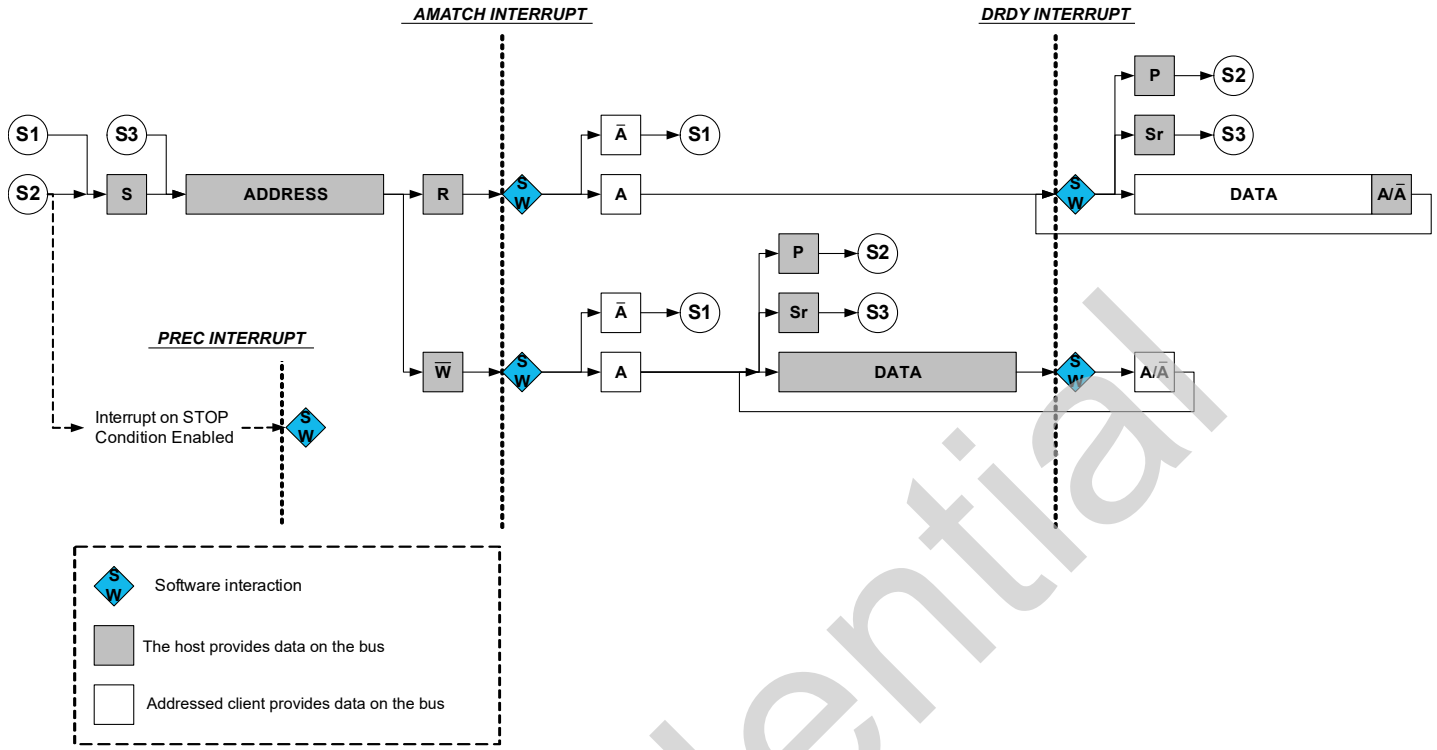
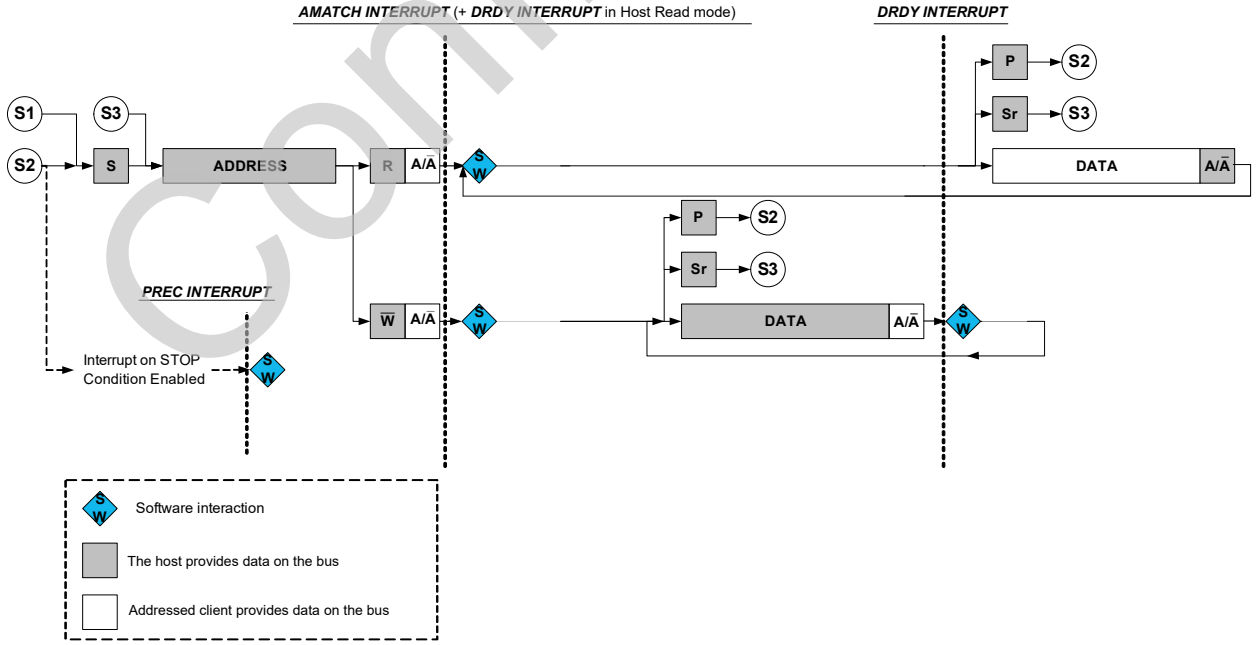
Figure 30-9. I²C Client Behavioral Diagram (SCLSM=0)



In the second strategy (CTRLA.SCLSM=1), interrupts only occur after the ACK bit is sent as shown in the following figure *I²C Client Behavioral Diagram (SCLSM=1)*. This strategy can be used when it is not necessary to check DATA before acknowledging. For host reads, an address and data interrupt will be issued simultaneously after the address acknowledge. However, for host writes, the first data interrupt will be seen after the first data byte has been received by the client and the acknowledge bit has been sent to the host.

Figure 30-10. I²C Client Behavioral Diagram (SCLSM=1)



30.6.2.5.1 Receiving Address Packets (SCLSM=0)

When CTRLA.SCLSM=0, the I²C client stretches the SCL line according to Figure 30-9. When the I²C client is properly configured, it will wait for a Start condition.

When a Start condition is detected, the successive address packet will be received and checked by the address match logic. If the received address is not a match, the packet will be rejected, and the I²C client will wait for a new Start condition. If the received address is a match, the Address Match bit in the Interrupt Flag register (INTFLAG.AMATCH) will be set.

SCL will be stretched until the I²C client clears INTFLAG.AMATCH. As the I²C client holds the clock by forcing SCL low, the software has unlimited time to respond.

The direction of a transaction is determined by reading the Read/Write Direction bit in the Status register (STATUS.DIR). This bit will be updated only when a valid address packet is received.

If the Transmit Collision bit in the Status register (STATUS.COLL) is set, this indicates that the last packet addressed to the I²C client had a packet collision. A collision causes the SDA and SCL lines to be released without any notification to software. Therefore, the next AMATCH interrupt is the first indication of the previous packet's collision. Collisions are intended to follow the SMBus Address Resolution Protocol (ARP).

After the address packet has been received from the I²C host, one of two cases will arise based on transfer direction.

Case 1: Address packet accepted – Read flag set

The STATUS.DIR bit is '1', indicating an I²C host read operation. The SCL line is forced low, stretching the bus clock. If an ACK is sent, I²C client hardware will set the Data Ready bit in the Interrupt Flag register (INTFLAG.DRDY), indicating data are needed for transmit. If a NACK is sent, the I²C client will wait for a new Start condition and address match.

Typically, software will immediately acknowledge the address packet by sending an ACK/NACK bit. The I²C client Command bit field in the Control B register (CTRLB.CMD) can be written to '0x3' for both read and write operations as the command execution is dependent on the STATUS.DIR bit. Writing '1' to INTFLAG.AMATCH will also cause an ACK/NACK to be sent corresponding to the CTRLB.ACKACT bit.

Case 2: Address packet accepted – Write flag set

The STATUS.DIR bit is cleared, indicating an I²C host write operation. The SCL line is forced low, stretching the bus clock. If an ACK is sent, the I²C client will wait for data to be received. Data, repeated start or stop can be received.

If a NACK is sent, the I²C client will wait for a new Start condition and address match. Typically, software will immediately acknowledge the address packet by sending an ACK/NACK. The I²C client command CTRLB.CMD = 3 can be used for both read and write operation as the command execution is dependent on STATUS.DIR.

Writing '1' to INTFLAG.AMATCH will also cause an ACK/NACK to be sent corresponding to the CTRLB.ACKACT bit.

30.6.2.5.2 Receiving Address Packets (SCLSM=1)

When SCLSM=1, the I²C client will stretch the SCL line only after an ACK (see Figure 30-10). When the I²C client is properly configured, it will wait for a Start condition to be detected.

When a Start condition is detected, the successive address packet will be received and checked by the address match logic.

If the received address is not a match, the packet will be rejected and the I²C client will wait for a new Start condition.

If the address matches, the acknowledge action as configured by the Acknowledge Action bit Control B register (CTRLB.ACKACT) will be sent and the Address Match bit in the Interrupt Flag register (INTFLAG.AMATCH) is set. SCL will be stretched until the I²C client clears INTFLAG.AMATCH. As the I²C client holds the clock by forcing SCL low, the software is given unlimited time to respond to the address.

The direction of a transaction is determined by reading the Read/Write Direction bit in the Status register (STATUS.DIR). This bit will be updated only when a valid address packet is received.

If the Transmit Collision bit in the Status register (STATUS.COLL) is set, the last packet addressed to the I²C client had a packet collision. A collision causes the SDA and SCL lines to be released without any notification to software. The next AMATCH interrupt is, therefore, the first indication of the previous packet's collision. Collisions are intended to follow the SMBus Address Resolution Protocol (*ARP*).

After the address packet has been received from the I²C host, INTFLAG.AMATCH can be set to '1' to clear it.

30.6.2.5.3 Receiving and Transmitting Data Packets

After the I²C client has received an address packet, it will respond according to the direction either by waiting for the data packet to be received or by starting to send a data packet by writing to DATA.DATA. When a data packet is received or sent, INTFLAG.DRDY will be set. After receiving data, the I²C client will send an acknowledge according to CTRLB.ACKACT.

Case 1: Data received

INTFLAG.DRDY is set, and SCL is held low, pending for SW interaction.

Case 2: Data sent

When a byte transmission is successfully completed, the INTFLAG.DRDY Interrupt flag is set. If NACK is received, indicated by STATUS.RXNACK=1, the I²C client must expect a stop or a repeated start to be received. The I²C client must release the data line to allow the I²C host to generate a stop or repeated start. Upon detecting a Stop condition, the Stop Received bit in the Interrupt Flag register (INTFLAG.PREC) will be set and the I²C client will return to IDLE state.

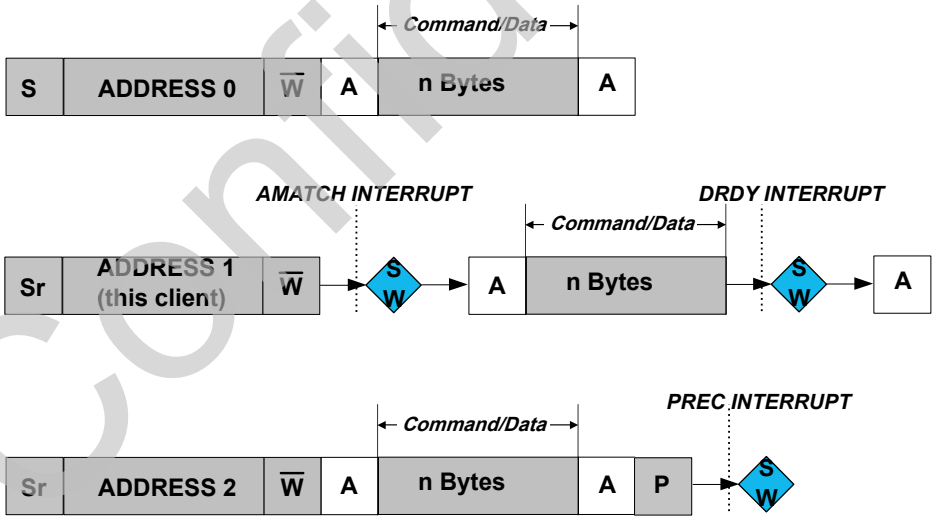
30.6.2.5.4 PMBus Group Command

When the PMBus Group Command bit in the CTRLB register is set (CTRLB.GCMD=1) and 7-bit addressing is used, INTFLAG.PREC will be set if the client has been addressed since the last STOP condition. When CTRLB.GCMD=0, a STOP condition without address match will not set INTFLAG.PREC.

The group command protocol is used to send commands to more than one device. The commands are sent in one continuous transmission with a single STOP condition at the end. When the STOP condition is detected by the clients addressed during the group command, they all begin executing the command they received.

The following figure shows an example where this client, bearing ADDRESS 1, is addressed after a repeated START condition. There can be multiple clients addressed before and after this client. Eventually, at the end of the group command, a single STOP is generated by the host. At this point a STOP interrupt is asserted.

Figure 30-11. PMBus Group Command Example



30.6.3 Additional Features

30.6.3.1 SMBus

The I²C includes three hardware SCL low time-outs which allow a time-out to occur for SMBus SCL low time-out, host extend time-out, and client extend time-out. This allows for SMBus functionality. These time-outs are driven by the GCLK_SERCOM_SLOW clock. The GCLK_SERCOM_SLOW clock is used to accurately time the time-out and must be configured to use a 32KHz_LPCLK. The I²C interface also allows for a SMBus compatible SDA hold time.

- $T_{TIMEOUT}$: SCL low time of 25..35ms – Measured for a single SCL low period. It is enabled by CTRLA.LOWTOUTEN.

- $T_{LOW:SEXT}$: Cumulative clock low extend time of 25 ms – Measured as the cumulative SCL low extend time by a client device in a single message from the initial START to the STOP. It is enabled by CTRLA.SEXTTOEN.
- $T_{LOW:MEXT}$: Cumulative clock low extend time of 10 ms – Measured as the cumulative SCL low extend time by the host device within a single byte from START-to-ACK, ACK-to-ACK, or ACK-to-STOP. It is enabled by CTRLA.MEXTTOEN.

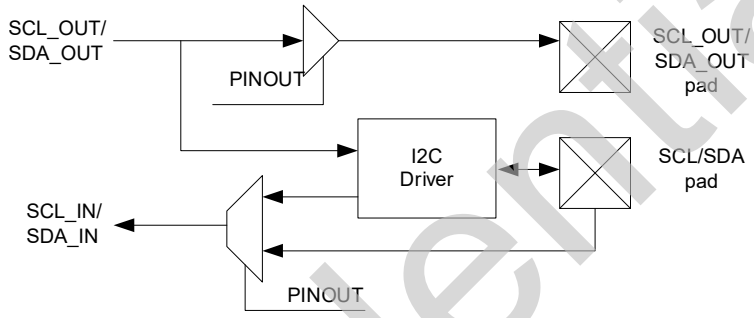
30.6.3.2 Smart Mode

The I^2C interface has a Smart mode that simplifies application code and minimizes the user interaction needed to adhere to the I^2C protocol. The Smart mode accomplishes this by automatically issuing an ACK or NACK (based on the content of CTRLB.ACKACT) as soon as DATA.DATA is read.

30.6.3.3 4-Wire Mode

Writing a '1' to the Pin Usage bit in the Control A register (CTRLA.PINOUT) will enable 4-Wire mode operation. In this mode, the internal I^2C tri-state drivers are bypassed, and an external I^2C compliant tri-state driver is needed when connecting to an I^2C bus.

Figure 30-12. I^2C Pad Interface



30.6.3.4 Quick Command

Setting the Quick Command Enable bit in the Control B register (CTRLB.QCEN) enables quick command. When quick command is enabled, the corresponding Interrupt flag (INTFLAG.SB or INTFLAG.MB) is set immediately after the client acknowledges the address. At this point, the software can either issue a Stop command or a repeated start by writing CTRLB.CMD or ADDR.ADDR.

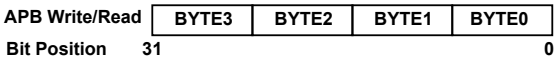
30.6.3.5 32-bit Extension

For better system bus utilization, 32-bit data receive and transmit can be enabled by writing to the Data 32-bit bit field in the Control C register (CTRLC.DATA32B = 1). When enabled, write and read transaction to/from the DATA register are 32 bit in size.

If frames are not multiples of 4 Bytes, the Length Counter (LENGTH.LEN) and Length Enable (LENGTH.LENEN) must be configured before data transfer begins. LENGTH.LEN must be enabled only when CTRLC.DATA32B is enabled.

The following figure shows the order of transmit and receive when using 32-bit mode. Bytes are transmitted or received and stored in order from 0 to 3.

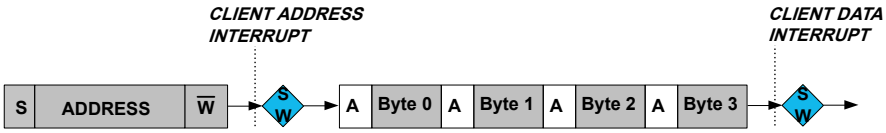
Figure 30-13. 32-bit Extension Byte Ordering



32-bit Extension Client Operation

The following figure shows a transaction with 32-bit Extension enabled (CTRLC.DATA32B=1). In client operation, the Address Match interrupt in the Interrupt Flag Status and Clear register (INTFLAG.AMATCH) is set after the address is received and available in the DATA register. The Data Ready interrupt (INTFLAG.DRDY) will then be raised for every 4 Bytes transferred.

Figure 30-14. 32-bit Extension Client Operation



The LENGTH register can be written before the frame begins, or when the AMATCH interrupt is set. If the frame size is not LENGTH.LEN Bytes, the Length Error status bit (STATUS.LENERR) is raised. If LENGTH.LEN is not a multiple of 4 Bytes, the final INTFLAG.DRDY interrupt is raised when the last Byte is received for host reads. For host writes, the last data byte will be automatically NACKed. On address recognition, the internal length counter is reset in preparation for the incoming frame.

When SCL clock stretch mode is selected (CTRLA.SCLSM=1) and the transaction is a host write, the selected Acknowledge Action (CTRLB.ACKACT) will only be used to ACK/NACK each 4th byte. All other bytes are ACKed. This allows the user to write CTRLB.ACKACT=1 in the final interrupt, so that the last byte in a 32-bit word will be NACKed.

Writing to the LENGTH register while a frame is in progress will produce unpredictable results. If LENGTH.LENEN is not set and a frame is not a multiple of 4 Bytes, the remainder will be lost.

32-bit Extension Host Operation

When using the I^2C configured as Host, the Address register must be written with the desired address (ADDR.ADDR), and optionally, the transaction Length and transaction Length Enable bits (ADDR.LEN and ADDR.LENEN) can be written. When ADDR.LENEN is written to '1' along with ADDR.ADDR, ADDR.LEN determines the number of data bytes in the transaction from 0 to 255. Then, the ADDR.LEN bytes are transferred, followed by an automatically generated NACK (for host reads) and a STOP.

The INTFLAG.SB or INTFLAG.MB are raised for every 4 Bytes transferred. If the transaction is a host read and ADDR.LEN is not a multiple of 4 Bytes, the final INTFLAG.SB is set when the last byte is received.

When SCL clock stretch mode is enabled (CTRLA.SCLSM=1) and the transaction is a host read, the selected Acknowledge Action (CTRLB.ACKACT) will only be used to ACK/NACK each 4th Byte. All other bytes are ACKed. This allows the user to set CTRLB.ACKACT=1 in the final interrupt, so that the last byte in a 32-bit word will be NACKed.

If a NACK is received by the client for a host write transaction before ADDR.LEN bytes, a STOP will be automatically generated, and the length error (STATUS.LENERR) is raised along with the INTFLAG.ERROR interrupt.

30.6.4 DMA, Interrupts and Events

Each interrupt source has its own interrupt flag. The interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG) will be set when the interrupt condition is meet. Each interrupt can be individually enabled by writing '1' to the corresponding bit in the Interrupt Enable Set register (INTENSET), and disabled by writing '1' to the corresponding bit in the Interrupt Enable Clear register (INTENCLR). An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request is active until the interrupt flag is cleared, the interrupt is disabled or the I^2C is reset. See the *INTFLAG* (Client) or *INTFLAG* (Host) register from Related Links for details on how to clear interrupt flags.

Table 30-1. Module Request for SERCOM I²C Client

Condition	Request		
	DMA	Interrupt	Event
Data needed for transmit (TX) (Client transmit mode)	Yes (request cleared when data is written)		NA
Data received (RX) (Client receive mode)	Yes (request cleared when data is read)		
Data Ready (DRDY)		Yes	
Address Match (AMATCH)		Yes	
Stop received (PREC)		Yes	
Error (ERROR)		Yes	

Table 30-2. Module Request for SERCOM I²C Host

Condition	Request		
	DMA	Interrupt	Event
Data needed for transmit (TX) (Host transmit mode)	Yes (request cleared when data is written)		NA
Data needed for transmit (RX) (Host transmit mode)	Yes (request cleared when data is read)		
Host on Bus (MB)		Yes	
Stop received (SB)		Yes	
Error (ERROR)		Yes	

Related Links

30.8.6. INTFLAG

30.10.7. INTFLAG

30.6.4.1 DMA Operation

Smart mode must be enabled for DMA operation in the Control B register by writing CTRLB.SMEN=1.

30.6.4.1.1 Client DMA

When using the I²C client with DMA, an address match will cause the address Interrupt flag (INTFLAG.ADDRMATCH) to be raised. After the interrupt has been serviced, data transfer will be performed through DMA.

The I²C client generates the following requests:

- Write data received (RX): The request is set when host write data is received. The request is cleared when DATA is read.
- Read data needed for transmit (TX): The request is set when data is needed for a host read operation. The request is cleared when DATA is written.

- Write data received (RX): If the FIFO is disabled, the request is set when host write data is received. If the FIFO is enabled, the request is set when the RX FIFO threshold is reached (CTRLC.RXTRHOLD). The request is cleared when DATA is read.
- Read data needed for transmit (TX): If the FIFO is disabled, the request is set when data is needed for a host read operation. If the FIFO is enabled, the request is set when the TX FIFO threshold is reached (CTRLC.TXTRHOLD). The request is cleared when DATA is written.

30.6.4.1.2 Host DMA

When using the I²C host with DMA, the ADDR register must be written with the desired address (ADDR.ADDR), transaction length (ADDR.LEN), and transaction length enable (ADDR.LENEN). When ADDR.LENEN is written to 1 along with ADDR.ADDR, ADDR.LEN determines the number of data bytes in the transaction from 0 to 255. DMA is then used to transfer ADDR.LEN bytes followed by an automatically generated NACK (for host reads) and a STOP.

If a NACK is received by the client for a host write transaction before ADDR.LEN bytes, a STOP will be automatically generated and the length error (STATUS.LENERR) will be raised along with the INTFLAG.ERROR interrupt.

The I²C host generates the following requests:

- Read data received (RX): The request is set when host read data is received. The request is cleared when DATA is read.
- Write data needed for transmit (TX): The request is set when data is needed for a host write operation. The request is cleared when DATA is written.
- Read data received (RX): If the FIFO is disabled, the request is set when host read data is received. If the FIFO is enabled, the request is set when the RX FIFO threshold is reached. The request is cleared when DATA is read.
- Write data needed for transmit (TX): If the FIFO is disabled, the request is set when data is needed for a host write operation. If the FIFO is enabled, the request is set when the TX FIFO threshold is reached (CTRLC.TXTRHOLD). The request is cleared when DATA is written.

30.6.4.2 Interrupts

The I²C client has the following interrupt sources. These are asynchronous interrupts. They can wake-up the device from any Sleep mode:

- Error (ERROR)
- Data Ready (DRDY)
- Address Match (AMATCH)
- Stop Received (PREC)

The I²C host has the following interrupt sources. These are asynchronous interrupts. They can wake-up the device from any Sleep mode:

- Error (ERROR)
- Client on Bus (SB)
- Host on Bus (MB)

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

The status of enabled interrupts can be read from either INTENSET or INTENCLR. An interrupt request is generated when the Interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the Interrupt flag is cleared, the interrupt is disabled or the I²C is reset. For details on how to clear Interrupt flags, see *INTFLAG* register from Related Links.

The value of INTFLAG indicates which interrupt is executed. Note that interrupts must be globally enabled for interrupt requests. See *Nested Vector Interrupt Controller (NVIC)* from Related Links.

Related Links

8.2. Nested Vector Interrupt Controller (NVIC)
30.8.6. INTFLAG
30.10.7. INTFLAG

30.6.4.3 Events

Not applicable.

30.6.4.4 Sleep Mode Operation

I²C Host Operation

The generic clock (GCLK_SERCOMx_CORE) will continue to run in idle sleep mode. If the Run In Standby bit in the Control A register (CTRLA.RUNSTDBY) is '1', the GLK_SERCOMx_CORE will also run in Standby Sleep mode. Any interrupt can wake-up the device.

If CTRLA.RUNSTDBY=0, the GLK_SERCOMx_CORE will be disabled after any ongoing transaction is finished. Any interrupt can wake-up the device.

I²C Client Operation

Writing CTRLA.RUNSTDBY=1 will allow the Address Match interrupt to wake-up the device.

When CTRLA.RUNSTDBY=0, all receptions will be dropped.

30.7 Register Summary

See *SERCOM0/SERCOM1/SERCOM2* module in the *Product Memory Mapping Overview* from Related Links for base address based on the SERCOM instant used.

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x00	CTRLA	7:0	RUNSTDBY			MODE[2:0]			ENABLE	SWRST
		15:8								
		23:16	SEXTTOEN							PINOUT
		31:24		LOWTOUT			SCLSM		SPEED[1:0]	
0x04	CTRLB	7:0								
		15:8	AMODE[1:0]					AACKEN	GCMD	SMEN
		23:16	FIFOCLR[1:0]					ACKACT	CMD[1:0]	
		31:24								
0x08	CTRLC	7:0					SDASETUP[3:0]			
		15:8								
		23:16								
		31:24								DATA32B
0x0C ... 0x13	Reserved									
0x14	INTENCLR	7:0	ERROR			RXFF	TXFE	DRDY	AMATCH	PREC
0x15	Reserved									
0x16	INTENSET	7:0	ERROR			RXFF	TXFE	DRDY	AMATCH	PREC
0x17	Reserved									
0x18	INTFLAG	7:0	ERROR			RXFF	TXFE	DRDY	AMATCH	PREC
0x19	Reserved									
0x1A	STATUS	7:0	CLKHOLD	LOWTOUT		SR	DIR	RXNACK	COLL	BUSERR
		15:8					LENERR		SEXTTOUT	
0x1C	SYNCBUSY	7:0				LENGTH		SYSOP	ENABLE	SWRST
		15:8								
		23:16								
		31:24								
0x20 ... 0x21	Reserved									
0x22	LENGTH	7:0	LEN[7:0]							
		15:8								LENEN
0x24	ADDR	7:0	ADDR[6:0]							GENCEN
		15:8						ADDR[9:7]		
		23:16	ADDRMASK[6:0]							
		31:24						ADDRMASK[9:7]		
0x28	DATA	7:0	DATA[7:0]							
		15:8	DATA[15:8]							
		23:16	DATA[23:16]							
		31:24	DATA[31:24]							
0x2C ... 0x33	Reserved									
0x34	FIFOSPACE	7:0				TXSPACE[4:0]				
		15:8				RXSPACE[4:0]				
0x36	FIFOPTR	7:0					CPUWRPTR[3:0]			
		15:8					CPURDPTR[3:0]			

Related Links

7. Product Memory Mapping Overview

30.8 Register Description - I²C Client

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.

Some registers are synchronized when read and/or written. Synchronization is denoted by the “Write-Synchronized” or the “Read-Synchronized” property in each individual register description.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable-protection is denoted by the “Enable-Protected” property in each individual register description.

30.8.1 Control A

Name: CTRLA
Offset: 0x00
Reset: 0x00000000
Property: PAC Write-Protection, Enable-Protected, Write-Synchronized

Bit	31	30	29	28	27	26	25	24
		LOWTOUT			SCLSM		SPEED[1:0]	
Access		R/W			R/W		R/W	R/W
Reset		0			0		0	0

Bit	23	22	21	20	19	18	17	16
	SEXTTOEN							PINOUT
Access	R/W							R/W
Reset	0							0

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

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

Bit 30 – LOWTOUT SCL Low Time-Out
This bit enables the SCL low time-out. If SCL is held low for 25 ms-35 ms, the client will release its clock hold, if enabled, and reset the internal state machine. Any interrupt flags set at the time of time-out will remain set.
This bit is not synchronized.

Value	Description
0	Time-out disabled.
1	Time-out enabled.

Bit 27 – SCLSM SCL Clock Stretch Mode
This bit controls when SCL will be stretched for software interaction.
This bit is not synchronized.

Value	Description
0	SCL stretch according to Figure 30-9
1	SCL stretch only after ACK bit according to Figure 30-10

Bits 25:24 – SPEED[1:0] Transfer Speed
These bits define bus speed.
These bits are not synchronized.

Value	Description
0x0	Standard-mode (Sm) up to 100 kHz and Fast-mode (Fm) up to 400 kHz
0x1	Fast-mode Plus (Fm+) up to 1 MHz
0x2	Reserved
0x3	Reserved

Bit 23 – SEXTTOEN Client SCL Low Extend Time-Out
This bit enables the client SCL low extend time-out. If SCL is cumulatively held low for greater than 25 ms from the initial START to a STOP, the client will release its clock hold if enabled and reset the internal state machine. Any interrupt flags set at the time of time-out will remain set. If the address was recognized, PREC will be set when a STOP is received.

This bit is not synchronized.

Value	Description
0	Time-out disabled
1	Time-out enabled

Bit 16 – PINOUT Pin Usage

This bit sets the pin usage to either two- or four-wire operation:
This bit is not synchronized.

Value	Description
0	4-wire operation disabled
1	4-wire operation enabled

Bit 7 – RUNSTDBY Run in Standby

This bit defines the functionality in standby sleep mode.
This bit is not synchronized.

Value	Description
0	Disabled – All reception is dropped.
1	Wake on address match, if enabled.

Bits 4:2 – MODE[2:0] Operating Mode

These bits must be written to 0x04 to select the I^2C client serial communication interface of the SERCOM.
These bits are not synchronized.

Bit 1 – ENABLE Enable

Due to synchronization, there is delay from writing CTRLA.ENABLE until the peripheral is enabled/disabled.
The value written to CTRL.ENABLE will read back immediately and the Enable Synchronization Busy bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.
This bit is not enable-protected.

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

Bit 0 – SWRST Software Reset

Writing '0' to this bit has no effect.
Writing '1' to this bit resets all registers in the SERCOM, except DBGCTRL, to their initial state, and the SERCOM will be disabled.
Writing '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded. Any register write access during the ongoing reset will result in an APB error. Reading any register will return the reset value of the register.
Due to synchronization, there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the reset is complete.
This bit is not enable-protected.
Note: During a SWRST, access to registers/bits without SWRST are disallowed until SYNCBUSY.SWRST cleared by hardware.

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

30.8.2 Control B

Name: CTRLB
Offset: 0x04
Reset: 0x00000000
Property: PAC Write-Protection, Enable-Protected

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	FIFOCLR[1:0]					ACKACT	CMD[1:0]	
Access	R/W	R/W				R/W	W	W
Reset	0	0				0	0	0

Bit	15	14	13	12	11	10	9	8
	AMODE[1:0]					AACKEN	GCMD	SMEN
Access	R/W	R/W				R/W	R/W	R/W
Reset	0	0				0	0	0

Bit	7	6	5	4	3	2	1	0
Access								
Reset								

Bits 23:22 – FIFOCLR[1:0] FIFO Clear
When these bits are set, the corresponding FIFO will be cleared. The bits will automatically clear when SYNCBUSY.SYSOP = 0.
These bits are not enable-protected.

FIFOCLR[1:0]	Name	Description
0x0	NONE	No action
0x1	TXFIFO	Clear TX FIFO
0x2	RXFIFO	Clear RX FIFO
0x3	BOTH	Clear both TX/RX FIFO

Bit 18 – ACKACT Acknowledge Action
This bit defines the client's acknowledge behavior after an address or data byte is received from the host.
The acknowledge action is executed when a command is written to the CMD bits. If smart mode is enabled (CTRLB.SMEN=1), the acknowledge action is performed when the DATA register is read.
ACKACT shall not be updated more than once between each peripheral interrupts request.
This bit is not enable-protected.

Value	Description
0	Send ACK
1	Send NACK

Bits 17:16 – CMD[1:0] Command
This bit field triggers the client operation as the below. The CMD bits are strobe bits, and always read as zero.
The operation is dependent on the client interrupt flags, INTFLAG.DRDY and INTFLAG.AMATCH, in addition to STATUS.DIR.
All interrupt flags (INTFLAG.DRDY, INTFLAG.AMATCH and INTFLAG.PREC) are automatically cleared when a command is given.
This bit is not enable-protected.

Table 30-3. Command Description

CMD[1:0]	DIR	Action
0x0	X	(No action)
0x1	X	(Reserved)
0x2	Used to complete a transaction in response to a data interrupt (DRDY)	
	0 (Host write)	Execute acknowledge action succeeded by waiting for any start (S/Sr) condition
	1 (Host read)	Wait for any start (S/Sr) condition
0x3	Used in response to an address interrupt (AMATCH)	
	0 (Host write)	Execute acknowledge action succeeded by reception of next byte
	1 (Host read)	Execute acknowledge action succeeded by client data interrupt
	Used in response to a data interrupt (DRDY)	
	0 (Host write)	Execute acknowledge action succeeded by reception of next byte
	1 (Host read)	Execute a byte read operation followed by ACK/NACK reception

Bits 15:14 – AMODE[1:0] Address Mode

These bits set the addressing mode.

Value	Name	Description
0x0	MASK	The client responds to the address written in ADDR.ADDR masked by the value in ADDR.ADDRMASK.
0x1	2_ADDRS	The client responds to the two unique addresses in ADDR.ADDR and ADDR.ADDRMASK.
0x2	RANGE	The client responds to the range of addresses between and including ADDR.ADDR and ADDR.ADDRMASK. ADDR.ADDR is the upper limit.
0x3	—	Reserved.

Bit 10 – AACKEN Automatic Acknowledge Enable

This bit enables the address to be automatically acknowledged if there is an address match.

Value	Description
0	Automatic acknowledge is disabled.
1	Automatic acknowledge is enabled.

Bit 9 – GCMD PMBus Group Command

This bit enables PMBus group command support. When enabled, the Stop Received interrupt flag (INTFLAG.PREC) will be set when a STOP condition is detected if the client has been addressed since the last STOP condition on the bus.

Value	Description
0	Group command is disabled.
1	Group command is enabled.

Bit 8 – SMEN Smart Mode Enable

When smart mode is enabled, data is acknowledged automatically when DATA.DATA is read.

Value	Description
0	Smart mode is disabled.
1	Smart mode is enabled.

30.8.3 Control C

Name: CTRLC
Offset: 0x08
Reset: 0x00000000
Property: PAC Write-Protection, Enable-Protected

Bit	31	30	29	28	27	26	25	24
								DATA32B
Access								R/W
Reset								0

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
					SDASETUP[3:0]			
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit 24 – DATA32B Data 32 Bit

This bit enables 32-bit data writes and reads to/from the DATA register.

Value	Description
0	Data transaction to/from DATA are 8-bit in size
1	Data transaction to/from DATA are 32-bit in size

Bits 3:0 – SDASETUP[3:0] SDA Setup Time

These bits select the minimum SDA-to-SCL setup time, measured from the release of SDA to the release of SCL:

$t_{\mathrm{SU:DAT}} = (GCLK\_SERCOMx \times \mathrm{APB\ period}\ (PBx\_CLK)) \times (6 + 16 \times \mathrm{SDASETUP})$

30.8.4 Interrupt Enable Clear

Name: INTENCLR
Offset: 0x14
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
	ERROR			RXFF	TXFE	DRDY	AMATCH	PREC
Access	R/W			R/W	R/W	R/W	R/W	R/W
Reset	0			0	0	0	0	0

Bit 7 – ERROR Error Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Error Interrupt Enable bit, which disables the Error interrupt.

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

Bit 4 – RXFF RX FIFO Full Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the RX FIFO Full bit, which disables the RX FIFO Full interrupt.

Value	Description
0	The RX FIFO Full interrupt is disabled.
1	The RX FIFO Full interrupt is enabled.

Bit 3 – TXFE TX FIFO Empty Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the TX FIFO Empty bit, which disables the TX FIFO Empty interrupt.

Value	Description
0	The TX FIFO Empty interrupt is disabled.
1	The TX FIFO Empty interrupt is enabled.

Bit 2 – DRDY Data Ready Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Data Ready bit, which disables 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 clear the Address Match Interrupt Enable bit, which disables 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 clear the Stop Received Interrupt Enable bit, which disables the Stop Received interrupt.

Value	Description
0	The Stop Received interrupt is disabled.
1	The Stop Received interrupt is enabled.

30.8.5 Interrupt Enable Set

Name: INTENSET
Offset: 0x16
Reset: 0x00
Property: PAC Write-Protection

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

Bit	7	6	5	4	3	2	1	0
	ERROR			RXFF	TXFE	DRDY	AMATCH	PREC
Access	R/W			R/W	R/W	R/W	R/W	R/W
Reset	0			0	0	0	0	0

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 4 – RXFF RX FIFO Full Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the RX FIFO Full bit, which enables the RX FIFO Full interrupt.

Value	Description
0	The RX FIFO Full interrupt is disabled.
1	The RX FIFO Full interrupt is enabled.

Bit 3 – TXFE TX FIFO Empty Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the TX FIFO Empty bit, which enables the TX FIFO Empty interrupt.

Value	Description
0	The TX FIFO Empty interrupt is disabled.
1	The TX FIFO Empty 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 Interrupt Enable 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.

30.8.6 Interrupt Flag Status and Clear

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

Bit	7	6	5	4	3	2	1	0
	ERROR			RXFF	TXFE	DRDY	AMATCH	PREC
Access	R/W			R/W	R/W	R/W	R/W	R/W
Reset	0			0	0	0	0	0

Bit 7 – ERROR Error
This bit is set when any error is detected. Errors that will set this flag have corresponding status flags in the STATUS register. The corresponding bits in STATUS are LENERR, SEXTTOUT, LOWTOUT, COLL and BUSERR.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the flag.

Bit 4 – RXFF RX FIFO Full
This flag is set when RX FIFO Threshold locations are fulfilled.
The flag is cleared when the RX FIFO is empty.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the RX FIFO Full interrupt flag.

Bit 3 – TXFE TX FIFO Empty
This flag is set when TX FIFO Threshold locations are available.
The flag is cleared when the TX FIFO is full.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the TX FIFO Empty interrupt flag.

Bit 2 – DRDY Data Ready
This flag is set when a I^2C client byte transmission is successfully completed.
The flag is cleared by hardware when either:
- Writing to the DATA register.
- Reading the DATA register with Smart mode enabled.
- Writing a valid command to the CMD register.

Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Data Ready Interrupt flag.

Bit 1 – AMATCH Address Match
This flag is set when the I^2C client address match logic detects that a valid address has been received.
The flag is cleared by hardware when CTRL.CMD is written.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Address Match Interrupt flag. When cleared, an ACK/NACK will be sent according to CTRLB.ACKACT.

Bit 0 – PREC Stop Received
This flag is set when a Stop condition is detected for a transaction being processed. A Stop condition detected between a bus host and another client will not set this flag, unless the PMBus Group Command is enabled in the Control B register (CTRLB.GCMD=1).
This flag is cleared by hardware after a command is issued on the next address match.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Stop Received Interrupt flag.

30.8.7 Status

Name: STATUS
Offset: 0x1A
Reset: 0x0000
Property: -

Bit	15	14	13	12	11	10	9	8
					LENERR		SEXTTOUT	
Access					R/W		R/W	
Reset					0		0	

Bit	7	6	5	4	3	2	1	0
	CLKHOLD	LOWTOUT		SR	DIR	RXNACK	COLL	BUSERR
Access	R	R/W		R	R	R	R/W	R/W
Reset	0	0		0	0	0	0	0

Bit 11 – LENERR Transaction Length Error
This bit is set when the length counter is enabled (LENGTH.LENEN) and a STOP or repeated START is received before or after the length in LENGTH.LEN is reached.
This bit is cleared automatically when responding to a new start condition with ACK or NACK (CTRLB.CMD=0x3) or when INTFLAG.AMATCH is cleared.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the status.

Bit 11 – LENERR Transaction Length Error
This bit is set when the length counter is enabled (LENGTH.LENEN) and a STOP or repeated START is received before or after the length in LENGTH.LEN is reached.
This bit is cleared automatically if responding to a new start condition with ACK or NACK (write 3 to CTRLB.CMD) or when INTFLAG.AMATCH is cleared.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the status.

Value	Description
0	No length error has occurred.
1	Length error has occurred.

Bit 9 – SEXTTOUT Client SCL Low Extend Time-Out
This bit is set if a client SCL low extend time-out occurs.
This bit is cleared automatically if responding to a new start condition with ACK or NACK (write 3 to CTRLB.CMD) or when INTFLAG.AMATCH is cleared.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the status.

Value	Description
0	No SCL low extend time-out has occurred.
1	SCL low extend time-out has occurred.

Bit 7 – CLKHOLD Clock Hold
The client Clock Hold bit (STATUS.CLKHOLD) is set when the client is holding the SCL line low, stretching the I2C clock. Software must consider this bit a read-only status flag that is set when INTFLAG.DRDY or INTFLAG.AMATCH is set.
This bit is automatically cleared when the corresponding interrupt is also cleared.

Bit 6 – LOWTOUT SCL Low Time-out
This bit is set if an SCL low time-out occurs.
This bit is cleared automatically if responding to a new start condition with ACK or NACK (write 3 to CTRLB.CMD) or when INTFLAG.AMATCH is cleared.

Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the status.

Value	Description
0	No SCL low time-out has occurred.
1	SCL low time-out has occurred.

Bit 4 – SR Repeated Start

When INTFLAG.AMATCH is raised due to an address match, SR indicates a repeated start or start condition.
This flag is only valid while the INTFLAG.AMATCH flag is one.

Value	Description
0	Start condition on last address match
1	Repeated start condition on last address match

Bit 3 – DIR Read / Write Direction

The Read/Write Direction (STATUS.DIR) bit stores the direction of the last address packet received from a host .

Value	Description
0	Host write operation is in progress.
1	Host read operation is in progress.

Bit 2 – RXNACK Received Not Acknowledge

This bit indicates whether the last data packet sent was acknowledged or not.

Value	Description
0	Host responded with ACK.
1	Host responded with NACK.

Bit 1 – COLL Transmit Collision

If set, the I2C client was not able to transmit a high data or NACK bit, the I2C client will immediately release the SDA and SCL lines and wait for the next packet addressed to it.
This flag is intended for the SMBus address resolution protocol (ARP). A detected collision in non-ARP situations indicates that there has been a protocol violation, and must be treated as a bus error.
Note: This status will not trigger any interrupt, and must be checked by software to verify that the data were sent correctly. This bit is cleared automatically if responding to an address match with an ACK or a NACK (writing 0x3 to CTRLB.CMD), or INTFLAG.AMATCH is cleared.

Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the status.

Value	Description
0	No collision detected on last data byte sent.
1	Collision detected on last data byte sent.

Bit 0 – BUSERR Bus Error

The Bus Error bit (STATUS.BUSERR) indicates that an illegal bus condition has occurred on the bus, regardless of bus ownership. An illegal bus condition is detected if a protocol violating start, repeated start or stop is detected on the I2C bus lines. A start condition directly followed by a stop condition is one example of a protocol violation. If a time-out occurs during a frame, this is also considered a protocol violation, and will set STATUS.BUSERR.
This bit is cleared automatically if responding to an address match with an ACK or a NACK (writing 0x3 to CTRLB.CMD) or INTFLAG.AMATCH is cleared.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the status.

Value	Description
0	No bus error detected.
1	Bus error detected.

30.8.8 Synchronization Busy

Name: SYNCBUSY
Offset: 0x1C
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
				LENGTH		SYSOP	ENABLE	SWRST
Access				R		R	R	R
Reset				0		0	0	0

Bit 4 – LENGTH LENGTH Synchronization Busy
Writing LENGTH requires synchronization. When written, this bit will be set until synchronization is complete. If LENGTH is written while SYNCBUSY.LENGTH is asserted, an APB error will be generated.
Note: In client mode, the clock is only running during data transfer, so SYNCBUSY.LENGTH will remain asserted until the next data transfer begins.

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

Bit 2 – SYSOP System Operation Synchronization Busy
Writing CTRLB.FIFOCLR when the SERCOM is enabled requires synchronization. When written, the SYNCBUSY.SYSOP bit will be set until synchronization is complete.

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

Bit 1 – ENABLE SERCOM Enable Synchronization Busy
Enabling and disabling the SERCOM (CTRLA.ENABLE) requires synchronization. Ongoing synchronization is indicated by SYNCBUSY.ENABLE = 1 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. Ongoing synchronization is indicated by SYNCBUSY.SWRST = 1 until synchronization is complete.
Note: During a SWRST, access to registers/bits without SWRST are disallowed until SYNCBUSY.SWRST cleared by hardware.

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

30.8.9 Length

Name: LENGTH
Offset: 0x22
Reset: 0x0000
Property: PAC Write-Protection, Write-Synchronized

Bit	15	14	13	12	11	10	9	8
								LENEN
Access								R/W
Reset								0

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

Bit 8 – LENEN Data Length Enable
In 32-bit Extension mode (CTRLC.DATA32B=1), this bit field enables the length counter.

Value	Description
0	Length counter is disabled.
1	Length counter is enabled.

Bits 7:0 – LEN[7:0] Data Length
In 32-bit Extension mode (CTRLC.DATA32B=1) with Data Length counting enabled (LENGTH.LENEN), this bit field configures the data length from 0 to 255 Bytes after which the flag INTFLAG.DRDY is raised.

30.8.10 Address

Name: ADDR
Offset: 0x24
Reset: 0x00000000
Property: PAC Write-Protection, Enable-Protected

Bit	31	30	29	28	27	26	25	24
						ADDRMASK[9:7]		
Access						R/W	R/W	R/W
Reset						0	0	0

Bit	23	22	21	20	19	18	17	16
	ADDRMASK[6:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reset	0	0	0	0	0	0	0	

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

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

Bits 26:17 – ADDRMASK[9:0] Address Mask
These bits act as a second address match register, an address mask register or the lower limit of an address range, depending on the CTRLB.AMODE setting.

Bits 10:1 – ADDR[9:0] Address
These bits contain the I^2C client address used by the client address match logic to determine if a host has addressed the client.
When using 7-bit addressing, the client address is represented by ADDR[6:0].
When the address match logic detects a match, INTFLAG.AMATCH is set and STATUS.DIR is updated to indicate whether it is a read or a write transaction.

Bit 0 – GENCEN General Call Address Enable
A general call address is an address consisting of all-zeroes, including the direction bit (host write).

Value	Description
0	General call address recognition disabled.
1	General call address recognition enabled.

30.8.11 Data

Name: DATA
Offset: 0x28
Reset: 0x00000000
Property: Read/Write

Bit	31	30	29	28	27	26	25	24
	DATA[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
	DATA[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
	DATA[15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

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

Bits 31:0 – DATA[31:0] Data

The client data register I/O location (DATA.DATA) provides access to the host transmit and receive data buffers. Reading valid data or writing data to be transmitted can be successfully done only when SCL is held low by the client (STATUS.CLKHOLD is set). An exception occurs when reading the last data byte after the stop condition has been received.

Accessing DATA.DATA auto-triggers I^2C bus operations. The operation performed depends on the state of CTRLB.ACKACT, CTRLB.SMEN and the type of access (read/write).

When CTRLC.DATA32B=1, read and write transactions from/to the DATA register are 32 bit in size. Otherwise, reads and writes are 8 bit.

30.8.12 FIFO Space

Name: FIFOSPACE
Offset: 0x34
Reset: 0x0000
Property: -

This register allows the user to identify the number of bytes present in each TX and RX FIFO.

Bit	15	14	13	12	11	10	9	8
				RXSPACE[4:0]				
Access				R	R	R	R	R
Reset				0	0	0	0	0

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

Bits 12:8 – RXSPACE[4:0] RX FIFO Filled Space
These bits return the number filled locations in the RX FIFO (bytes or words, depending on CTRLC.DATA32B setting).

Bits 4:0 – TXSPACE[4:0] TX FIFO Empty Space
These bits return the number of available locations in the TX FIFO (bytes or words, depending on CTRLC.DATA32B setting).

30.8.13 FIFO CPU Pointers

Name: FIFOPTR
Offset: 0x36
Reset: 0x0000
Property: -

This register provides a copy of internal CPU TX and RX FIFO pointers.

Bit	15	14	13	12	11	10	9	8
					CPURDPTR[3:0]			
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

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

Bits 11:8 – CPURDPTR[3:0] RX FIFO Filled Space
These bits return the CPURDPTR pointer value. These bits can be written only if the SERCOM is halted during debugging. Reading DATA register, will return RXFIFO[CPURDPTR] location value.

Bits 3:0 – CPUWRPTR[3:0] TX FIFO Filled Space
These bits return the CPURDPTR pointer value. These bits can be written only if the SERCOM is halted during debugging. When writting to DATA register, the DATA will be written to TXFIFO[CPUWRPTR] location.

30.9 Register Summary

See *SERCOM0/SERCOM1/SERCOM2* module in the *Product Memory Mapping Overview* from Related Links for base address based on the SERCOM instant used.

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x00	CTRLA	7:0	RUNSTDBY			MODE[2:0]			ENABLE	SWRST
		15:8								
		23:16	SEXTTOEN	MEXTTOEN						PINOUT
		31:24		LOWTOUT	INACTOUT[1:0]		SCLSM		SPEED[1:0]	
0x04	CTRLB	7:0								
		15:8							QCEN	SMEN
		23:16	FIFOCLR[1:0]					ACKACT	CMD[1:0]	
		31:24								
0x08	CTRLC	7:0								
		15:8								
		23:16								
		31:24								DATA32B
0x0C	BAUD	7:0	BAUD[7:0]							
		15:8	BAUDLOW[7:0]							
		23:16								
		31:24								
0x10 ... 0x13	Reserved									
0x14	INTENCLR	7:0	ERROR			RXFF	TXFE		SB	MB
0x15	Reserved									
0x16	INTENSET	7:0	ERROR			RXFF	TXFE		SB	MB
0x17	Reserved									
0x18	INTFLAG	7:0	ERROR			RXFF	TXFE		SB	MB
0x19	Reserved									
0x1A	STATUS	7:0	CLKHOLD	LOWTOUT	BUSSTATE[1:0]			RXNACK	ARBLOST	BUSERR
		15:8						LENERR	SEXTTOUT	MEXTTOUT
0x1C	SYNCBUSY	7:0						SYSOP	ENABLE	SWRST
		15:8								
		23:16								
		31:24								
0x20 ... 0x23	Reserved									
0x24	ADDR	7:0	ADDR[7:0]							
		15:8	TENBITEN		LENEN			ADDR[10:8]		
		23:16	LEN[7:0]							
		31:24								
0x28	DATA	7:0	DATA[7:0]							
		15:8	DATA[15:8]							
		23:16	DATA[23:16]							
		31:24	DATA[31:24]							
0x2C ... 0x2F	Reserved									
0x30	DBGCTRL	7:0								DBGSTOP
0x31 ... 0x33	Reserved									
0x34	FIFOSPACE	7:0				TXSPACE[4:0]				
		15:8				RXSPACE[4:0]				
0x36	FIFOPTR	7:0					CPUWRPTR[3:0]			
		15:8					CPURDPTR[3:0]			

Related Links

7. Product Memory Mapping Overview

30.10 Register Description – I²C Host

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.

Some registers are synchronized when read and/or written. Synchronization is denoted by the “Write-Synchronized” or the “Read-Synchronized” property in each individual register description.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable-protection is denoted by the “Enable-Protected” property in each individual register description.

30.10.1 Control A

Name: CTRLA
Offset: 0x00
Reset: 0x00000000
Property: PAC Write-Protection, Enable-Protected, Write-Synchronized

Bit	31	30	29	28	27	26	25	24
		LOWTOUT	INACTOUT[1:0]		SCLSM		SPEED[1:0]	
Access		R/W	R/W	R/W	R/W		R/W	R/W
Reset		0	0	0	0		0	0

Bit	23	22	21	20	19	18	17	16
	SEXTTOEN	MEXTTOEN						PINOUT
Access	R/W	R/W						R/W
Reset	0	0						0

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

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

Bit 30 – LOWTOUT SCL Low Time-Out

This bit enables the SCL low time-out. If SCL is held low for 25ms-35ms, the host will release its clock hold, if enabled, and complete the current transaction. A stop condition will automatically be transmitted. INTFLAG.SB or INTFLAG.MB will be set as normal, but the clock hold will be released. The STATUS.LOWTOUT and STATUS.BUSERR status bits will be set.
This bit is not synchronized.

Value	Description
0	Time-out disabled.
1	Time-out enabled.

Bits 29:28 – INACTOUT[1:0] Inactive Time-Out

If the inactive bus time-out is enabled and the bus is inactive for longer than the time-out setting, the bus state logic will be set to idle. An inactive bus arise when either an I^2C host or client is holding the SCL low.
Enabling this option is necessary for SMBus compatibility, but can also be used in a non-SMBus set-up.
Calculated time-out periods are based on a 100kHz baud rate.
These bits are not synchronized.

Value	Name	Description
0x0	DIS	Disabled
0x1	55US	5-6 SCL cycle time-out (50-60µs)
0x2	105US	10-11 SCL cycle time-out (100-110µs)
0x3	205US	20-21 SCL cycle time-out (200-210µs)

Bit 27 – SCLSM SCL Clock Stretch Mode

This bit controls when SCL will be stretched for software interaction.
This bit is not synchronized.

Value	Description
0	SCL stretch according to Figure 30-5
1	SCL stretch only after ACK bit, Figure 30-6

Bits 25:24 – SPEED[1:0] Transfer Speed

These bits define bus speed.
These bits are not synchronized.

Value	Description
0x0	Standard-mode (Sm) up to 100 kHz and Fast-mode (Fm) up to 400 kHz
0x1	Fast-mode Plus (Fm+) up to 1 MHz
0x2	Reserved
0x3	Reserved

Bit 23 – SEXTTOEN Client SCL Low Extend Time-Out

This bit enables the client SCL low extend time-out. If SCL is cumulatively held low for greater than 25ms from the initial START to a STOP, the host will release its clock hold if enabled, and complete the current transaction. A STOP will automatically be transmitted.
SB or MB will be set as normal, but CLKHOLD will be release. The MEXTTOUT and BUSERR status bits will be set.
This bit is not synchronized.

Value	Description
0	Time-out disabled
1	Time-out enabled

Bit 22 – MEXTTOEN Host SCL Low Extend Time-Out

This bit enables the host SCL low extend time-out. If SCL is cumulatively held low for greater than 10ms from START-to-ACK, ACK-to-ACK, or ACK-to-STOP the host will release its clock hold if enabled, and complete the current transaction. A STOP will automatically be transmitted.
SB or MB will be set as normal, but CLKHOLD will be released. The MEXTTOUT and BUSERR status bits will be set.
This bit is not synchronized.

Value	Description
0	Time-out disabled
1	Time-out enabled

Bit 16 – PINOUT Pin Usage

This bit set the pin usage to either two- or four-wire operation:
This bit is not synchronized.

Value	Description
0	4-wire operation disabled.
1	4-wire operation enabled.

Bit 7 – RUNSTDBY Run in Standby

This bit defines the functionality in standby sleep mode.
This bit is not synchronized.

Value	Description
0	GCLK_SERCOMx_CORE is disabled and the I^2C host will not operate in standby sleep mode.
1	GCLK_SERCOMx_CORE is enabled in all sleep modes.

Bits 4:2 – MODE[2:0] Operating Mode

These bits must be written to 0x5 to select the I^2C host serial communication interface of the SERCOM.
These bits are not synchronized.

Bit 1 – ENABLE Enable

Due to synchronization, there is delay from writing CTRLA.ENABLE until the peripheral is enabled/disabled.
The value written to CTRL.ENABLE will read back immediately and the Synchronization Enable Busy bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.
This bit is not enable-protected.

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

Bit 0 – SWRST Software Reset

Writing '0' to this bit has no effect.
Writing '1' to this bit resets all registers in the SERCOM, except DBGCTRL, to their initial state, and the SERCOM will be disabled.
Writing '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded. Any register write access during the ongoing reset will result in an APB error. Reading any register will return the reset value of the register.
Due to synchronization there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the reset is complete.
This bit is not enable-protected.
Note: During a SWRST, access to registers/bits without SWRST are disallowed until SYNCBUSY.SWRST cleared by hardware.

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

30.10.2 Control B

Name: CTRLB
Offset: 0x04
Reset: 0x00000000
Property: PAC Write-Protection, Enable-Protected, Write-Synchronized

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	FIFOCLR[1:0]					ACKACT	CMD[1:0]	
Access	R/W	R/W				R/W	W	W
Reset	0	0				0	0	0

Bit	15	14	13	12	11	10	9	8
							QCEN	SMEN
Access							R/W	R/W
Reset							0	0

Bit	7	6	5	4	3	2	1	0
Access								
Reset								

Bits 23:22 – FIFOCLR[1:0] FIFO Clear
When these bits are set, the corresponding FIFO will be cleared. The bits will automatically clear when SYNCBUSY.SYSOP = 0.
These bits are not enable-protected.

FIFOCLR[1:0]	Name	Description
0x0	NONE	No action
0x1	TXFIFO	Clear TX FIFO
0x2	RXFIFO	Clear RX FIFO
0x3	BOTH	Clear both TX/RX FIFO

Bit 18 – ACKACT Acknowledge Action
This bit defines the I^2C Host's acknowledge behavior after a data byte is received from the I^2C Client. The acknowledge action is executed when a command is written to CTRLB.CMD, or if Smart mode is enabled (CTRLB.SMEN is written to one), when DATA.DATA is read.
This bit is not enable-protected.
This bit is not write-synchronized.

Value	Description
0	Send ACK.
1	Send NACK.

Bits 17:16 – CMD[1:0] Command
Writing these bits triggers a Host operation as described below. The CMD bits are strobe bits, and always read as zero. The acknowledge action is only valid in Host Read mode. In Host Write mode, a command will only result in a repeated Start or Stop condition. The CTRLB.ACKACT bit and the CMD bits can be written at the same time, and then the acknowledge action will be updated before the command is triggered.
Commands can only be issued when either the Client on Bus Interrupt flag (INTFLAG.SB) or Host on Bus Interrupt flag (INTFLAG.MB) is '1'.

If CMD 0x1 is issued, a repeated start will be issued followed by the transmission of the current address in ADDR.ADDR. If another address is desired, ADDR.ADDR must be written instead of the CMD bits. This will trigger a repeated start followed by transmission of the new address.
Issuing a command will set the System Operation bit in the Synchronization Busy register (SYNCBUSY.SYSOP).

Table 30-4. Command Description

CMD[1:0]	Direction	Action
0x0	X	(No action)
0x1	X	Execute acknowledge action succeeded by repeated Start
0x2	0 (Write)	No operation
	1 (Read)	Execute acknowledge action succeeded by a byte read operation
0x3	X	Execute acknowledge action succeeded by issuing a Stop condition

These bits are not enable-protected.

Bit 9 – QCEN Quick Command Enable

This bit is not write-synchronized.

Value	Description
0	Quick Command is disabled.
1	Quick Command is enabled.

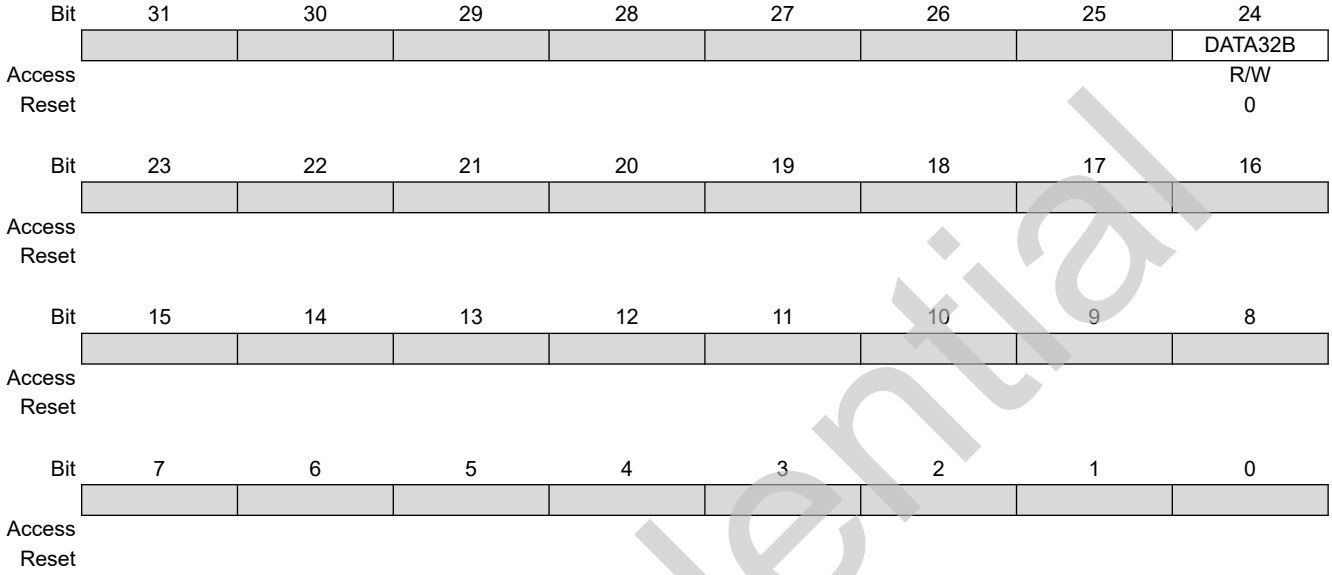
Bit 8 – SMEN Smart Mode Enable

When Smart mode is enabled, acknowledge action is sent when DATA.DATA is read.
This bit is not write-synchronized.

Value	Description
0	Smart mode is disabled.
1	Smart mode is enabled.

30.10.3 Control C

Name: CTRLC
Offset: 0x08
Reset: 0x00000000
Property: PAC Write-Protection, Enable-Protected



Bit 24 – DATA32B Data 32 Bit
This bit enables 32-bit data writes and reads to/from the DATA register.

Value	Description
0	Data transactions to/from DATA are 8-bit in size
1	Data transactions to/from DATA are 32-bit in size

30.10.4 Baud Rate

Name: BAUD
Offset: 0x0C
Reset: 0x0000
Property: PAC Write-Protection, Enable-Protected

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
	BAUDLOW[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	7	6	5	4	3	2	1	0
	BAUD[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 15:8 – BAUDLOW[7:0] Host Baud Rate Low
If this bit field is non-zero, the SCL low time will be described by the value written.
For more details on how to calculate the frequency, see *Clock Generation – Baud-Rate Generator* from Related Links.

Bits 7:0 – BAUD[7:0] Host Baud Rate
This bit field is used to derive the SCL high time if BAUD.BAUDLOW is non-zero. If BAUD.BAUDLOW is zero, BAUD will be used to generate both high and low periods of the SCL.
For more details on how to calculate the frequency, see *Clock Generation – Baud-Rate Generator* from Related Links.

Related Links
27.6.2.3. Clock Generation – Baud-Rate Generator

30.10.5 Interrupt Enable Clear

Name: INTENCLR
Offset: 0x14
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
	ERROR			RXFF	TXFE		SB	MB
Access	R/W			R/W	R/W		R/W	R/W
Reset	0			0	0		0	0

Bit 7 – ERROR Error Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Error Interrupt Enable bit, which disables the Error interrupt.

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

Bit 4 – RXFF RX FIFO Full Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the RX FIFO Full bit, which disables the RX FIFO Full interrupt.

Value	Description
0	The RX FIFO Full interrupt is disabled.
1	The RX FIFO Full interrupt is enabled.

Bit 3 – TXFE TX FIFO Empty Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the TX FIFO Empty bit, which disables the TX FIFO Empty interrupt.

Value	Description
0	The TX FIFO Empty interrupt is disabled.
1	The TX FIFO Empty interrupt is enabled.

Bit 1 – SB Client on Bus Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Client on Bus Interrupt Enable bit, which disables the Client on Bus interrupt.

Value	Description
0	The Client on Bus interrupt is disabled.
1	The Client on Bus interrupt is enabled.

Bit 0 – MB Host on Bus Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Host on Bus Interrupt Enable bit, which disables the Host on Bus interrupt.

Value	Description
0	The Host on Bus interrupt is disabled.
1	The Host on Bus interrupt is enabled.

30.10.6 Interrupt Enable Set

Name: INTENSET
Offset: 0x16
Reset: 0x00
Property: PAC Write-Protection

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

Bit	7	6	5	4	3	2	1	0
	ERROR			RXFF	TXFE		SB	MB
Access	R/W			R/W	R/W		R/W	R/W
Reset	0			0	0		0	0

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 4 – RXFF RX FIFO Full Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the RX FIFO Full bit, which enables the RX FIFO Full interrupt.

Value	Description
0	The RX FIFO Full interrupt is disabled.
1	The RX FIFO Full interrupt is enabled.

Bit 3 – TXFE TX FIFO Empty Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the TX FIFO Empty bit, which enables the TX FIFO Empty interrupt.

Value	Description
0	The TX FIFO Empty interrupt is disabled.
1	The TX FIFO Empty interrupt is enabled.

Bit 1 – SB Client on Bus Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Client on Bus Interrupt Enable bit, which enables the Client on Bus interrupt.

Value	Description
0	The Client on Bus interrupt is disabled.
1	The Client on Bus interrupt is enabled.

Bit 0 – MB Host on Bus Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Host on Bus Interrupt Enable bit, which enables the Host on Bus interrupt.

Value	Description
0	The Host on Bus interrupt is disabled.
1	The Host on Bus interrupt is enabled.

30.10.7 Interrupt Flag Status and Clear

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

Bit	7	6	5	4	3	2	1	0
	ERROR			RXFF	TXFE		SB	MB
Access	R/W			R/W	R/W		R/W	R/W
Reset	0			0	0		0	0

Bit 7 – ERROR Error
This flag is cleared by writing '1' to it.
This bit is set when any error is detected. Errors that will set this flag have corresponding status bits in the STATUS register. These status bits are LENERR, SEXTTOUT, MEXTTOUT, LOWTOUT, ARBLOST, and BUSERR.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the flag.

Bit 4 – RXFF RX FIFO Full
This flag is set when RX FIFO Threshold locations are fulfilled.
The flag is cleared when the RX FIFO is empty.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the RX FIFO Full interrupt flag.

Bit 3 – TXFE TX FIFO Empty
This flag is set when TX FIFO Threshold locations are available.
The flag is cleared when the TX FIFO is full.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the TX FIFO Empty interrupt flag.

Bit 1 – SB Client on Bus
The Client on Bus flag (SB) is set when a byte is successfully received in Host Read mode, for example, no arbitration lost or bus error occurred during the operation. When this flag is set, the host forces the SCL line low, stretching the I^2C clock period. The SCL line will be released and SB will be cleared on one of the following actions:

- Writing to ADDR.ADDR
- Writing to DATA.DATA
- Reading DATA.DATA when Smart mode is enabled (CTRLB.SMEN)
- Writing a valid command to CTRLB.CMD

Writing '1' to this bit location will clear the SB flag. The transaction will not continue or be terminated until one of the above actions is performed.
Writing '0' to this bit has no effect.

Bit 0 – MB Host on Bus
This flag is set when a byte is transmitted in Host Write mode. The flag is set regardless of the occurrence of a bus error or an Arbitration Lost condition. MB is also set when arbitration is lost during sending of NACK in Host Read mode, or when issuing a Start condition if the bus state is unknown. When this flag is set and arbitration is not lost, the host forces the SCL line low, stretching the I^2C clock period. The SCL line will be released and MB will be cleared on one of the following actions:

- Writing to ADDR.ADDR
- Writing to DATA.DATA
- Reading DATA.DATA when Smart mode is enabled (CTRLB.SMEN)
- Writing a valid command to CTRLB.CMD

Writing '1' to this bit location will clear the MB flag. The transaction will not continue or be terminated until one of the above actions is performed.
Writing '0' to this bit has no effect.

30.10.8 Status

Name: STATUS
Offset: 0x1A
Reset: 0x0000
Property: Write-Synchronized

Bit	15	14	13	12	11	10	9	8
						LENERR	SEXTTOUT	MEXTTOUT
Access						R/W	R/W	R/W
Reset						0	0	0

Bit	7	6	5	4	3	2	1	0
	CLKHOLD	LOWTOUT	BUSSTATE[1:0]			RXNACK	ARBLOST	BUSERR
Access	R	R/W	R/W	R/W		R	R/W	R/W
Reset	0	0	0	0		0	0	0

Bit 10 – LENERR Transaction Length Error
This bit is set when automatic length is used for a DMA and/or 32-bit transaction and the client sends a NACK before ADDR.LEN bytes have been written by the host.
Writing '1' to this bit location will clear STATUS.LENERR. This flag is automatically cleared when writing to the ADDR register.
Writing '0' to this bit has no effect.
This bit is not write-synchronized.

Bit 9 – SEXTTOUT Client SCL Low Extend Time-Out
This bit is set if a client SCL low extend time-out occurs.
This bit is automatically cleared when writing to the ADDR register.
Writing '1' to this bit location will clear SEXTTOUT. Normal use of the I^2C interface does not require the SEXTTOUT flag to be cleared by this method.
Writing '0' to this bit has no effect.
This bit is not write-synchronized.

Bit 8 – MEXTTOUT Host SCL Low Extend Time-Out
This bit is set if a Host SCL low time-out occurs.
Writing '1' to this bit location will clear STATUS.MEXTTOUT. This flag is automatically cleared when writing to the ADDR register.
Writing '0' to this bit has no effect.
This bit is not write-synchronized.

Bit 7 – CLKHOLD Clock Hold
This bit is set when the host is holding the SCL line low, stretching the I^2C clock. Software should consider this bit when INTFLAG.SB or INTFLAG.MB is set.
This bit is cleared when the corresponding Interrupt flag is cleared and the next operation is given.
Writing '0' to this bit has no effect.
Writing '1' to this bit has no effect.
This bit is not write-synchronized.

Bit 6 – LOWTOUT SCL Low Time-Out
This bit is set if an SCL low time-out occurs.
Writing '1' to this bit location will clear this bit. This flag is automatically cleared when writing to the ADDR register.
Writing '0' to this bit has no effect.
This bit is not write-synchronized.

Bits 5:4 – BUSSTATE[1:0] Bus State
These bits indicate the current I^2C Bus state.

When in UNKNOWN state, writing 0x1 to BUSSTATE forces the bus state into the IDLE state. The bus state cannot be forced into any other state.
Writing BUSSTATE to idle will set SYNCBUSY.SYSOP.

Value	Name	Description
0x0	UNKNOWN	The Bus state is unknown to the I²C host and will wait for a Stop condition to be detected or wait to be forced into an Idle state by software
0x1	IDLE	The Bus state is waiting for a transaction to be initialized
0x2	OWNER	The I²C host is the current owner of the bus
0x3	BUSY	Some other I²C host owns the bus

Bit 2 – RXNACK Received Not Acknowledge

This bit indicates whether the last address or data packet sent was acknowledged or not.
Writing '0' to this bit has no effect.
Writing '1' to this bit has no effect.
This bit is not write-synchronized.

Value	Description
0	Client responded with ACK.
1	Client responded with NACK.

Bit 1 – ARBLOST Arbitration Lost

This bit is set if arbitration is lost while transmitting a high data bit or a NACK bit, or while issuing a Start or Repeated Start condition on the bus. The Host on Bus Interrupt flag (INTFLAG.MB) will be set when STATUS.ARBLOST is set.
Writing the ADDR.ADDR register will automatically clear STATUS.ARBLOST.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear it.
This bit is not write-synchronized.

Bit 0 – BUSERR Bus Error

This bit indicates that an illegal Bus condition has occurred on the bus, regardless of bus ownership. An illegal Bus condition is detected if a protocol violating start, repeated start or stop is detected on the I²C bus lines. A Start condition directly followed by a Stop condition is one example of a protocol violation. If a time-out occurs during a frame, this is also considered a protocol violation, and will set BUSERR.
If the I²C host is the bus owner at the time a bus error occurs, STATUS.ARBLOST and INTFLAG.MB will be set in addition to BUSERR.
Writing the ADDR.ADDR register will automatically clear the BUSERR flag.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear it.
This bit is not write-synchronized.

30.10.9 Synchronization Busy

Name: SYNCBUSY
Offset: 0x1C
Reset: 0x00000000

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

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

Bit 2 – SYSOP System Operation Synchronization Busy
Writing CTRLB.CMD, STATUS.BUSSTATE, ADDR or DATA when the SERCOM is enabled requires synchronization. When written, the SYNCBUSY.SYSOP bit will be set until synchronization is complete.
Writing CTRLB.CMD, STATUS.BUSSTATE, ADDR or DATA when the SERCOM is enabled requires synchronization. When written, the SYNCBUSY.SYSOP bit will be set until synchronization is complete.

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

Bit 1 – ENABLE SERCOM Enable Synchronization Busy
Enabling and disabling the SERCOM (CTRLA.ENABLE) requires synchronization. When written, the SYNCBUSY.ENABLE bit will be set until synchronization is complete.

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.
Note: During a SWRST, access to registers/bits without SWRST are disallowed until SYNCBUSY.SWRST cleared by hardware.

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

30.10.10 Address

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

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

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

Bit	15	14	13	12	11	10	9	8
	TENBITEN		LENEN			ADDR[10:8]		
Access	R/W		R/W			R/W	R/W	R/W
Reset	0		0			0	0	0

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

Bits 23:16 – LEN[7:0] Transaction Length
These bits define the transaction length of a DMA and/or 32-bit transaction from 0 to 255 bytes. The Transfer Length Enable (LENEN) bit must be written to '1' in order to use DMA.

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

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

Bit 13 – LENEN Transfer Length Enable

Value	Description
0	Automatic transfer length disabled.
1	Automatic transfer length enabled.

Bits 10:0 – ADDR[10:0] Address
When ADDR is written, the consecutive operation will depend on the bus state:
UNKNOWN: INTFLAG.MB and STATUS.BUSERR are set, and the operation is terminated.
BUSY: The I^2C host will await further operation until the bus becomes IDLE.
IDLE: The I^2C host will issue a start condition followed by the address written in ADDR. If the address is acknowledged, SCL is forced and held low, and STATUS.CLKHOLD and INTFLAG.MB are set.
OWNER: A repeated start sequence will be performed. If the previous transaction was a read, the acknowledge action is sent before the repeated start bus condition is issued on the bus. Writing ADDR to issue a repeated start is performed while INTFLAG.MB or INTFLAG.SB is set.
STATUS.BUSERR, STATUS.ARBLOST, INTFLAG.MB and INTFLAG.SB will be cleared when ADDR is written.
The ADDR register can be read at any time without interfering with ongoing bus activity, as a read access does not trigger the host logic to perform any bus protocol related operations.
The I^2C host control logic uses bit 0 of ADDR as the bus protocol's read/write flag (R/W); 0 for write and 1 for read.

30.10.11 Data

Name: DATA
Offset: 0x28
Reset: 0x00000000
Property: Read/Write

Bit	31	30	29	28	27	26	25	24
	DATA[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
	DATA[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
	DATA[15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

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

Bits 31:0 – DATA[31:0] Data

The host data register I/O location (DATA) provides access to the host transmit and receive data buffers. Reading valid data or writing data to be transmitted can be successfully done only when SCL is held low by the host (STATUS.CLKHOLD is set). An exception is reading the last data byte after the stop condition has been sent. Accessing DATA.DATA auto-triggers I^2C bus operations. The operation performed depends on the state of CTRLB.ACKACT, CTRLB.SMEN and the type of access (read/write).

When CTRLC.DATA32B=1, read and write transactions from/to the DATA register are 32 bit in size. Otherwise, reads and writes are 8 bit.

30.10.12 Debug Control

Name: DBGCTRL
Offset: 0x30
Reset: 0x00
Property: PAC Write-Protection

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

Bit 0 – DBGSTOP Debug Stop Mode
This bit controls functionality when the CPU is halted by an external debugger.

Value	Description
0	The baud-rate generator continues normal operation when the CPU is halted by an external debugger.
1	The baud-rate generator is halted when the CPU is halted by an external debugger.

30.10.13 FIFO Space

Name: FIFOSPACE
Offset: 0x34
Reset: 0x0000
Property: -

This register allows the user to identify the number of bytes present in each TX and RX FIFO.

Bit	15	14	13	12	11	10	9	8
				RXSPACE[4:0]				
Access				R	R	R	R	R
Reset				0	0	0	0	0

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

Bits 12:8 – RXSPACE[4:0] RX FIFO Filled Space
These bits return the number filled locations in the RX FIFO (bytes or words, depending on CTRLC.DATA32B setting).

Bits 4:0 – TXSPACE[4:0] TX FIFO Empty Space
These bits return the number of available locations in the TX FIFO (bytes or words, depending on CTRLC.DATA32B setting).

30.10.14 FIFO CPU Pointers

Name: FIFOPTR
Offset: 0x36
Reset: 0x0000
Property: -

This register provides a copy of internal CPU TX and RX FIFO pointers.

Bit	15	14	13	12	11	10	9	8
					CPURDPTR[3:0]			
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

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

Bits 11:8 – CPURDPTR[3:0] RX FIFO Filled Space
These bits return the CPURDPTR pointer value. These bits can be written only if the SERCOM is halted during debugging. Reading DATA register, will return RXFIFO[CPURDPTR] location value.

Bits 3:0 – CPUWRPTR[3:0] TX FIFO Filled Space
These bits return the CPURDPTR pointer value. These bits can be written only if the SERCOM is halted during debugging. When writting to DATA register, the DATA will be written to TXFIFO[CPUWRPTR] location.

31. Quad Serial Peripheral Interface (QSPI)

31.1 Overview

The Quad SPI Interface (QSPI) circuit is a synchronous serial data link that provides communication with external devices in Host mode.

The QSPI can be used in “SPI mode” to interface serial peripherals, such as ADCs, DACs, LCD controllers and sensors, or in “Serial Memory Mode” to interface serial Flash memories.

The QSPI allows the system to execute code directly from a serial Flash memory (XIP) without code shadowing to SRAM. The serial Flash memory mapping is seen in the system as other memories (ROM, SRAM, embedded Flash memories, and so on.). XIP mode is not available for a secured or code protected device for security reasons. This is achieved by blocking off AHB accesses (only instruction) on the CPU side from QSPI. QSPI returns error on the transactions where CPU is requesting for instructions from QSPI connected external memories.

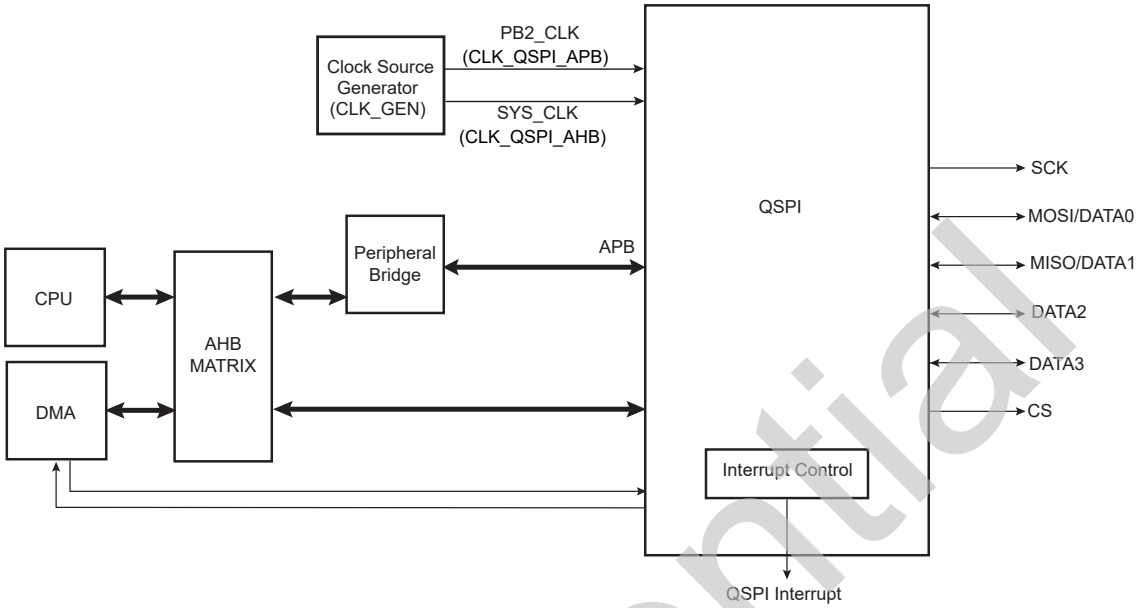
With the support of the quad-SPI protocol, the QSPI allows the system to use high performance serial Flash memories which are small and inexpensive, in place of larger and more expensive parallel Flash memories.

31.2 Features

- Host SPI Interface:
  - Programmable clock phase and clock polarity
  - Programmable transfer delays between consecutive transfers, between clock and data, between deactivation and activation of chip select (CS)
- SPI Mode:
  - To use serial peripherals, such as ADCs, DACs, LCD controllers, CAN controllers, and sensors
  - 8-bit, 16-bit, or 32-bit programmable data length
- Serial Memory Mode:
  - To use serial Flash memories operating in single-bit SPI, Dual SPI and Quad SPI
  - Supports “execute in place” (XIP). The system can execute code directly from a Serial Flash memory
  - Flexible instruction register, to be compatible with all serial Flash memories
  - 32-bit Address mode (default is 24-bit address) to support serial Flash memories larger than 128 Mbit
  - Continuous Read mode
  - Scrambling/Unscrambling “On-the-Fly”
  - Double data rate support
- Connection to DMA Channel Capabilities Optimizes Data Transfers
  - One channel for the receiver and one channel for the transmitter
- Register Write Protection

31.3 Block Diagram

Figure 31-1. QSPI Block Diagram



31.4 Signal Description

Table 31-1. Quad-SPI Signals

Signal	Description	Type
SCK	Serial Clock	Output
CS	Chip Select	Output
MOSI(DATA0)	Data Output (Data Input Output 0)	Output (Input/Output)
MISO(DATA1)	Data Input (Data Input Output 1)	Input (Input/Output)
DATA2	Data Input Output 2	Input/Output
DATA3	Data Input Output 3	Input/Output

Notes:

1. MOSI and MISO are used for single-bit SPI operation.
2. DATA0-DATA1 are used for Dual SPI operation.
3. DATA0-DATA3 are used for Quad SPI operation.

