R/W		R/W		R/W
Reset			0	0		0		0

Bit 5 – STALL: Transmit STALL x Interrupt Enable
Writing a zero to this bit has no effect.

Writing a one to this bit will clear the Transmit Stall x Interrupt Enable bit and disable the corresponding interrupt request.

Value	Description
0	The Transmit Stall x interrupt is disabled.
1	The Transmit Stall x interrupt is enabled and an interrupt request will be generated when the Transmit Stall x Interrupt Flag is set.

Bit 4 – RXSTP: Received Setup Interrupt Enable
Writing a zero to this bit has no effect.

Writing a one to this bit will clear the Received Setup Interrupt Enable bit and disable the corresponding interrupt request.

Value	Description
0	The Received Setup interrupt is disabled.
1	The Received Setup interrupt is enabled and an interrupt request will be generated when the Received Setup Interrupt Flag is set.

Bit 2 – TRFAIL: Transfer Fail x Interrupt Enable
The user should look into the descriptor table status located in ram to be informed about the error condition : ERRORFLOW, CRC.

Writing a zero to this bit has no effect.

Writing a one to this bit will clear the Transfer Fail x Interrupt Enable bit and disable the corresponding interrupt request.

Value	Description
0	The Transfer Fail bank x interrupt is disabled.
1	The Transfer Fail bank x interrupt is enabled and an interrupt request will be generated when the Transfer Fail x Interrupt Flag is set.

Bit 0 – TRCPT: Transfer Complete x interrupt Enable
Writing a zero to this bit has no effect.

Writing a one to this bit will clear the Transfer Complete x interrupt Enable bit and disable the corresponding interrupt request.

Value	Description
0	The Transfer Complete bank x interrupt is disabled.
1	The Transfer Complete bank x interrupt is enabled and an interrupt request will be generated when the Transfer Complete x Interrupt Flag is set.

3 – Advance capture mode (CAPTMIN CAPTMAX LOCMIN LOCMAX DERIV0) doesn't work if an upper channel is not in one of these mode. Example: when CC[0]=CAPTMIN, CC[1]=CAPTMAX, CC[2]=CAPTEN, and CC[3]=CAPTEN, CAPTMIN and CAPTMAX won't work.
Errata reference: 14817
Fix/Workaround:
Basic capture mode must be set in lower channel and advance capture mode in upper channel.
Example: CC[0]=CAPTEN , CC[1]=CAPTEN , CC[2]=CAPTMIN, CC[3]=CAPTMAX
All capture will be done as expected.

40.3   Device Variant C

The device variant (last letter of the ordering number) is independent of the die revision (DSU.DID.REVISION): The device variant denotes functional differences, whereas the die revision marks evolution of the die.

40.3.1   Die Revision F

40.3.1.1   Device

1 – The SYSTICK calibration value is incorrect.
Errata reference: 14155
Fix/Workaround:
The correct SYSTICK calibration value is 0x40000000. This value should not be used to initialize the Systick RELOAD value register, which should be initialized instead with a value depending on the main clock frequency and on the tick period required by the application. For a detailed description of the SYSTICK module, refer to the official ARM Cortex-M0+ documentation.

2 – 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.
Errata reference: 12368
Fix/Workaround:
For pin PA24 and PA25, the GPIO pull-up and pull-down must be disabled before enabling alternative functions on them.

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 – If the external XOSC32K is broken, neither the external pin RST nor the GCLK software reset can reset the GCLK generators using XOSC32K as source clock.
Errata reference: 12164
Fix/Workaround: