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®-M4F
Core Size	32-Bit Single-Core
Speed	120MHz
Connectivity	EBI/EMI, Ethernet, I²C, IrDA, LINbus, MMC/SD, QSPI, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, I²S, POR, PWM, WDT
Number of I/O	81
Program Memory Size	512KB (512K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	192K x 8
Voltage - Supply (Vcc/Vdd)	1.71V ~ 3.63V
Data Converters	A/D 28x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-TQFP
Supplier Device Package	100-TQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsame53n19a-aut

Value	Description
0	The output is not enabled.
1	The output is enabled and driven by the SUPC.

21.12.14 Tamper Control

Name: TAMPCTRL
Offset: 0x60
Reset: 0x00000000
Property: PAC Write-Protection, Enable-Protected

Bit	31	30	29	28	27	26	25	24
				DEBNC4	DEBNC3	DEBNC2	DEBNC1	DEBNC0
Access								
Reset				0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
				TAMLVL4	TAMLVL3	TAMLVL2	TAMLVL1	TAMLVL0
Access								
Reset				0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
							IN4ACT[1:0]	
Access								
Reset							0	0

Bit	7	6	5	4	3	2	1	0
	IN3ACT[1:0]		IN2ACT[1:0]		IN1ACT[1:0]		IN0ACT[1:0]	
Access								
Reset	0	0	0	0	0	0	0	0

Bits 24, 25, 26, 27, 28 – DEBNC Debounce Enable of Tamper Input INn

Value	Description
0	Debouncing is disabled for Tamper input INn
1	Debouncing is enabled for Tamper input INn

Bits 16, 17, 18, 19, 20 – TAMLVL Tamper Level Select of Tamper Input INn

Value	Description
0	A falling edge condition will be detected on Tamper input INn.
1	A rising edge condition will be detected on Tamper input INn.

Bits 0:1, 2:3, 4:5, 6:7, 8:9 – INACT Tamper Channel n Action
These bits determine the action taken by Tamper Channel n.

Value	Name	Description
0x0	OFF	Off (Disabled)
0x1	WAKE	Wake and set Tamper flag
0x2	CAPTURE	Capture timestamp and set Tamper flag
0x3	ACTL	Compare RTC signal routed between INn and OUT pins . When a mismatch occurs, capture timestamp and set Tamper flag

22.8.11 Busy Channels

Name: BUSYCH
Offset: 0x28
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
	BUSYCHn[31:24]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	BUSYCHn[23:16]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	BUSYCHn[15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

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

Bits 31:0 – BUSYCHn[31:0] Busy Channel n [x=31..0]
This bit is cleared when the channel trigger action for DMA channel n is complete, when a bus error for DMA channel n is detected, or when DMA channel n is disabled.

This bit is set when DMA channel n starts a DMA transfer.

24.9.16 GMAC Transmit Pause Quantum Register

Name: TPQ
Offset: 0x03C
Reset: 0x0000FFFF
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
	TPQ[15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

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

Bits 15:0 – TPQ[15:0] Transmit Pause Quantum
Written with the pause quantum value for pause frame transmission.

24.9.35 GMAC PTP Event Frame Received Seconds High Register

Name: EFRSH
Offset: 0x0EC
Reset: 0x00000000
Property: Read-only

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	RUD[15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

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

Bits 15:0 – RUD[15:0] Register Update
The register is updated with the value that the IEEE 1588 timer seconds register held when the SFD of a PTP transmit primary event crosses the MII interface. An interrupt is issued when the register is updated.

Writing a zero to this bit has no effect.

Writing a '1' to this bit will clear the DFLL Ready Interrupt Enable bit, which disables the DFLL Ready interrupt.

Bits 2, 3 – XOSCFAIL XOSC n Clock Failure Interrupt Enable
0: The XOSC n Clock Failure interrupt is disabled.

1: The XOSC0 Clock Failure interrupt is enabled, and an interrupt request will be generated when the XOSC0 Clock Failure Interrupt flag is set.

Writing a zero to this bit has no effect.

Writing a '1' to this bit will clear the XOSC n Clock Failure Interrupt Enable bit, which disables the XOSC n Clock Failure interrupt.

Bits 0, 1 – XOSCRDY XOSC n Ready Interrupt Enable
0: The XOSC n Ready interrupt is disabled.

1: The XOSC0 Ready interrupt is enabled, and an interrupt request will be generated when the XOSC n Ready Interrupt flag is set.

Writing a zero to this bit has no effect.

Writing a '1' to this bit will clear the XOSC n Ready Interrupt Enable bit, which disables the XOSC n Ready interrupt.

31.6 Register Summary

Offset	Name	Bit Pos.								
0x00	CTRLA	7:0								SWRST
0x01 ... 0x03	Reserved									
0x04	SWEVT	7:0	CHANNEL[7:0]							
		15:8	CHANNEL[15:8]							
		23:16	CHANNEL[23:16]							
		31:24	CHANNEL[31:24]							
0x08	PRICTRL	7:0	RREN			PRI[4:0]				
0x09 ... 0x0F	Reserved									
0x10	INTPEND	7:0				ID[4:0]				
		15:8	BUSY	READY					EVD	OVR
0x12 ... 0x13	Reserved									
0x14	INTSTATUS	7:0	CHINT7	CHINT6	CHINT5	CHINT4	CHINT3	CHINT2	CHINT1	CHINT0
		15:8					CHINT11	CHINT10	CHINT9	CHINT8
		23:16								
		31:24								
0x18	BUSYCH	7:0	BUSYCHx7	BUSYCHx6	BUSYCHx5	BUSYCHx4	BUSYCHx3	BUSYCHx2	BUSYCHx1	BUSYCHx0
		15:8					BUSYCHx11	BUSYCHx10	BUSYCHx9	BUSYCHx8
		23:16								
		31:24								
0x1C	READYUSR	7:0	READYUSR7	READYUSR6	READYUSR5	READYUSR4	READYUSR3	READYUSR2	READYUSR1	READYUSR0
		15:8					READYUSR11	READYUSR10	READYUSR9	READYUSR8
		23:16								
		31:24								
0x20	CHANNEL0	7:0	EVGEN[7:0]							
		15:8	ONDEMAND	RUNSTDBY			EDGSEL[1:0]		PATH[1:0]	
		23:16								
		31:24								
0x24	CHINTENCLR0	7:0							EVD	OVR
0x25	CHINTENSET0	7:0							EVD	OVR
0x26	CHINTFLAG0	7:0							EVD	OVR
0x27	CHSTATUS0	7:0							BUSYCH	RDYUSR
0x28	CHANNEL1	7:0	EVGEN[7:0]							
		15:8	ONDEMAND	RUNSTDBY			EDGSEL[1:0]		PATH[1:0]	
		23:16								
		31:24								
0x2C	CHINTENCLR1	7:0							EVD	OVR

32.9.7 Data Output Value Set

Name: OUTSET
Offset: 0x18
Reset: 0x00000000
Property: PAC Write-Protection

This register allows the user to set one or more output I/O pin drive levels high, without doing a read-modify-write operation. Changes in this register will also be reflected in the Data Output Value (OUT), Data Output Value Toggle (OUTTGL) and Data Output Value Clear (OUTCLR) registers.

Tip: The I/O pins are assembled in pin groups (”PORT groups”) with up to 32 pins. Group 0 consists of the PA pins, group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

Bit	31	30	29	28	27	26	25	24
	OUTSET[31:24]							
Access	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	OUTSET[23:16]							
Access	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	OUTSET[15:8]							
Access	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	OUTSET[7:0]							
Access	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – OUTSET[31:0] PORT Data Output Value Set
Writing '0' to a bit has no effect.

Writing '1' to a bit will set the corresponding bit in the OUT register, which sets the output drive level high for I/O pins configured as outputs via the Data Direction register (DIR). For pins configured as inputs via Data Direction register (DIR) with pull enabled via the Pull Enable register (PULLEN), these bits will set the input pull direction to an internal pull-up.

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.

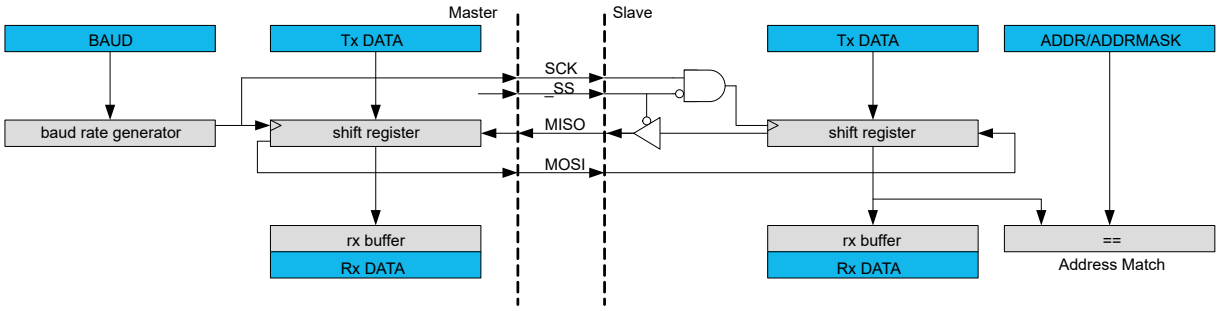
Value	Description
0	Enable synchronization is not busy.
1	Enable synchronization is busy.

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.

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

35.3 Block Diagram

Figure 35-1. Full-Duplex SPI Master Slave Interconnection



35.4 Signal Description

Table 35-1. SERCOM SPI Signals

Signal Name	Type	Description
PAD[3:0]	Digital I/O	General SERCOM pins

One signal can be mapped to one of several pins.

Related Links

6. I/O Multiplexing and Considerations

35.5 Product Dependencies

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

35.5.1 I/O Lines

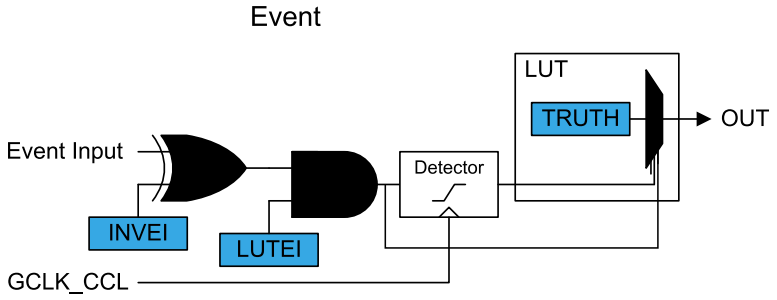
In order to use the SERCOM's I/O lines, the I/O pins must be configured using the IO Pin Controller (PORT).

When the SERCOM is configured for SPI operation, the SERCOM controls the direction and value of the I/O pins according to the table below. Both PORT control bits PINCFGn.PULLEN and PINCFGn.DRVSTR are still effective. If the receiver is disabled, the data input pin can be used for other purposes. In master mode, the slave select line ($\overline{SS}$) is hardware controlled when the Master Slave Select Enable bit in the Control B register (CTRLB.MSSEN) is '1'.

Table 35-2. SPI Pin Configuration

Pin	Master SPI	Slave SPI
MOSI	Output	Input
MISO	Input	Output
SCK	Output	Input
$\overline{SS}$	Output (CTRLB.MSSEN=1)	Input

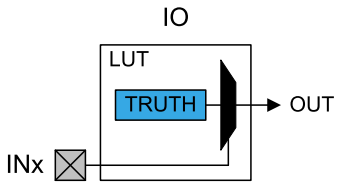
Figure 41-6. Event Input Selection



I/O Pin Inputs (IO)

When the IO pin is selected as LUT input (LUTCTRLx.INSELy=IO), the corresponding LUT input will be connected to the pin, as shown in the figure below.

Figure 41-7. I/O Pin Input Selection



Analog Comparator Inputs (AC)

The AC outputs can be used as input source for the LUT (LUTCTRLx.INSELy=AC).

The analog comparator outputs are distributed following the formula:

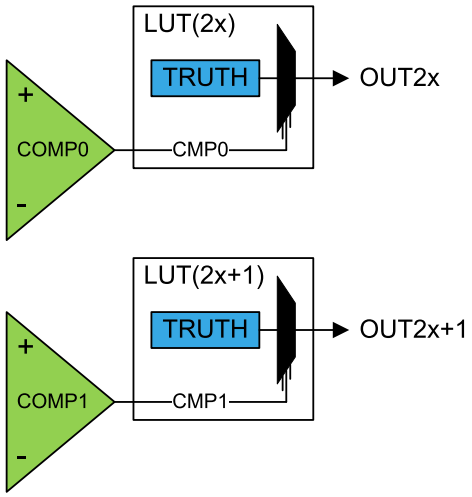
```
IN[N][i]=AC[N % ComparatorOutput_Number]
```

With *N* representing the LUT number and *i*=[0,1,2] representing the LUT input index.

Before selecting the comparator output, the AC must be configured first.

The output of comparator 0 is available on even LUTs ("LUT(2x)": LUT0, LUT2) and the comparator 1 output is available on odd LUTs ("LUT(2x+1)": LUT1, LUT3), as shown in the figure below.

Figure 41-8. AC Input Selection



```
// vPUKCL_Process() is a macro command, which populates the service name
// and then calls the library...
vPUKCL_Process(FastCopy,pvPUKCLParam);
if (PUKCL(u2Status) == PUKCL_OK)
            {
            ...
            }
else // Manage the error
```

43.3.4.5.6 Constraints

The parameter placements that are not allowed are are as follows.

If nu1XBase equals zero, no checks are made on nu1XBase (fixed) and u2XLength (unused).

The following conditions must be avoided to ensure that the service works correctly:

- nu1XBase or nu1RBase are not aligned on 32-bit boundaries
- u2XLength or u2RLength is either: <4, >0xffc or not a 32-bit length or u2XLength >u2RLength
- {nu1XBase, u2XLength} or {nu1RBase, u2RLength} do not entirely lie in Crypto RAM
- {nu1XBase, u2XLength} overlaps {nu1RBase,u2RLength}

43.3.4.5.7 Status Returned Values

Table 43-14. FastCopy Service Return Codes

Returned status	Importance	Meaning
PUKCL_OK	–	Service functioned correctly

43.3.4.6 Conditional Copy/Clear

43.3.4.6.1 Purpose

This service conditionally performs a copy from a memory area to another or a memory area clear.

43.3.4.6.2 How to Use the Service

43.3.4.6.3 Description

This service copies a number X into another number R, padding with zero on the MSB side up to the length specified for R. This copy operation is performed under the conditions specified in the options.

If the condition is verified, R = X.

The copy or clear action is made under condition.

The four possible options for the condition are described in the following table. Two of the conditions check the Specific.CarryIn bit (see 43.3.3.2 Accessing Different Library Services).

The processing is done as follows:

- If the condition is not verified, nothing is processed.
- If the condition is verified the copy or clear follows the rules:
  - If the lengths of R and X are equal, a complete fast copy is processed
  - If the length of R is strictly greater than the length of X, X is first copied in the Low Significant Bytes side of R, and R is padded with zeros on the Most Significant Bytes side.
  - If the pointer on the X area equals zero, R is filled with zeros.

The service name for this operation is `CondCopy`.

43.3.4.7.2 How to Use the Service

43.3.4.7.3 Description

This service processes the following operation (if not computing a modular reduction of the result):

$R = [Z] \pm (MulValue \times X + CarryOperand)$

Or (if computing a modular reduction of the result):

$R = ([Z] \pm (MulValue \times X + CarryOperand)) mod\ N$

The service name for this operation is `Smult`.

The result of the Small Multiply Operation is stored on u2RLength bytes, so the choice of this length compared to u2XLength may lead to:

- A truncation if the result is too big to be stored on u2RLengthbytes.
- A padding on the MSB side if the result does not take all the u2RLengthbytes.
  However, in all cases this rule must be followed:

Important: The length of R must be greater than or equal to the length of X.

In these computations, the following parameters need to be provided:

- R the result (pointed by{nu1RBase,u2Rlength})
- X one input number or GF(2^n) polynomial (pointed by{nu1XBase,u2XLength})
- Z one optional input number or GF(2^n) polynomial (pointed by{nu1ZBase,u2Rlength}).
- MulValue one input number or GF(2^n)polynomial on one word (provided in u4MulValue)
- CarryOperand (provided through the CarryOptions and Carry values).

Important: Even if neither accumulation nor subtraction is specified, the nu1ZBase must always be filled and point to a Crypto RAM space. It this case, nu1ZBase can point to the same space as the nu1RBase.

If using the modular reduction option, the Multiply operation is followed by a reduction (see 43.3.5.1 Modular Reduction) and the following parameters must be additionally provided:

- N—the modulus (pointed by {nu1ModBase,u2Modlength +4})
- Cns—the reduction constant
  - In case of Big reduction, Cns is pointed by {nu1CnsBase,64bytes}.
  - In case of Fast or Normalized reduction, Cns is pointed by {nu1CnsBase,u2ModLength +8}

Modular Reduction Form	Input Dynamic	Result Dynamic	Comments
	GF(2n): Input < ((P[x])^2) * (X^{32})		
Normalized	InputLength < NLength + 4 bytes	GF(p): 0 ≤ Res < N GF(2^n): Res < P[X]	The correction step does not runs in constant time. Needs a precomputed constant. The Normalize function cannot be applied to the product of two numbers of length u2NLength.
Using Euclidean division	InputLength < 2 * NLength + 4 bytes	GF(p): 0 ≤ Res < N GF(2^n): Res < P[X]	Does not need any precomputed constant.