See *I/O Ports and Peripheral Pin Select (PPS)* from Related Links for details on the pin mapping for the QSPI peripheral.

Related Links

5. I/O Ports and Peripheral Pin Select (PPS)

31.5 Product Dependencies

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

31.5.1 I/O Lines

Using the QSPI I/O lines requires the I/O pins to be configured using the System Configuration registers (See *System Configuration and Register Locking (CFG)* from Related Links) (QSPI_HSEN of CFGCON1/DEVCFG1 register) for direct or PPS. If QSPI pins are selected through PPS, the PPS registers has to be configured (See *I/O Ports and Peripheral Pin Select (PPS)* from Related Links).

If QSPI_HSEN =1, QSPI uses dedicated pins.

If QSPI_HSEN = 0, QSPI uses PPS path, and I/O pins are multiplexed to pins groups defined in PPS section.

Related Links

5. I/O Ports and Peripheral Pin Select (PPS)
18. System Configuration and Register Locking (CFG)

31.5.2 Power Management

The QSPI will continue to operate in any Sleep mode where the selected source clock is running. The QSPI interrupts can be used to wake up the device from sleep modes. See *Power Management Unit (PMU)* from Related Links for details on the different sleep modes.

Related Links

14. Power Management Unit (PMU)

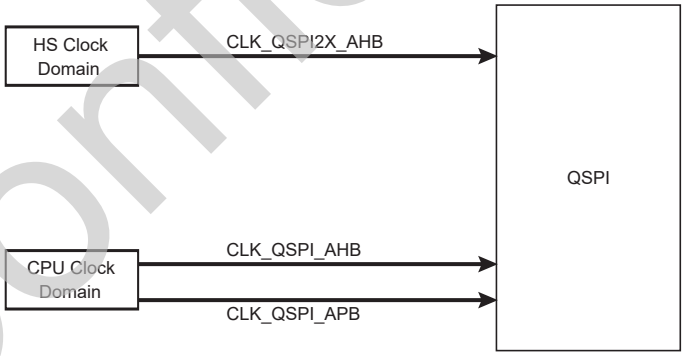
31.5.3 Clocks

An AHB clock (CLK_QSPI_AHB) is required to clock the QSPI. In PIC32CX-BZ3, SYS_CLK is the AHB clock and can be configured in the CRU.

A FAST clock (CLK_QSPI2X_AHB) is required to clock the QSPI. This clock can be enabled and disabled in the CFGCON1 register, bit 29 (CFGCON1.QSPIDDRM). When using QSPI DDR (Double Data Rate) Mode, the System Clock (SYS_CLK) must be less than or equal to 32 MHz.

PB2_CLK is CLK_QSPI_APB clock and can be configured in CRU registers.

Figure 31-2. QSPI Clock Organization



Important: The CLK_QSPI2x_AHB must be two times faster to CLK_QSPI_AHB when the QSPI is operated in the DDR mode. In Single Data Rate (SDR), the CLK_QSPI2x_AHB is not used.

The CLK_QSPI_APB, CLK_QSPI_AHB, and CLK_QSPI2X_AHB, respectively, are all synchronous, but can be divided by a prescaler.

31.5.4 DMA

The DMA request lines are connected to the DMA Controller (DMAC). Using the QSPI DMA requests requires the DMA Controller to be configured first.

Note: DMAC write access must be 32-bit aligned. If a single byte is to be written in a 32-bit word, the rest of the word must be filled with 'ones'.

31.5.5 Interrupts

The interrupt request lines are connected to the interrupt controller. Using the QSPI interrupts requires the interrupt controller to be configured first. See *Nested Vector Interrupt Controller (NVIC)* from Related Links.

Related Links

8.2. Nested Vector Interrupt Controller (NVIC)

31.5.6 Events

Not applicable.

31.5.7 Debug Operation

When the CPU is halted in debug mode the QSPI continues normal operation. If the QSPI is configured in a way that requires it to be periodically serviced by the CPU through interrupts or similar, improper operation or data loss may result during debugging.

31.5.8 Register Access Protection

All registers with write-access are optionally write-protected by the peripheral access controller (PAC), except the following registers:

- Control A (CTRLA) register
- Transmit Data (TXDATA) register
- Interrupt Flag Status and Clear (INTFLAG) register
- Scrambling Key (SCRAMBKEY) register

PAC write-protection is denoted by the ‘PAC Write-Protection’ property in the register description.

Write-protection does not apply to accesses through an external debugger.

31.6 Functional Description

31.6.1 Principle of Operation

The QSPI is a high-speed synchronous data transfer interface. It allows high-speed communication between the device and peripheral or serial memory devices.

The QSPI operates as a host. It initiates and controls all data transactions.

When transmitting, the TXDATA register can be loaded with the next character to be transmitted during the current transmission.

When receiving, the data is transferred to the RXDATA register, and the receiver is ready for a new character.

31.6.2 Basic Operation

31.6.2.1 Initialization

After Power-On Reset, this peripheral is enabled .

31.6.2.2 Enabling, Disabling, and Resetting

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

The peripheral is disabled by writing a '0' to CTRLA.ENABLE.

The peripheral is reset by writing a '1' to the Software Reset bit (CTRLA.SWRST).

31.6.3 Transfer Data Rate

By default, the QSPI module is enabled in single data rate mode. In this operating mode, the CLK_QSPI2X_AHB clock is not used and must be disabled.

The dual data rate operating mode is enabled by writing a '1' to the Double Data Rate Enable bit in the CFGCON1 register (CFGCON1.QSPIDDRM). This operating mode requires the CLK_QSPI2X_AHB clock and must be enabled before writing the DDREN bit.

31.6.4 Serial Clock Baud Rate

The QSPI Baud rate clock is generated by dividing the module clock (CLK_QSPI_AHB) by a value between 1 and 255.

This allows a maximum operating baud rate at up to Host Clock and a minimum operating baud rate of CLK_QSPI_AHB divided by 255.

At reset, BAUD = 0 and the user has to program a valid value before performing the first transfer.

31.6.5 Serial Clock Phase and Polarity

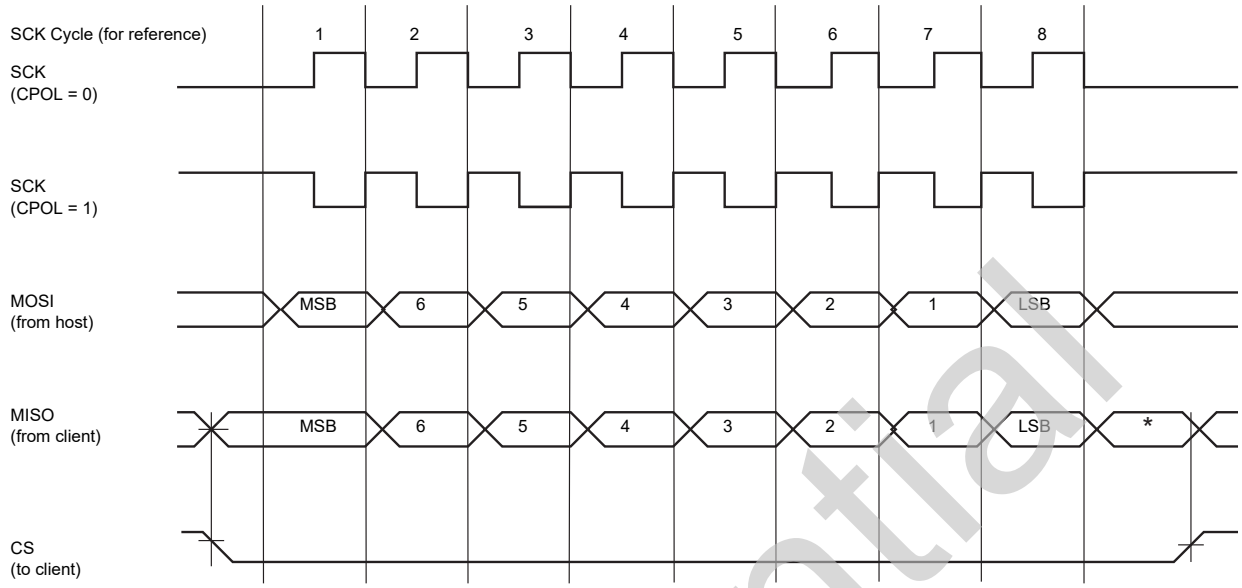
Four combinations of polarity and phase are available for data transfers. Writing the Clock Polarity bit in the QSPI Baud register (BAUD.CPOL) selects the polarity. The Clock Phase bit in the BAUD register programs the clock phase (BAUD.CPHA). These two parameters determine the edges of the clock signal on which data is driven and sampled. Each of the two parameters has two possible states, resulting in four possible combinations.
Note: The polarity/phase combinations are incompatible. Thus, the interfaced client must use the same parameter values to communicate.

Table 31-2. SPI Transfer Mode

Clock Mode	BAUD.CPOL	BAUD.CPHA	Shift SCK Edge	Capture SCK Edge	SCK Inactive Level
0	0	0	Falling	Rising	Low
1	0	1	Rising	Falling	Low
2	1	0	Rising	Falling	High
3	1	1	Falling	Rising	High

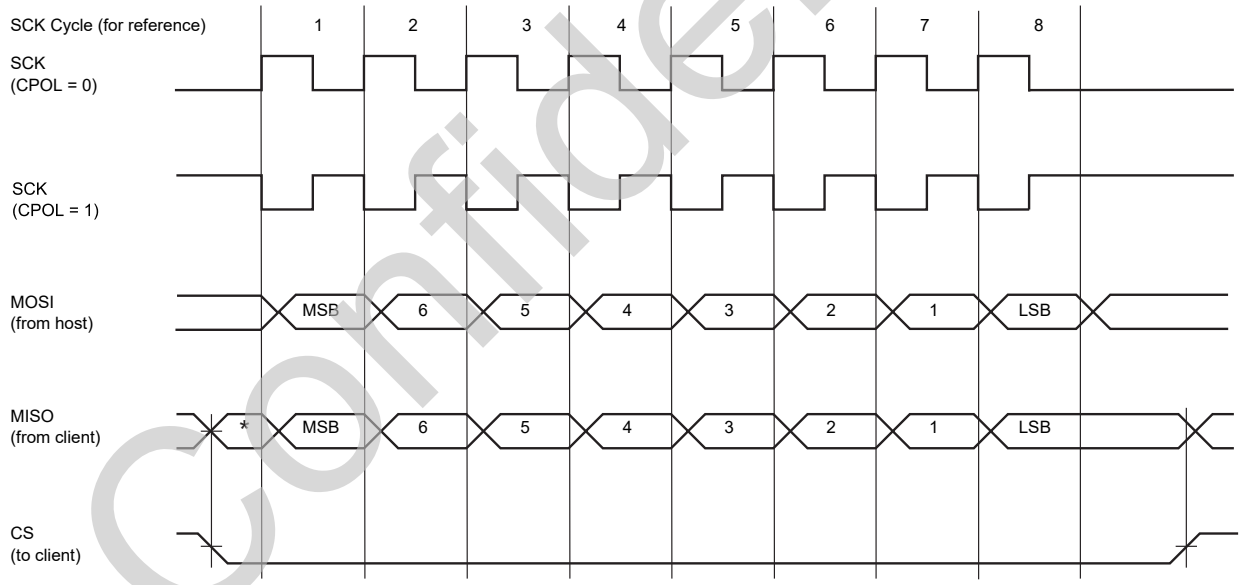
Figure 31-3. QSPI Transfer Modes (BAUD.CPHA = 0, 8-bit transfer)

Figure 31-4.



* Not defined, but normally MSB of previous character received

QSPI Transfer Modes (BAUD.CPHA = 1, 8-bit transfer)



* Not defined, but normally LSB of previous character received

31.6.6 Transfer Delays

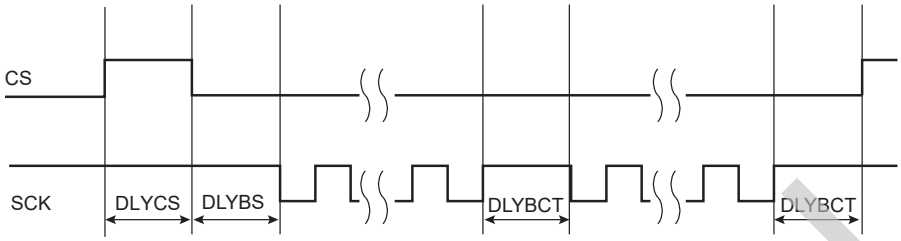
The QSPI supports several consecutive transfers while the chip select is active. Three delays can be programmed to modify the transfer waveforms:

- The delay between the inactivation and the activation of CS is programmed by writing the Minimum Inactive CS Delay bit field in the Control B register (CTRLB.DLYCS), allowing to tune the minimum time of CS at high level.
- The delay between consecutive transfers is programmed by writing the Delay Between Consecutive Transfers bit field in the Control B register (CTRLB.DLYBCT), allowing to insert a delay between two consecutive transfers. In Serial Memory mode, this delay is not programmable and DLYBCT settings are ignored.

- The delay before SCK is programmed by writing the Delay Before SCK bit field in the BAUD register (BAUD.DLYBS), allowing to delay the start of SPCK after the chip select has been asserted.

These delays allow the QSPI to be adapted to the interfaced peripherals and their speed and bus release time.

Figure 31-5. Programmable Delay



31.6.7 QSPI SPI Mode

In this mode, the QSPI acts as a regular SPI Host.

To activate this mode, the MODE bit in the Control B register must be cleared (CTRLB.MODE=0).

31.6.7.1 SPI Mode Operations

The QSPI in standard SPI mode operates on the clock generated by the internal programmable baud rate generator. It fully controls the data transfers to and from the client connected to the SPI bus. The QSPI drives the chip select line to the client (CS) and the serial clock signal (SCK).

The QSPI features a single internal shift register and two holding registers: the Transmit Data Register (TXDATA) and the Receive Data Register (RXDATA). The holding registers maintain the data flow at a constant rate.

After enabling the QSPI, a data transfer begins when the processor writes to the TXDATA. The written data is immediately transferred into the internal shift register and transfer on the SPI bus starts. While the data in the internal shift register is shifted on the MOSI line, the MISO line is sampled and shifted into the internal shift register. Receiving data cannot occur without transmitting data.

If new data is written in TXDATA during the transfer, it stays in TXDATA until the current transfer is completed. Then, the received data is transferred from the internal shift register to the RXDATA, the data in TXDATA is loaded into the internal shift register, and a new transfer starts.

The transfer of data written in TXDATA in the internal shift register is indicated by the Transmit Data Register Empty (DRE) bit in the Interrupt Flag Status and Clear register (INTFLAG.DRE). When new data is written in TXDATA, this bit is cleared. The DRE bit is used to trigger the Transmit DMA channel.

The end of transfer is indicated by the Transmission Complete flag (INTFLAG.TXC). If the transfer delay for the last transfer was configured to be greater than 0 (CTRLB.DLYBCT), TXC is set after the completion of the delay. The module clock (CLK_QSPI_AHB) can be switched off at this time.

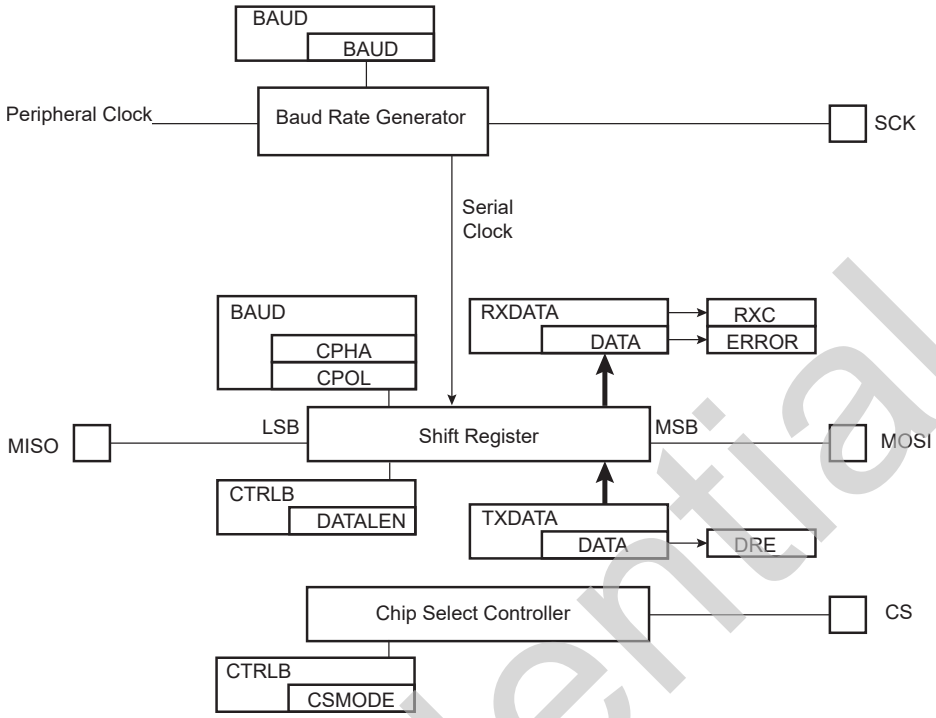
Ongoing transfer of received data from the internal shift register into RXDATA is indicated by the Receive Data Register Full flag (INTFLAG.RXC). When the received data is read, the RXC bit is cleared.

If the RXDATA has not been read before new data is received, the Overrun Error flag in INTFLAG register (INTFLAG.ERROR) is set. As long as this flag is set, data is loaded in RXDATA.

The SPI Mode Block Diagram shows a flow chart describing how transfers are handled.

31.6.7.2 SPI Mode Block Diagram

Figure 31-6. SPI Mode Block Diagram



31.6.7.3 SPI Mode Flow Diagram

Figure 31-7. SPI Mode Flow Diagram

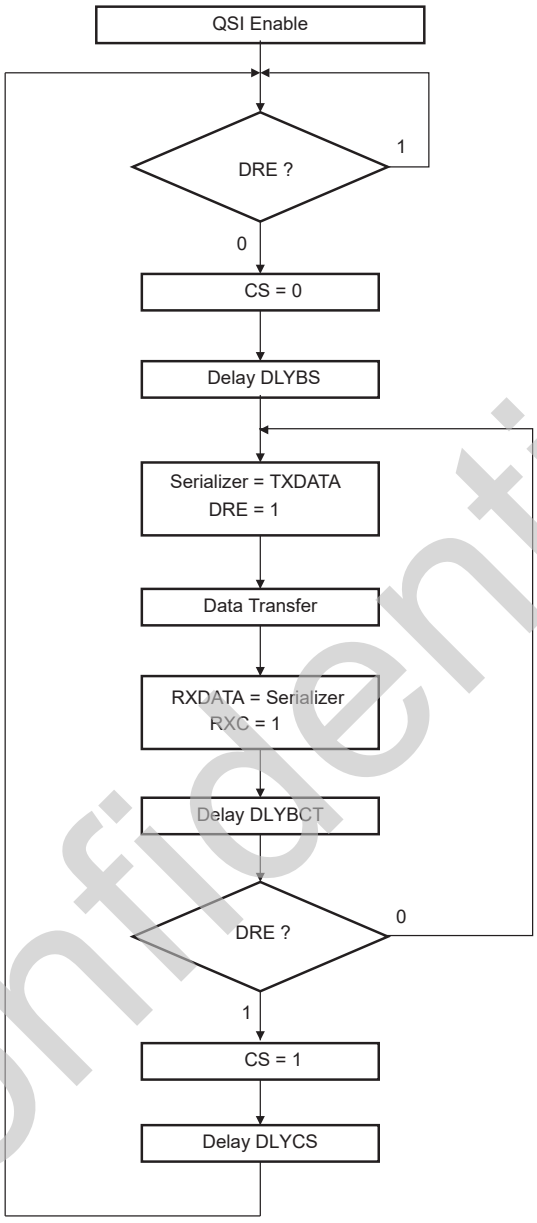
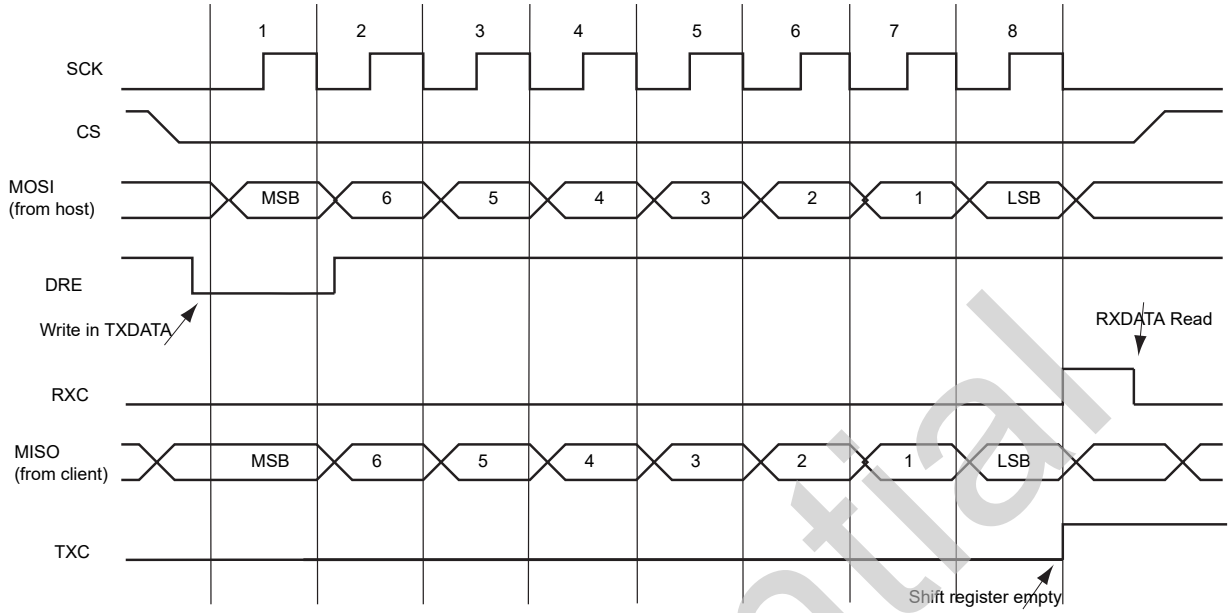


Figure 31-8. Interrupt Flags Behaviour



31.6.7.4 Peripheral Deselection with DMA

When the Direct Memory Access Controller is used, the Chip Select line will remain low during the whole transfer since the Transmit Data Register Empty flag in the Interrupt Flag Status and Clear register (INTFLAG.DRE) is managed by the DMA itself. The reloading of the TXDATA by the DMA is done as soon as INTFLAG.DRE flag is set. In this case, setting the Chip Select Mode bit field in the Control B register (CTRLB.CSMODE) to 0x1 is not mandatory.

However, it may happen that when other DMA channels connected to other peripherals are in use as well, the QSPI DMA could be delayed by another DMA transfer with a higher priority on the bus. Having DMA buffers in slower memories like Flash memory or SDRAM (compared to fast internal SRAM), may lengthen the reload time of the TXDATA by the DMA as well. This means that TXDATA might not be reloaded in time to keep the Chip Select line low. In this case the Chip Select line may toggle between data transfer and according to some SPI Client devices, and the communication might get lost. Writing CTRLB.CSMODE = 0x1 can prevent this loss.

When CTRLB.CSMODE = 0x0, the CS does not rise in all cases between two transfers on the same peripheral. During a transfer on a Chip Select, the INTFLAG.DRE flag is raised as soon as the content of the TXDATA is transferred into the internal shifter. When this flag is detected the TXDATA can be reloaded. If this reload occurs before the end of the current transfer and if the next transfer is performed on the same Chip Select as the current transfer, the Chip Select is not de-asserted between the two transfers. This may lead to difficulties for interfacing with some serial peripherals requiring the Chip Select to be de-asserted after each transfer. To facilitate interfacing with such devices, it is recommended to write CTRLB.CSMODE to 0x2.

31.6.7.5 Peripheral Deselection without DMA

During multiple data transfers on a Chip Select without the DMA, the TXDATA is loaded by the processor, and the Transmit Data Register Empty flag in the Interrupt Flag Status and Clear register (INTFLAG.DRE) rises as soon as the content of the RXDATA is transferred into the internal shift register. When this flag is detected high, the TXDATA can be reloaded. If this reload-by-processor occurs before the end of the current transfer and if the next transfer is performed on the same Chip Select as the current transfer, the Chip Select is not de-asserted between the two transfers.

Depending on the application software handling the flags or servicing other interrupts or other tasks, the processor may not reload the TXDATA in time to keep the Chip Select active (low). A null Delay Between Consecutive Transfer bit field value in the CTRLB register (CTRLB.DLYBCT) will give even less time for the processor to reload the TXDATA. With some SPI client peripherals, requiring the Chip Select line to remain active (low) during a full set of transfers might lead to communication errors.

To facilitate interfacing with such devices, the Chip Select Mode bit field in the CTRLB register (CTRLB.CSMODE) can be written to 0x1. This allows the Chip Select lines to remain in their current state (low = active) until the end of transfer is indicated by the Last Transfer bit in the CTRLA register (CTRLA.LASTXFER). Even if the TXDATA is not reloaded the Chip Select will remain active. To have the Chip Select line rise at the end of the last data transfer, the LASTXFER bit in the CTRLA must be set before writing the last data to transmit into the TXDATA.

31.6.8 QSPI Serial Memory Mode

In this mode the QSPI acts as a serial Flash memory controller. The QSPI can be used to read data from the serial Flash memory allowing the CPU to execute code from it (XIP execute in place). The QSPI can also be used to control the serial Flash memory (Program, Erase, Lock, and so on) by sending specific commands. In this mode, the QSPI is compatible with single-bit SPI, Dual-SPI and Quad-SPI protocols.

To activate this mode, the MODE bit in Control B register must be set to one (CTRLB.MODE = 1).

In serial memory mode, data cannot be transferred by the TXDATA and the RXDATA, but by writing or reading the QSPI memory space (0x0400 0000–0x0500 0000).

Caching can be enabled using the CMCC module along with configuring CFGCON1.QSCHE_EN bit.

Important: QSPI memory space region can be cached to improve data transfer speed. However, external Flash devices which have command/status registers mapped in the QSPI memory space region must be managed carefully by applying any one of the following configurations:

- Data cache must be disabled.
- If data cache is required, then cache line must be invalidated before reading the status register.

31.6.8.1 Instruction Frame

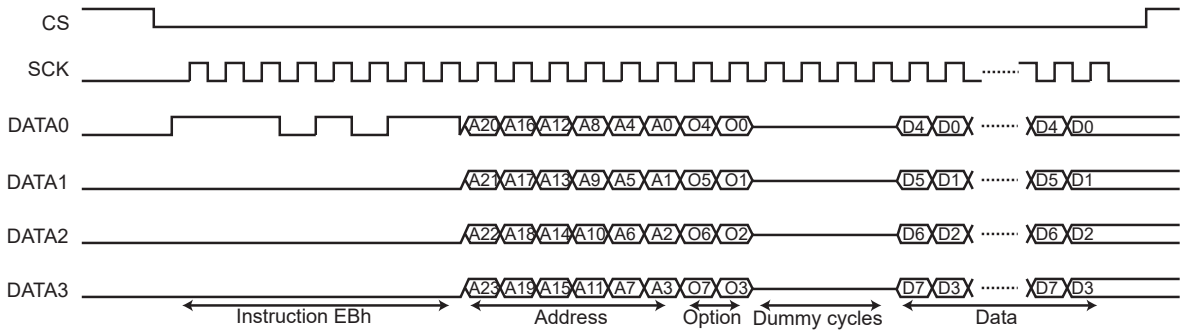
In order to control serial flash memories, the QSPI is able to sent instructions by the SPI bus (ex: READ, PROGRAM, ERASE, LOCK, etc.). Because instruction set implemented in serial flash memories is memory vendor dependant, the QSPI includes a complete instruction registers, which makes it very flexible and compatible with all serial flash memories.

An instruction frame includes:

- An instruction code (size: 8 bits). The instruction can be optional in some cases.
- An address (size: 24 bits or 32 bits). The address is optional but is required by instructions such as READ, PROGRAM, ERASE, LOCK. By default the address is 24 bits long, but it can be 32 bits long to support serial flash memories larger than 128 Mbit (16 Mbyte).
- An option code (size: 1/2/4/8 bits). The option code is optional but is useful for activate the “XIP mode” or the “Continuous Read Mode” for READ instructions, in some serial flash memory devices. These modes allow to improve the data read latency.
- Dummy cycles. Dummy cycles are optional but required by some READ instructions.
- Data bytes are optional. Data bytes are present for data transfer instructions such as READ or PROGRAM.

The instruction code, the address/option and the data can be sent with Single-bit SPI, Dual SPI or Quad SPI protocols.

Figure 31-9. Instruction Frame



31.6.8.2 Instruction Frame Sending

To send an instruction frame, the user must first configure the address to send by writing the field ADDR in the Instruction Address Register (INSTRADDR.ADDR). This step is required if the instruction frame includes an address and no data. When data is present, the address of the instruction is defined by the address of the data accesses in the QSPI memory space, and not by the INSTRADDR register.

If the instruction frame includes the instruction code and/or the option code, the user must configure the instruction code and/or the option code to send by writing the fields INST and OPTCODE bit fields in the Instruction Control Register (INSTRCTRL.OPTCODE, INSTRCTRL.INSTR).

Then, the user must write the Instruction Frame Register (INSTRFRAME) to configure the instruction frame depending on which instruction must be sent. If the instruction frame does not include data, writing in this register triggers the send of the instruction frame in the QSPI. If the instruction frame includes data, the send of the instruction frame is triggered by the first data access in the QSPI memory space.

The instruction frame is configured by the following bits and fields of INSTRFRAME:

- WIDTH field is used to configure which data lanes are used to send the instruction code, the address, the option code and to transfer the data. It is possible to use two unidirectional data lanes (MISO-MOSI Single-bit SPI), two bidirectional data lanes (DATA0 - DATA1 Dual SPI) or four bidirectional data lanes (DATA0 - DATA3).

Table 31-3. WIDTH Encoding

INSTRFRAME	Instruction	Address/Option	Data
0	Single-bit SPI	Single-bit SPI	Single-bit SPI
1	Single-bit SPI	Single-bit SPI	Dual SPI
2	Single-bit SPI	Single-bit SPI	Quad SPI
3	Single-bit SPI	Dual SPI	Dual SPI
4	Single-bit SPI	Quad SPI	Quad SPI
5	Dual SPI	Dual SPI	Dual SPI
6	Quad SPI	Quad SPI	Quad SPI
7	Reserved		

- INSTREN bit enables sending an instruction code.
- ADDREN bit enables sending of an address after the instruction code.
- OPTCODEEN bit enables sending of an option code after the address.
- DATAEN bit enables the transfer of data (READ or PROGRAM instruction).
- OPTCODELEN field configures the option code length (0 -> 1-bit / 1 -> 2-bit / 2 -> 4-bit / 3 -> 8-bit). The value written in OPTCODELEN must be consistent with value written in the field WIDTH. For example: OPTCODELEN = 0 (1-bit option code) is not coherent with WIDTH = 6 (option code sent with QuadSPI protocol, thus the minimum length of the option code is 4-bit).
- ADDRLEN bit configures the address length (0 -> 24 bits / 1-> 32 bits)
- TFRTYPE field defines which type of data transfer must be performed.
- DUMMYLEN field configures the number of dummy cycles when reading data from the serial flash memory. Between the address/option and the data, with some instructions, dummy cycles are inserted by the serial flash memory.

If data transfer is enabled, the user can access the serial memory by reading or writing the QSPI memory space following these rules:

- Reading from the serial memory, but not memory data (for example reading the JEDEC-ID or the STATUS), requires TFRTYPE to be written to 0x0.
- Reading from the serial memory, and particularly memory data, requires TFRTYPE to be written to '1'.
- Writing to the serial memory, but not memory data (for example writing the configuration or STATUS), requires TFRTYPE to be written to 0x2.
- Writing to the serial memory, and particularly memory data, requires TFRTYPE to be written to 0x3.

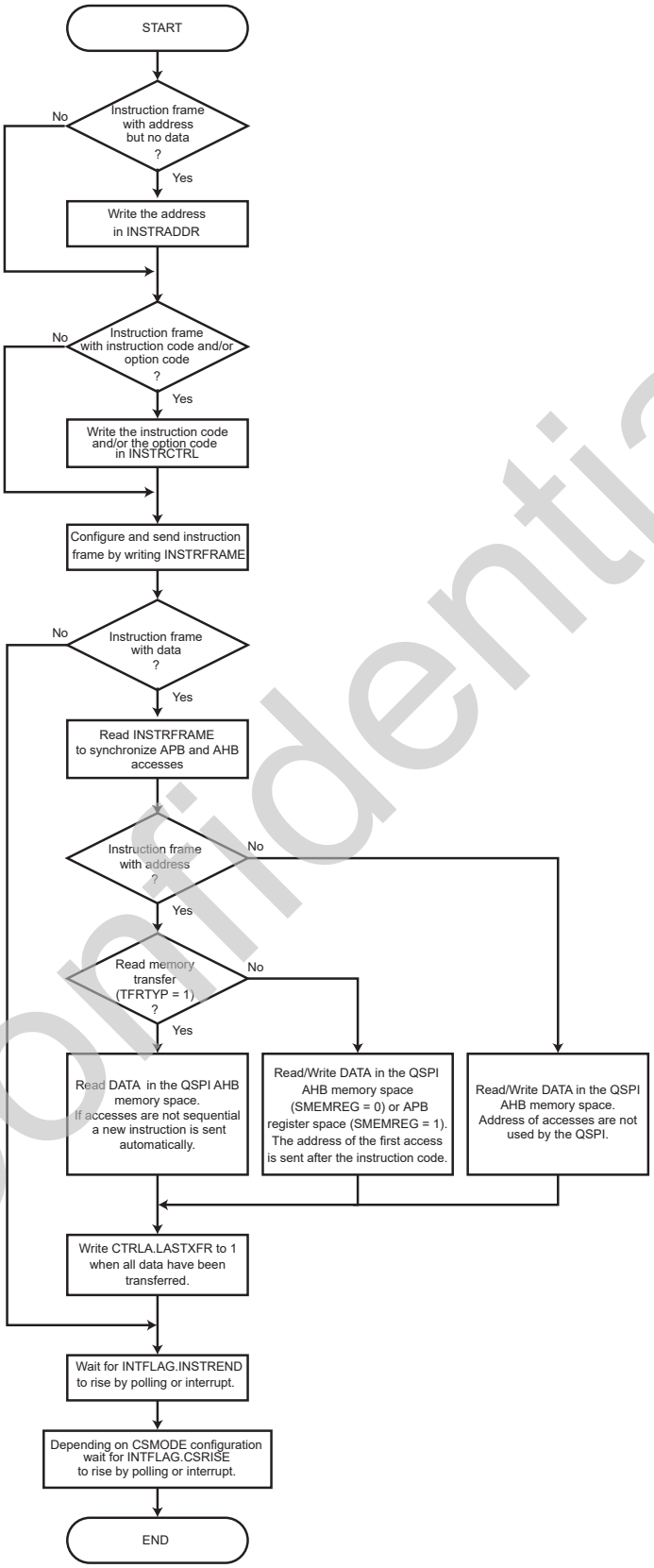
If TFRTYP has a value other than 0x1 and CTRLB.SMEMREG=0, the address sent in the instruction frame is the address of the first system bus accesses. The addresses of the subsequent access actions are not used by the QSPI. At each system bus access, an SPI transfer is performed with the same size. For example, a half-word system bus access leads to a 16-bit SPI transfer, and a byte system bus access leads to an 8-bit SPI transfer.

If CTRLB.SMEMREG=1, accesses are made via the QSPI registers and the address sent in the instruction frame is the address defined in the INSTRADDR register. Each time the INSTRFRAME or TXDATA registers are written, an SPI transfer is performed with a byte size. Another byte is read each time RXDATA register is read or written each time TXDATA register is written. The SPI transfer ends by writing the LASTXFER bit in Control A register (CTRLA.LASTXFER).

If TFRTYP=0x1, the address of the first instruction frame is the one of the first read access in the QSPI memory space. Each time the read accesses become non-sequential (addresses are not consecutive), a new instruction frame is sent with the last system bus access address. In this way, the system can read data at a random location in the serial memory. The size of the SPI transfers may differ from the size of the system bus read accesses.

When data transfer is not enabled, the end of the instruction frame is indicated when the INSTREND interrupt flag in the INTFLAG register is set. When data transfer is enabled, the user must indicate when data transfer is completed in the QSPI memory space by setting the bit LASTXFR in the CTRLA. The end of the instruction frame is indicated when the INSTREND interrupt flag in the INTFLAG register is set.

Figure 31-10. Instruction Transmission Flow Diagram



31.6.8.3 Read Memory Transfer

The user can access the data of the serial memory by sending an instruction with DATAEN=1 and TFRTYP=0x1 in the Instruction Frame register (INSTRFRAME).

In this mode the QSPI is able to read data at random address into the serial flash memory, allowing the CPU to execute code directly from it (XIP execute-in-place).

In order to fetch data, the user must first configure the instruction frame by writing the INSTRFRAME. Then data can be read at any address in the QSPI address space mapping. The address of the system bus read accesses match the address of the data inside the serial Flash memory.

When Fetch Mode is enabled, several instruction frames can be sent before writing the bit LASTXFR in the CTRLA. Each time the system bus read accesses become non-sequential (addresses are not consecutive), a new instruction frame is sent with the corresponding address.

31.6.8.4 Continuous Read Mode

The QSPI is compatible with Continuous Read Mode (CRM) which is implemented in some Serial Flash memories.

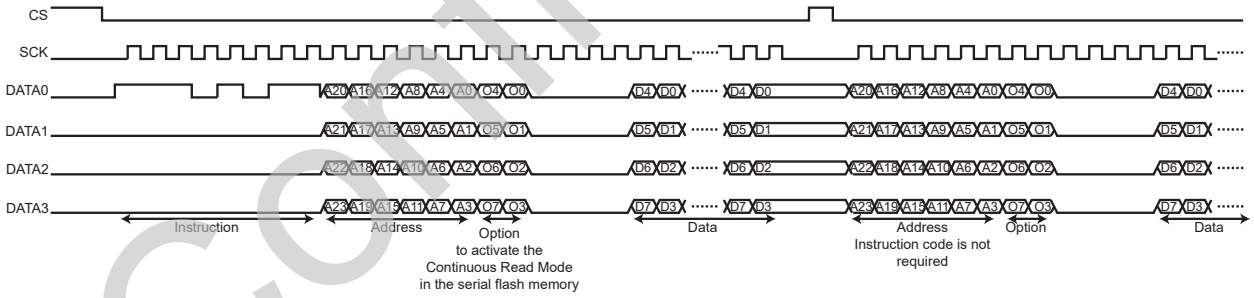
The CRM allows to reduce the instruction overhead by excluding the instruction code from the instruction frame. When CRM is activated in a Serial Flash memory (by a specific option code), the instruction code is stored in the memory. For the next instruction frames, the instruction code is not required, as the memory uses the stored one.

In the QSPI, CRM is used when reading data from the memory (INSTFRAME.TFRTYPE=0x1). The addresses of the system bus read accesses are often non-sequential, this leads to many instruction frames with always the same instruction code. By disabling the sending of the instruction code, the CRM reduces the access time of the data.

To be functional, this mode must be enabled in both the QSPI and the Serial Flash memory. The CRM is enabled in the QSPI by setting the CRM bit in the INSTRFRAME register (INSTFRAME.CRMODE=1, INSTFRAME.TFRTYPE must be 0x1). The CRM is enabled in the Serial Flash memory by sending a specific option code.

⚠ CAUTION If CRM is not supported by the Serial Flash memory or disabled, the CRMODE bit must not be set. Otherwise, data read out the Serial Flash memory is not valid.

Figure 31-11. Continuous Read Mode



31.6.8.5 Instruction Frame Transmission Examples

All waveforms in the following examples describe SPI transfers in SPI Clock mode 0 (BAUD.CPOL=0 and BAUD.CPHA=0). All system bus accesses described below refer to the system bus address phase. System bus wait cycles and system bus data phases are not shown.

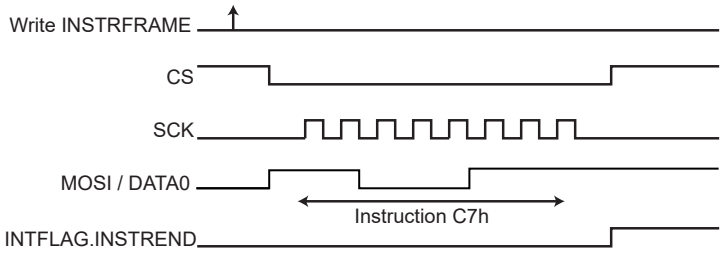
Example 31-1. Example 1

Instruction in Single-bit SPI, without address, without option, without data.

Command: CHIP ERASE (C7h).