To be able to use these modular reduction services (except the Euclidean division), first the implementer shall call the setup service, providing the modulus as well as one free memory space for the constant (this constant is used to speed up the modular reduction). In most commands (except the modular exponentiation), the quotient is stored in the high order bytes of the number to be reduced, using only eight bytes more than the maximum size of the number to be reduced.

The following rules must be respected to ensure the modular reduction services function correctly:

- The numbers to be reduced can have any significant length, given the fact it CANNOT BE GREATER than 2*u2ModLength + 4 bytes.
- The modulus SHALL ALWAYS HAVE a significant length of <u2ModLength> bytes. The modulus must be provided as a <u2ModLength + 4> bytes long number, padded on the most significant side with a 32-bit word cleared to zero. Not respecting this rule leads to unexpected and wrong results from the modular reduction.
- The normalization operation ALWAYS performs a modular reduction step, and will therefore have the same memory usage as this one.
- The very first operation before any modular operation SHALL BE a modular setup.

43.3.5.1 Modular Reduction

43.3.5.1.1 Purpose

This service is used to perform the various steps necessary to perform a modular reduction and accepts as input numbers in GF(p) or polynomials in GF(2^n) .

The available options for this service are:

- Work in the GF(2^n) or in the standard integer arithmetic field GF(p)
- Operation is the generation of the reduction constant.
- Operation is a Modular Reduction.
- Operation is a Normalization.

43.3.5.1.2 How to Use the Service

43.3.5.1.3 Description

This service performs one of the following operations:

- Control A (CTRLA), except ENABLE and SWRST bits
- Event Control register (EVCTRL)
- Calibration register (CALIB)

Enable-protection is denoted by the "Enable-Protected" property in the register description.

45.6.2.2 Enabling, Disabling, and Resetting

The ADC is enabled by writing a '1' to the Enable bit in the Control A register (CTRLA.ENABLE). The ADC is disabled by writing CTRLA.ENABLE=0.

The ADC is reset by writing a '1' to the Software Reset bit in the Control A register (CTRLA.SWRST). All registers in the ADC, except DBGCTRL, will be reset to their initial state, and the ADC will be disabled. Refer to 45.8.1 CTRLA for details.

45.6.2.3 Operation

In the most basic configuration, the ADC samples values from the configured internal or external sources (INPUTCTRL register). The rate of the conversion depends on the combination of the GCLK_ADCx frequency and the clock prescaler.

To convert analog values to digital values, the ADC needs to be initialized first, as described in the Initialization section. Data conversion can be started either manually by setting the Start bit in the Software Trigger register (SWTRIG.START=1), or automatically by configuring an automatic trigger to initiate the conversions. The ADC starts sampling the input only after the start of conversion is triggered. This means that even after the MUX selection is made, sample and hold (S&H) operation starts only on the conversion trigger. A free-running mode can be used to continuously convert an input channel. When using free-running mode the first conversion must be started, while subsequent conversions will start automatically at the end of previous conversions.

The ADC starts sampling the input only after the start of a conversion is triggered. This means that even after the MUX selection is made, sample and hold operation starts only on the conversion trigger.

The result of the conversion is stored in the Result register (RESULT) overwriting the result from the previous conversion.

To avoid data loss, if more than one channel is enabled, the conversion result must be read as soon as it is available (INTFLAG.RESRDY). Failing to do so will result in an overrun error condition, indicated by the OVERRUN bit in the Interrupt Flag Status and Clear register (INTFLAG.OVERRUN).

To enable one of the available interrupts sources, the corresponding bit in the Interrupt Enable Set register (INTENSET) must be written to '1'.

Related Links

45.6.2.1 Initialization

45.6.2.4 Prescaler Selection

The ADC is clocked by GCLK_ADCx. There is also a prescaler in the ADC to enable conversion at lower clock rates. Refer to CTRLA for details on prescaler settings. Refer to 45.6.2.8 Conversion Timing and Sampling Rate for details on timing and sampling rate.