- Write 0x0000_00C7 to INSTRCTRL register.
- Write 0x0000_0010 to INSTRFRAME register.
- Wait for INTFLAG.INSTREND to rise.

Figure 31-12. Instruction Transmission Waveform 1



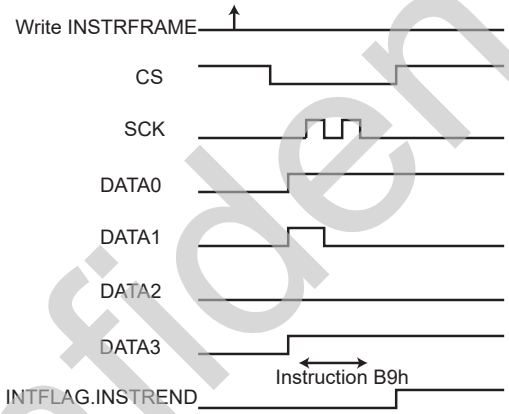
Example 31-2. Example 2

Instruction in Quad SPI, without address, without option, without data.

Command: POWER DOWN (B9h)

- Write 0x0000_00B9 to INSTRCTRL register.
- Write 0x0000_0016 to INSTRFRAME register.
- Wait for INTFLAG.INSTREND to rise.

Figure 31-13. Instruction Transmission Waveform 2



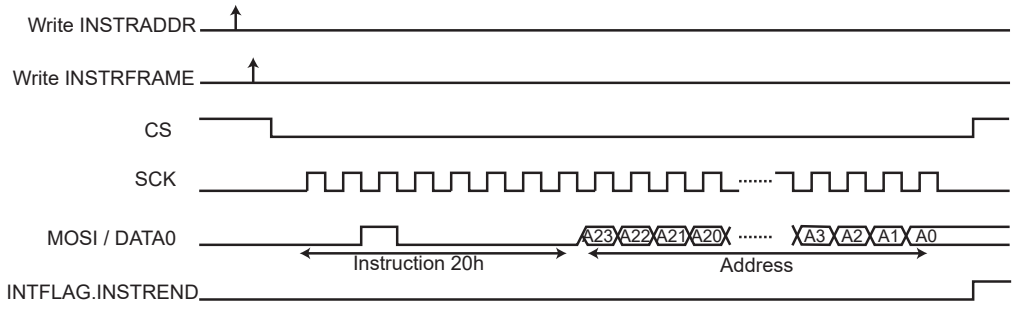
Example 31-3. Example 3

Instruction in Single-bit SPI, with address in Single-bit SPI, without option, without data.

Command: BLOCK ERASE (20h)

- Write the address (of the block to erase) to QSPI_AR.
- Write 0x0000_0020 to INSTRCTRL register.
- Write 0x0000_0030 toINSTRFRAME register.
- Wait for INTFLAG.INSTREND to rise.

Figure 31-14. Instruction Transmission Waveform 3



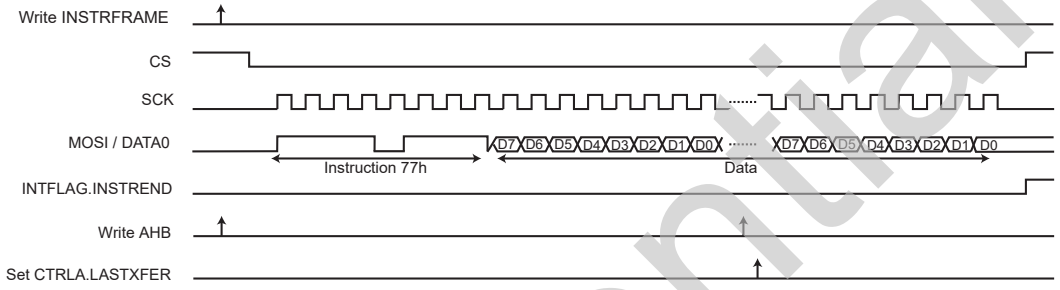
Example 31-4. Example 4

Instruction in Single-bit SPI, without address, without option, with data write in Single-bit SPI.

Command: SET BURST (77h)

- Write 0x0000_0077 to INSTRCTRL register.
- Write 0x0000_2090 to INSTRFRAME register.
- Read INSTRFRAME register (dummy read) to synchronize system bus accesses.
- Write data to the system bus memory space (0x0400_0000–0x0500_0000). The address of the system bus write accesses is not used.
- Write the LASTXFR bit in CTRLA register to '1'.
- Wait for INTFLAG.INSTREND to rise.

Figure 31-15. Instruction Transmission Waveform 4



Example 31-5. Example 5

Instruction in Single-bit SPI, with address in Dual SPI, without option, with data write in Dual SPI.

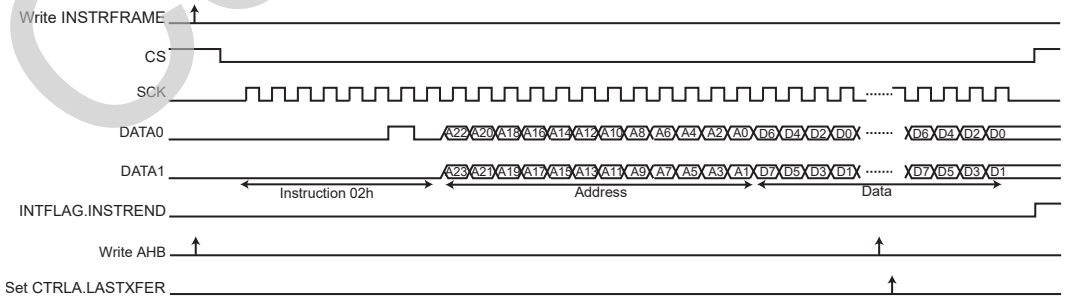
Command: BYTE/PAGE PROGRAM (02h)

- Write 0x0000_0002 to INSTRCTRL register.
- Write 0x0000_30B3 to INSTRFRAME register.
- Read INSTRFRAME register (dummy read) to synchronize system bus accesses.
- Write data to the QSPI system bus memory space (0x040 00000–0x0500_0000).

  The address of the first system bus write access is sent in the instruction frame.

  The address of the next system bus write accesses is not used.
- Write LASTXFR bit in CTRLA register to '1'.
- Wait for INTFLAG.INSTREND to rise.

Figure 31-16. Instruction Transmission Waveform 5



Example 31-6. Example 6

Instruction in Single-bit SPI, with address in Single-bit SPI, without option, with data read in Quad SPI, with eight dummy cycles.

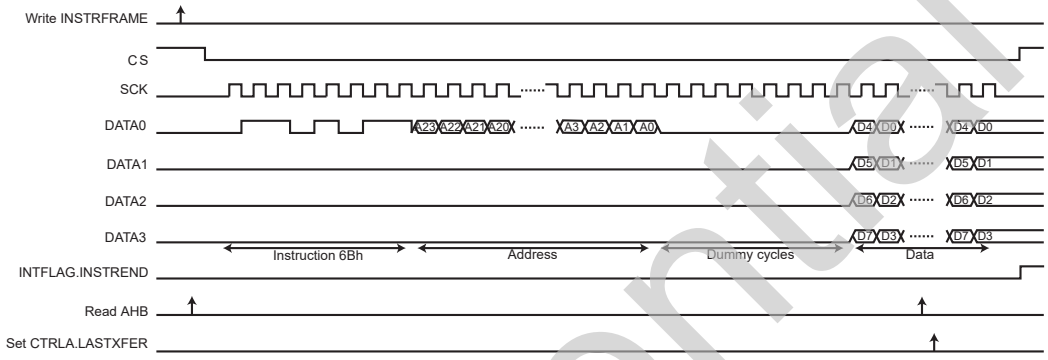
Command: QUAD_OUTPUT READ ARRAY (6Bh)

- Write 0x0000_006B to INSTRCTRL register.
- Write 0x0008_10B2 ti INSTRFRAME register.
- Read QSPI_IR (dummy read) to synchronize system bus accesses.
- Read data from the QSPI system bus memory space (0x040 00000–0x0500_0000).

  The address of the first system bus read access is sent in the instruction frame.

  The address of the next system bus read accesses is not used.
- Write the LASTXFR bit in CTRLA register to '1'.
- Wait for INTFLAG.INSTREND to rise.

Figure 31-17. Instruction Transmission Waveform 6



Example 31-7. Example 7

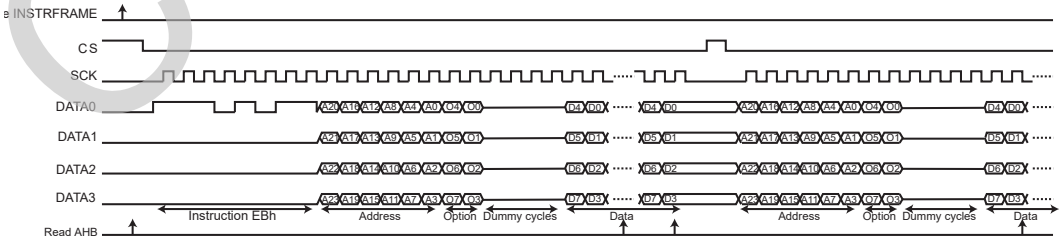
Instruction in Single-bit SPI, with address and option in Quad SPI, with data read from Quad SPI, with four dummy cycles, with fetch and continuous read.

Command: FAST READ QUAD I/O (EBh) - 8-BIT OPTION (0x30h)

- Write 0x0030_00EB to INSTRCTRL register.
- Write 0x0004_33F4 to INSTRFRAME register.
- Read INSTRFRAME register (dummy read) to synchronize system bus accesses.
- Read data from the QSPI system bus memory space (0x040 00000–0x0500_0000).

  Fetch is enabled, the address of the system bus read accesses is always used.
- Write LASTXFR bit in CTRLA register to '1'.
- Wait for INTFLAG.INSTREND to rise.

Figure 31-18. Instruction Transmission Waveform 7



Example 31-8. Example 8

Instruction in Quad SPI, with address in Quad SPI, without option, with data read from Quad SPI, with two dummy cycles, with fetch.

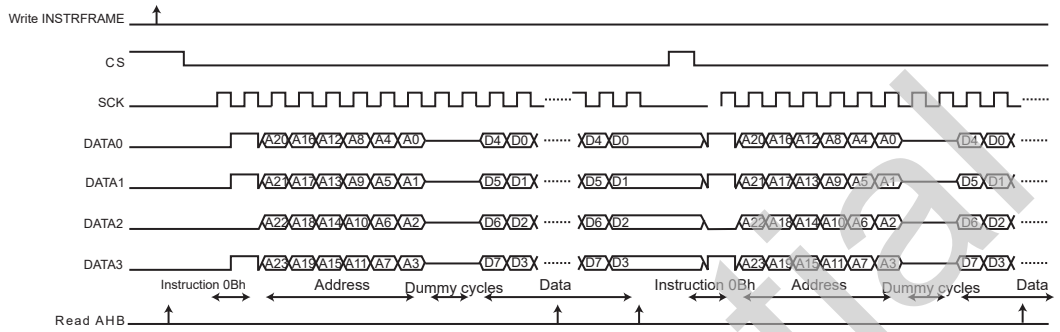
Command: HIGH-SPEED READ (0Bh)

- Write 0x0000_000B to INSTRCTRL register.

- Write 0x0002_20B6 to INSTRFRAME register.
- Read INSTRFRAME register (dummy read) to synchronize system bus accesses.
- Read data in the QSPI system bus memory space (0x040 00000–0x0500_0000).

  Fetch is enabled, the address of the system bus read accesses is always used.
- Write LASTXFR bit in CTRLA register to '1'.
- Wait for INTFLAG.INSTREND to rise.

Figure 31-19. Instruction Transmission Waveform 8



31.6.9 Scrambling/Unscrambling Function

The scrambling/unscrambling function cannot be performed on devices other than memories. Data is scrambled when written to memory and unscrambled when data is read.

The external data lines can be scrambled to prevent intellectual property data located in off-chip memories from being easily recovered by analyzing data at the package pin level of either the micro-controller or the QSPI client device (e.g., memory).

The scrambling/unscrambling function can be enabled by writing a '1' to the ENABLE bit in the Scrambling Control register (SCRAMBCTRL.ENABLE).

The scrambling and unscrambling are performed on-the-fly without impacting the throughput.

The scrambling method depends on the user-configurable Scrambling User Key in the Scrambling Key register (SCRAMBKEY.KEY). This register is only accessible in Write mode.

By default, the scrambling and unscrambling algorithm includes the scrambling user key, plus a device-dependent random value. This random value is not included when the Scrambling/Unscrambling Random Value Disable bit in the Scrambling Mode register (SCRAMBCTRL.RANDOMDIS) is written to '1'.

The random value is neither user-configurable nor readable. If SCRAMBCTRL.RANDOMDIS=0, data scrambled by a given circuit cannot be unscrambled by a different circuit.

If SCRAMBCTRL.RANDOMDIS=1, the scrambling/unscrambling algorithm includes only the scrambling user key, making it possible to manage data by different circuits.

The scrambling user key must be securely stored in a reliable Non-Volatile Memory to recover data from the off-chip memory. Any data scrambled with a given key cannot be recovered if the key is lost.

31.6.10 DMA Operation

The QSPI generates the following DMA requests:

- Data received (RX): The request is set when data is available in the RXDATA register, and cleared when RXDATA is read.
- Data transmit (TX): The request is set when the transmit buffer (TXDATA) is empty, and cleared when TXDATA is written.

Note: If DMA and RX memory modes are selected, a QSPI memory space read operation is required to force the first triggering.

If the CPU accesses the registers which are source of DMA request set/clear condition, the DMA request can be lost or the DMA transfer can be corrupted.

31.6.11 Interrupts

The QSPI has the following interrupt source:

- Interrupt Request (INTREQ): Indicates that at least one bit in the Interrupt Flag Status and Clear register (INTFLAG) is set to '1'.

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 '1' to the corresponding bit in the Interrupt Enable Set (INTENSET) register, and disabled by writing a '1' to the corresponding bit in the Interrupt Enable Clear (INTENCLR) register. An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled, or the QSPI is reset. All interrupt requests from the peripheral are ORed together on system level to generate one combined interrupt request to the NVIC. The user must read the INTFLAG register to determine which interrupt condition is present.

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

31.7 Register Summary

See *QSPI* module in the *Product Memory Mapping Overview* from Related Links for base address.

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x00	CTRLA	7:0							ENABLE	SWRST
		15:8								
		23:16								
		31:24								LASTXFER
0x04	CTRLB	7:0			CSMODE[1:0]		SMEMREG	WDRBT	LOOPEN	MODE
		15:8					DATALEN[3:0]			
		23:16	DLYBCT[7:0]							
		31:24	DLYCS[7:0]							
0x08	BAUD	7:0							CPHA	CPOL
		15:8	BAUD[7:0]							
		23:16	DLYBS[7:0]							
		31:24								
0x0C	RXDATA	7:0	DATA[7:0]							
		15:8	DATA[15:8]							
		23:16								
		31:24								
0x10	TXDATA	7:0	DATA[7:0]							
		15:8	DATA[15:8]							
		23:16								
		31:24								
0x14	INTENCLR	7:0					ERROR	TXC	DRE	RXC
		15:8						INSTREND		CSRISE
		23:16								
		31:24								
0x18	INTENSET	7:0					ERROR	TXC	DRE	RXC
		15:8						INSTREND		CSRISE
		23:16								
		31:24								
0x1C	INTFLAG	7:0					ERROR	TXC	DRE	RXC
		15:8						INSTREND		CSRISE
		23:16								
		31:24								
0x20	STATUS	7:0							ENABLE	
		15:8							CSSTATUS	
		23:16								
		31:24								
0x24 ... 0x2F	Reserved									
0x30	INSTRADDR	7:0	ADDR[7:0]							
		15:8	ADDR[15:8]							
		23:16	ADDR[23:16]							
		31:24	ADDR[31:24]							
0x34	INSTRCTRL	7:0	INSTR[7:0]							
		15:8								
		23:16	OPTCODE[7:0]							
		31:24								
0x38	INSTRFRAME	7:0	DATAEN	OPTCODEEN	ADDREN	INSTREN		WIDTH[2:0]		
		15:8	DDREN	CRMODE	TFRTYPE[1:0]			ADDRLEN	OPTCODELEN[1:0]	
		23:16				DUMMYLEN[4:0]				
		31:24								
0x3C ... 0x3F	Reserved									

.....continued

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x40	SCRAMBCTRL	7:0							RANDOMDIS	ENABLE
		15:8								
		23:16								
		31:24								
0x44	SCRAMBKEY	7:0	KEY[7:0]							
		15:8	KEY[15:8]							
		23:16	KEY[23:16]							
		31:24	KEY[31:24]							

Related Links

7. Product Memory Mapping Overview

31.8 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.

Optional write protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write Protection" property in each individual register description.

See *Peripheral Access Controller (PAC)* from Related Links.

Some registers are enable-protected, meaning they can only be written when the QSPI is disabled. Enable-protection is denoted by the Enable-protected property in each individual register description.

Related Links

20. Peripheral Access Controller (PAC)

31.8.1 Control A

Name: CTRLA
Offset: 0x00
Reset: 0x00000000
Property: -

Control A

Bit	31	30	29	28	27	26	25	24
								LASTXFER
Access								W
Reset								0

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
							ENABLE	SWRST
Access							W	W
Reset							0	0

Bit 24 – LASTXFER Last Transfer
0: No effect.
1: The chip select will be de-asserted after the character written in TD has been transferred.

Bit 1 – ENABLE Enable
Writing a '0' to this bit disables the QSPI.
Writing a '1' to this bit enables the QSPI to transfer and receive data.
As soon as ENABLE is reset, QSPI finishes its transfer.
All pins are set in input mode and no data is received or transmitted.
If a transfer is in progress, the transfer is finished before the QSPI is disable.

Bit 0 – SWRST Software Reset
Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets the QSPI. A software-triggered hardware reset of the QSPI interface is performed.
DMAC channels are not affected by software reset.

31.8.2 Control B

Name: CTRLB
Offset: 0x04
Reset: 0x00000000
Property: PAC Write-Protection

Control B

Bit	31	30	29	28	27	26	25	24
	DLYCS[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	23	22	21	20	19	18	17	16
	DLYBCT[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
					DATALEN[3:0]			
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
			CSMODE[1:0]		SMEMREG	WDRBT	LOOPEN	MODE
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bits 31:24 – DLYCS[7:0] Minimum Inactive CS Delay
This bit field defines the minimum delay between the inactivation and the activation of CS. The DLYCS time guarantees the client minimum deselect time.
If DLYCS is 0x00, one CLK_QSPI_AHB period will be inserted by default.
Otherwise, the following equation determines the delay:
DLYCS = Minimum inactive × fperipheral clock

Bits 23:16 – DLYBCT[7:0] Delay Between Consecutive Transfers
This field defines the delay between two consecutive transfers with the same peripheral without removing the chip select. The delay is always inserted after each transfer and before removing the chip select if needed.
When DLYBCT=0x00, no delay between consecutive transfers is inserted and the clock keeps its duty cycle over the character transfers. In Serial Memory mode (MODE=1), DLYBCT is ignored and no delay is inserted. Otherwise, the following equation determines the delay:
DLYBCT = (Delay Between Consecutive Transfers × fperipheral clock) / 32

Bits 11:8 – DATALEN[3:0] Data Length
The DATALEN field determines the number of data bits transferred. Reserved values must not be used.

Value	Name	Description
0x0	8BITS	8-bits transfer
0x1	9BITS	9-bits transfer
0x2	10BITS	10-bits transfer
0x3	11BITS	11-bits transfer
0x4	12BITS	12-bits transfer
0x5	13BITS	13-bits transfer
0x6	14BITS	14-bits transfer
0x7	15BITS	15-bits transfer
0x8	16BITS	16-bits transfer
0x9-0xF		Reserved

Bits 5:4 – CSMODE[1:0] Chip Select Mode

The CSMODE field determines how the chip select is de-asserted.

Value	Name	Description
0x0	NORELOAD	The chip select is de-asserted if TD has not been reloaded before the end of the current transfer.
0x1	LASTXFER	The chip select is de-asserted when the bit LASTXFER is written at 1 and the character written in TD has been transferred.
0x2	SYSTEMATICALLY	The chip select is de-asserted systematically after each transfer.
0x3		Reserved

Bit 3 – SMEMREG Serial Memory Register Mode

Value	Description
0	Serial memory registers are written via AHB access.
1	Serial memory registers are written via APB access. Reset the QSPI.

Bit 2 – WDRBT Wait Data Read Before Transfer

This bit determines the Wait Data Read Before Transfer option.

Bit 1 – LOOPEN Local Loopback Enable

This bit defines if the Local Loopback is enabled or disabled.

LOOPEN controls the local loopback on the data serializer for testing in SPI Mode only. (MISO is internally connected on MOSI).

Value	Description
0	Local Loopback is disabled.
1	Local Loopback is enabled.

Bit 0 – MODE Serial Memory Mode

This bit defines if the QSPI is in SPI Mode or Serial Memory Mode.