47.8.8 Synchronization Busy

Name: SYNCBUSY
Offset: 0x08
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
			DATABUF1	DATABUF0	DATA1	DATA0	ENABLE	SWRST
Access			R	R	R	R	R	R
Reset			0	0	0	0	0	0

Bit 5 – DATABUF1 Data Buffer DAC1
This bit is set when DATABUF1 register is written.

This bit is cleared when DATABUF1 synchronization is completed.

Value	Description
0	No ongoing synchronized access.
1	Synchronized access is ongoing.

Bit 4 – DATABUF0 Data Buffer DAC0
This bit is set when DATABUF0 register is written.

This bit is cleared when DATABUF0 synchronization is completed.

Value	Description
0	No ongoing synchronized access.
1	Synchronized access is ongoing.

Bit 3 – DATA1 Data DAC1
This bit is set when DATA1 register is written.

This bit is cleared when DATA1 synchronization is completed.

48.7.2.2 Control B Clear

Name: CTRLBCLR
Offset: 0x04
Reset: 0x00
Property: PAC Write-Protection, Read-Synchronized, Write-Synchronized

This register allows the user to clear bits in the CTRLB register without doing a read-modify-write operation. Changes in this register will also be reflected in the Control B Set register (CTRLBSET).

Bit	7	6	5	4	3	2	1	0
	CMD[2:0]					ONESHOT	LUPD	DIR
Access	R/W	R/W	R/W			R/W	R/W	R/W
Reset	0	0	0			0	0	0

Bits 7:5 – CMD[2:0] Command
These bits are used for software control of the TC. The commands are executed on the next prescaled GCLK_TC clock cycle. When a command has been executed, the CMD bit group will be read back as zero.

Writing 0x0 to these bits has no effect.

Writing a '1' to any of these bits will clear the pending command.

Bit 2 – ONESHOT One-Shot on Counter
This bit controls one-shot operation of the TC.

Writing a '0' to this bit has no effect

Writing a '1' to this bit will disable one-shot operation.

Value	Description
0	The TC will wrap around and continue counting on an overflow/underflow condition.
1	The TC will wrap around and stop on the next underflow/overflow condition.

Bit 1 – LUPD Lock Update
This bit controls the update operation of the TC buffered registers.

When CTRLB.LUPD is set, no any update of the registers with value of its buffered register is performed on hardware UPDATE condition. Locking the update ensures that all buffer registers are valid before an hardware update is performed. After all the buffer registers are loaded correctly, the buffered registers can be unlocked.

This bit has no effect when input capture operation is enabled.

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

Writing a '1' to this bit will clear the LUPD bit.

Value	Description
0	The CCBUFx and PERBUF buffer registers value are copied into CCx and PER registers on hardware update condition.
1	The CCBUFx and PERBUF buffer registers value are not copied into CCx and PER registers on hardware update condition.

48.7.2.5 Interrupt Enable Clear

Name: INTENCLR
Offset: 0x08
Reset: 0x00
Property: PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

Bit	7	6	5	4	3	2	1	0
			MC1	MC0			ERR	OVF
Access			R/W	R/W			R/W	R/W
Reset			0	0			0	0

Bit 5 – MC1 Match or Capture Channel x Interrupt Enable
Writing a '0' to these bits has no effect.

Writing a '1' to MCx will clear the corresponding Match or Capture Channel x Interrupt Enable bit, which disables the Match or Capture Channel x interrupt.

Value	Description
0	The Match or Capture Channel x interrupt is disabled.
1	The Match or Capture Channel x interrupt is enabled.

Bit 4 – MC0 Match or Capture Channel x Interrupt Enable
Writing a '0' to these bits has no effect.

Writing a '1' to MCx will clear the corresponding Match or Capture Channel x Interrupt Enable bit, which disables the Match or Capture Channel x interrupt.

Value	Description
0	The Match or Capture Channel x interrupt is disabled.
1	The Match or Capture Channel x interrupt is enabled.

Bit 1 – ERR Error Interrupt Disable
Writing a '0' to this bit has no effect.

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

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

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

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

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

CTRLA.RESOLUTION	Bits [n:0]
0x0 - NONE	-
0x1 - DITH4	3:0
0x2 - DITH5	4:0
0x3 - DITH6	5:0 (depicted)