Value	Name	Description
0	SPI	SPI operating mode
1	MEMORY	Serial Memory operating mode

31.8.3 Baud Rate

Name: BAUD
Offset: 0x08
Reset: 0x00000000
Property: PAC Write-Protection

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	DLYBS[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
	BAUD[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	7	6	5	4	3	2	1	0
							CPHA	CPOL
Access							R/W	R/W
Reset							0	0

Bits 23:16 – DLYBS[7:0] Delay Before SCK

This field defines the delay from CS valid to the first valid SCK transition.
When DLYBS equals zero, the CS valid to SCK transition is 1/2 the SCK clock period.
Otherwise, the following equation determines the delay:

Equation 31-1. Delay Before SCK

$$Delay\ \ Before\ SCK = \frac{DLYBS}{MCK}$$

Bits 15:8 – BAUD[7:0] Serial Clock Baud Rate

The QSPI uses a modulus counter to derive the SCK baud rate from the module clock (MCK) CLK_QSPI_AHB. The Baud rate is selected by writing a value from 1 to 255 in the BAUD field. The following equation determines the SCK baud rate:

Equation 31-2. SCK Baud Rate

$$SCK\ \ Baud\ \ Rate = \frac{MCK}{(BAUD + 1)}$$

Bit 1 – CPHA Clock Phase

CPHA determines which edge of SCK causes data to change and which edge causes data to be captured. CPHA is used with CPOL to produce the required clock/data relationship between host and client devices.

Value	Description
0	Data is captured on the leading edge of SCK and changed on the following edge of SCK.
1	Data is changed on the leading edge of SCK and captured on the following edge of SCK.

Bit 0 – CPOL Clock Polarity

CPOL is used to determine the inactive state value of the serial clock (SCK). It is used with CPHA to produce the required clock/data relationship between host and client devices.

Value	Description
0	The inactive state value of SCK is logic level zero.

Value	Description
0	The inactive state value of SCK is logic level 'one'.

31.8.4 Receive Data

Name: RXDATA
Offset: 0x0C
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
	DATA[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
	DATA[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 15:0 – DATA[15:0] Receive Data
Data received by the QSPI is stored in this register right-justified. Unused bits read zero.

31.8.5 Transmit Data

Name: TXDATA
Offset: 0x10
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
	DATA[15:8]							
Access	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	DATA[7:0]							
Access	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bits 15:0 – DATA[15:0] Transmit Data
Data to be transmitted by the QSPI is stored in this register. Information to be transmitted must be written to the transmit data register in a right-justified format.

31.8.6 Interrupt Enable Clear

Name: INTENCLR
Offset: 0x14
Reset: 0x00000000
Property: PAC Write-Protection

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
						INSTREND		CSRISE
Access						R/W		R/W
Reset						0		0

Bit	7	6	5	4	3	2	1	0
					ERROR	TXC	DRE	RXC
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit 10 – INSTREND Instruction End Interrupt Disable
Writing a '0' to this bit has no effect.
Writing a '1' will clear the corresponding interrupt request.

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

Bit 8 – CSRISE Chip Select Rise Interrupt Disable
Writing a '0' to this bit has no effect.
Writing a '1' will clear the corresponding interrupt request.

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

Bit 3 – ERROR Overrun Error Interrupt Disable
Writing a '0' to this bit has no effect.
Writing a '1' will clear the corresponding interrupt request.

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

Bit 2 – TXC Transmission Complete Interrupt Disable
Writing a '0' to this bit has no effect.
Writing a '1' will clear the corresponding interrupt request.

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

Bit 1 – DRE Transmit Data Register Empty Interrupt Disable

Writing a '0' to this bit has no effect.
Writing a '1' will clear the corresponding interrupt request.

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

Bit 0 – RXC Receive Data Register Full Interrupt Disable

Writing a '0' to this bit has no effect.
Writing a '1' will clear the corresponding interrupt request.

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

31.8.7 Interrupt Enable Set

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

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
						INSTREND		CSRISE
Access						R/W		R/W
Reset						0		0

Bit	7	6	5	4	3	2	1	0
					ERROR	TXC	DRE	RXC
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit 10 – INSTREND Instruction End Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' will set the corresponding interrupt request.

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

Bit 8 – CSRISE Chip Select Rise Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' will set the corresponding interrupt request.

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

Bit 3 – ERROR Overrun Error Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' will set the corresponding interrupt request.

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

Bit 2 – TXC Transmission Complete Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' will set the corresponding interrupt request.

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

Bit 1 – DRE Transmit Data Register Empty Interrupt Enable

Writing a '0' to this bit has no effect.
Writing a '1' will set the corresponding interrupt request.

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

Bit 0 – RXC Receive Data Register Full Interrupt Enable

Writing a '0' to this bit has no effect.
Writing a '1' will set the corresponding interrupt request.

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

31.8.8 Interrupt Flag Status and Clear

Name: INTFLAG
Offset: 0x1C
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
						INSTREND		CSRISE
Access						R/W		R/W
Reset						0		0

Bit	7	6	5	4	3	2	1	0
					ERROR	TXC	DRE	RXC
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit 10 – INSTREND Instruction End
This bit is set when an Instruction End has been detected.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the flag.

Bit 8 – CSRISE Chip Select Rise
The bit is set when a Chip Select Rise has been detected.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the flag.

Bit 3 – ERROR Overrun Error
This bit is set when an ERROR has occurred.
An ERROR occurs when RXDATA is loaded at least twice from the serializer.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the flag.

Bit 2 – TXC Transmission Complete
0: As soon as data is written in TXDATA.
1: TXDATA and internal shifter are empty. If a transfer delay has been defined, TXC is set after the completion of such delay.

Bit 1 – DRE Transmit Data Register Empty
0: Data has been written to TXDATA and not yet transferred to the serializer.
1: The last data written in the TXDATA has been transferred to the serializer.
This bit is '0' when the QSPI is disabled or at reset.
The bit is set as soon as ENABLE bit is set.

Bit 0 – RXC Receive Data Register Full
0: No data has been received since the last read of RXDATA.

1: Data has been received and the received data has been transferred from the serializer to RXDATA since the last read of RXDATA.

31.8.9 Status

Name: STATUS
Offset: 0x20
Reset: 0x00000200
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
							CSSTATUS	
Access							R	
Reset							1	

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

Bit 9 – CSSTATUS Chip Select

Value	Description
0	Chip Select is asserted.
1	Chip Select is not asserted.

Bit 1 – ENABLE Enable

Value	Description
0	QSPI is disabled.
1	QSPI is enabled.

31.8.10 Instruction Address

Name: INSTRADDR
Offset: 0x30
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
	ADDR[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
	ADDR[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
	ADDR[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
	ADDR[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – ADDR[31:0] Instruction Address
Address to send to the serial flash memory in the instruction frame.

31.8.11 Instruction Code

Name: INSTRCTRL
Offset: 0x34
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

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

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

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

Bits 23:16 – OPTCODE[7:0] Option Code
These bits define the option code to send to the serial flash memory.

Bits 7:0 – INSTR[7:0] Instruction Code
Instruction code to send to the serial flash memory.

31.8.12 Instruction Frame

Name: INSTRFRAME
Offset: 0x38
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
				DUMMYLEN[4:0]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	DDREN	CRMODE	TFRTYPE[1:0]			ADDRLEN	OPTCODELEN[1:0]	
Access	R/W	R/W	R/W	R/W		R/W	R/W	R/W
Reset	0	0	0	0		0	0	0

Bit	7	6	5	4	3	2	1	0
	DATAEN	OPTCODEEN	ADDREN	INSTREN		WIDTH[2:0]		
Access	R/W	R/W	R/W	R/W		R/W	R/W	R/W
Reset	0	0	0	0		0	0	0

Bits 20:16 – DUMMYLEN[4:0] Dummy Cycles Length
The DUMMYLEN field defines the number of dummy cycles required by the serial Flash memory before data transfer.

Bit 15 – DDREN Double Data Rate Enable
Note: Double Data Rate operating is only supported in Read.

Value	Description
0	Double Data Rate operating mode is disabled.
1	Double Data Rate operating mode is enabled.

Bit 14 – CRMODE Continuous Read Mode
This bit defines if the Continuous Read Mode is enabled or disabled.

Value	Description
0	Continuous Read Mode is disabled.
1	Continuous Read Mode is enabled.

Bits 13:12 – TFRTYPE[1:0] Data Transfer Type
These bits define the data type transfer.

Value	Name	Description
0x0	READ	Read transfer from the serial memory.Scrambling is not performed.Read at random location (fetch) in the serial flash memory is not possible.
0x1	READMEMORY	Read data transfer from the serial memory.If enabled, scrambling is performed.Read at random location (fetch) in the serial flash memory is possible.
0x2	WRITE	Write transfer into the serial memory.Scrambling is not performed.
0x3	WRITEMEMORY	Write data transfer into the serial memory. If enabled, scrambling is performed.

Bit 10 – ADDRLEN Address Length
The ADDRLEN bit determines the length of the address.

Value	Name	Description
0x0	24BITS	24-bits address length
0x1	32BITS	32-bits address length

Bits 9:8 – OPTCODELEN[1:0] Option Code Length

The OPTCODELEN field determines the length of the option code. The value written in OPTCODELEN must be coherent with value written in the field WIDTH. For example: OPTCODELEN=0 (1-bit option code) is not coherent with WIDTH=6 (option code sent with QuadSPI protocol, thus the minimum length of the option code is 4-bit).

Value	Name	Description
0x0	1BIT	1-bit length option code
0x1	2BITS	2-bits length option code
0x2	4BITS	4-bits length option code
0x3	8BITS	8-bits length option code

Bit 7 – DATAEN Data Enable

Value	Description
0	No data is sent/received to/from the serial flash memory.
1	Data is sent/received to/from the serial flash memory.

Bit 6 – OPTCODEEN Option Enable

Value	Description
0	The option is not sent to the serial flash memory
1	The option is sent to the serial flash memory.

Bit 5 – ADDREN Address Enable

Value	Description
0	The transfer address is not sent to the serial flash memory.
1	The transfer address is sent to the serial flash memory.

Bit 4 – INSTREN Instruction Enable

Value	Description
0	The instruction is not sent to the serial flash memory.
1	The instruction is sent to the serial flash memory.

Bits 2:0 – WIDTH[2:0] Instruction Code, Address, Option Code and Data Width

This field defines the width of the instruction code, the address, the option and the data.

Value	Name	Description
0x0	SINGLE_BIT_SPI	Instruction: Single-bit SPI / Address-Option: Single-bit SPI / Data: Single-bit SPI
0x1	DUAL_OUTPUT	Instruction: Single-bit SPI / Address-Option: Single-bit SPI / Data: Dual SPI
0x2	QUAD_OUTPUT	Instruction: Single-bit SPI / Address-Option: Single-bit SPI / Data: Quad SPI
0x3	DUAL_IO	Instruction: Single-bit SPI / Address-Option: Dual SPI / Data: Dual SPI
0x4	QUAD_IO	Instruction: Single-bit SPI / Address-Option: Quad SPI / Data: Quad SPI
0x5	DUAL_CMD	Instruction: Dual SPI / Address-Option: Dual SPI / Data: Dual SPI
0x6	QUAD_CMD	Instruction: Quad SPI / Address-Option: Quad SPI / Data: Quad SPI
0x7		Reserved

31.8.13 Scrambling Mode

Name: SCRAMBCTRL
Offset: 0x40
Reset: 0x00000000
Property: PAC Write-Protection

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
							RANDOMDIS	ENABLE
Access							R/W	R/W
Reset							0	0

Bit 1 – RANDOMDIS Scrambling/Unscrambling Random Value Disable

Value	Description
0	The scrambling/unscrambling algorithm includes the scrambling user key plus a random value that may differ from chip to chip.
1	The scrambling/unscrambling algorithm includes only the scrambling user key.

Bit 0 – ENABLE Scrambling/Unscrambling Enable

This bit defines if the scrambling/unscrambling is enabled or disabled.

Value	Description
0	Scrambling/unscrambling is disabled.
1	Scrambling/unscrambling is enabled.

31.8.14 Scrambling Key

Name: SCRAMBKEY
Offset: 0x44
Reset: 0x00000000
Property: PAC Write-Protection

Bit	31	30	29	28	27	26	25	24
	KEY[31:24]							
Access	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	KEY[23:16]							
Access	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	KEY[15:8]							
Access	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	KEY[7:0]							
Access	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – KEY[31:0] Scrambling User Key
This field defines the user key value.

32. Analog-to-Digital Converter (ADC)

32.1 Overview

The PIC32CX-BZ3 has one shared ADC module. This ADC module incorporates a multiplexer on the input to facilitate a group of inputs and provides a flexible automated scanning option through the input scan logic.

For the ADC module, the analog inputs are connected to the Sample and Hold (S&H) capacitor. The ADC module performs the conversion of the input analog signal based on the configurations set in the registers. When the conversion is complete, the final result is stored in the result buffer for the specific analog input and is passed to the digital filter and digital comparator if configured to use data from this particular sample.

32.2 Features

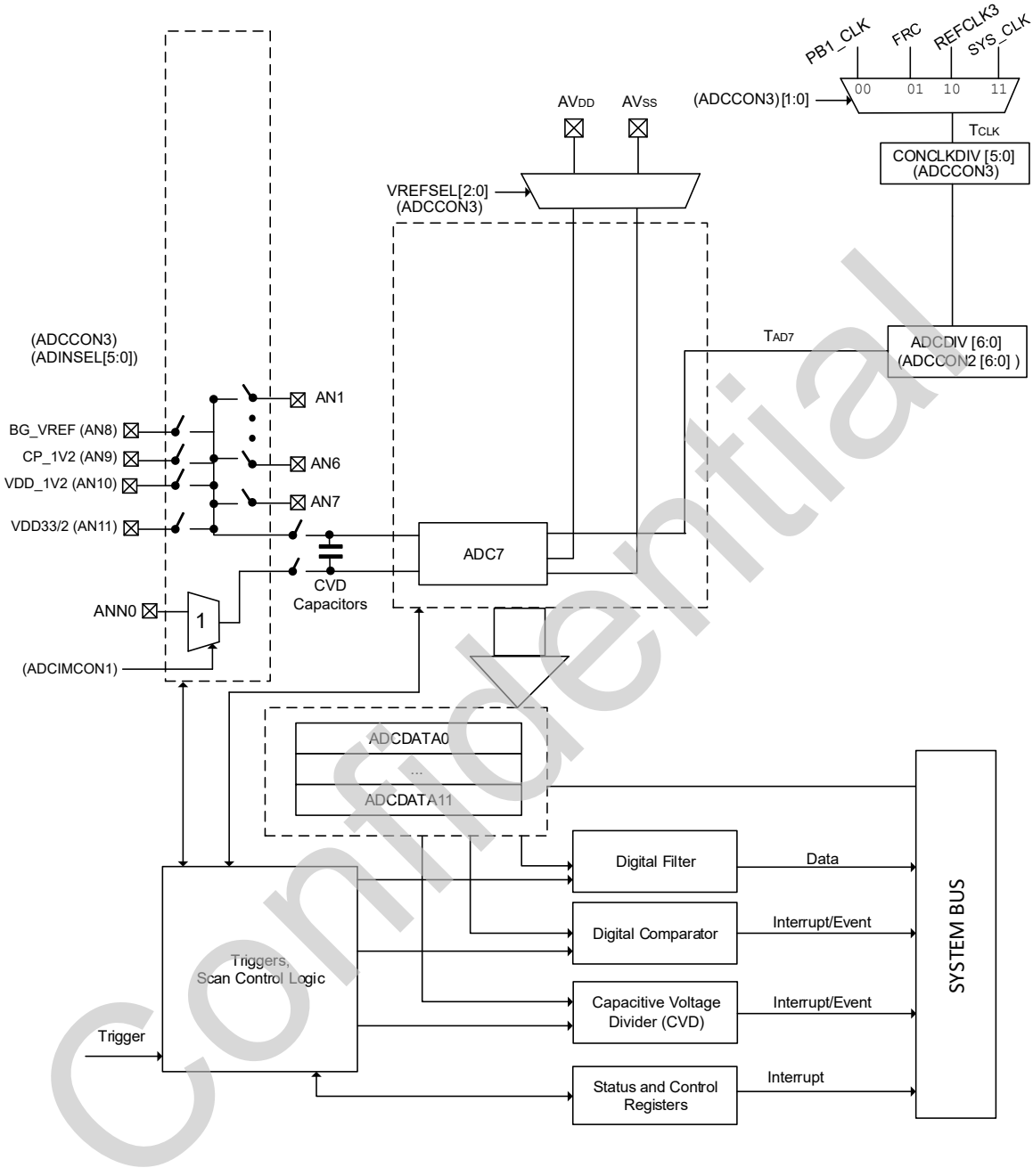
The PIC32CX-BZ3 12-bit High Speed Successive Approximation Register (SAR) Analog-to-Digital Converter (ADC) includes the following features:

- 12-bit resolution
- One ADC module, up to 2 Msps conversion rate
- Single-ended and/or differential input
- Supported in Standby Sleep mode
- Two digital comparators
- Two digital filters supporting two modes:
  - Oversampling mode
  - Averaging mode
- Designed for motor control, power conversion and general purpose applications

32.3 Block Diagram

A block diagram of the ADC module is illustrated in the following figure.

Figure 32-1. ADC Block Diagram



32.4 ADC Operation

The High Speed Successive Approximation Register (SAR) ADC is designed to support power conversion and motor control applications and consists of one shared ADC module. The shared ADC module has multiple analog inputs connected to its S&H circuit through a multiplexer. Multiple analog inputs share this ADC; therefore, it is termed the shared ADC module. The shared ADC module is used to measure analog signals of lower frequencies and signals that are static in nature (in other words, do not change significantly with time). However, this ADC module is capable of up to 2 Msps sample rate.

The analog inputs connected to the shared ADC module are Class 2 and Class 3 inputs. The number of inputs designated for each class depends on the specific device. For the PIC32CX-BZ3, the following arrangement is provided.

- Class 2 = AN0 to AN7
- Class 3 = AN8 to AN11

The property of each class of analog input is described in the following table.

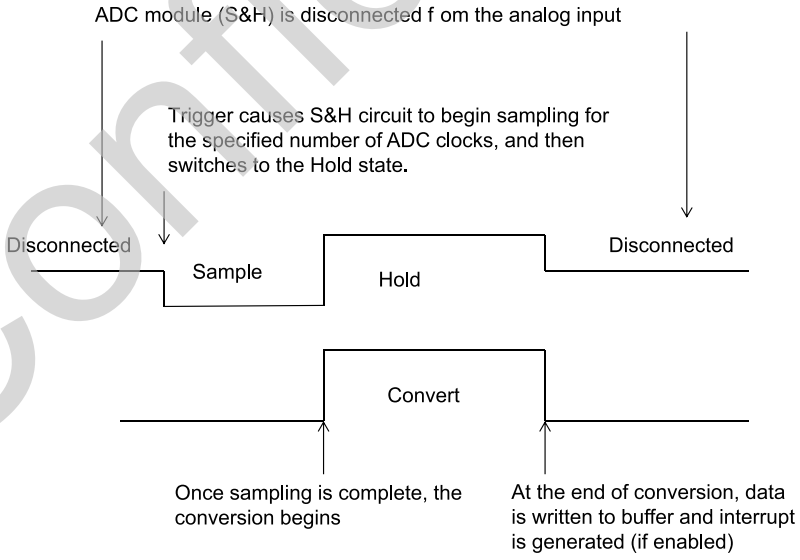
Table 32-1. Analog Input Class

ADC Module	Analog Input Class	Trigger	Trigger Action
Shared ADC module	Class 2	Individual trigger source or scan trigger	Starts sampling sequence or begins scan sequence
Shared ADC module with input scan	Class 3	Scan trigger	Starts scan sequence

Class 2 and Class 3 analog input properties:

- Class 2 inputs are used on the shared ADC module, either individually triggered or as part of a scan list. When used individually, they are triggered by their unique trigger selected by the ADCTRGx register.
- The analog inputs on the shared ADC have a natural order of priority (for example, AN6 has a higher priority than AN7).
- Class 3 inputs are used exclusively for scanning and share a common trigger source (scan trigger).
- Class 3 analog inputs share both the ADC module and the trigger source; therefore, the only method possible to convert them is to scan them sequentially for each incoming scan trigger event, where scanning occurs in the natural order of priority.
- The arrival of a trigger in the shared ADC module only starts the sampling. When the trigger arrives, the ADC module goes into sampling mode for the sampling time decided by the SAMC[9:0] bits (ADCCON2[25:16]). At the end of sampling, the ADC starts conversion. Upon completion of conversion, the ADC module is used to convert the next in line Class 2 or Class 3 inputs according to the natural order of priority. When a shared analog input (Class 2 or Class 3) has completed all conversion and no trigger is pending, the ADC module is disconnected from all analog inputs

Figure 32-2. Sample and Conversion Sequence for Shared ADC Modules

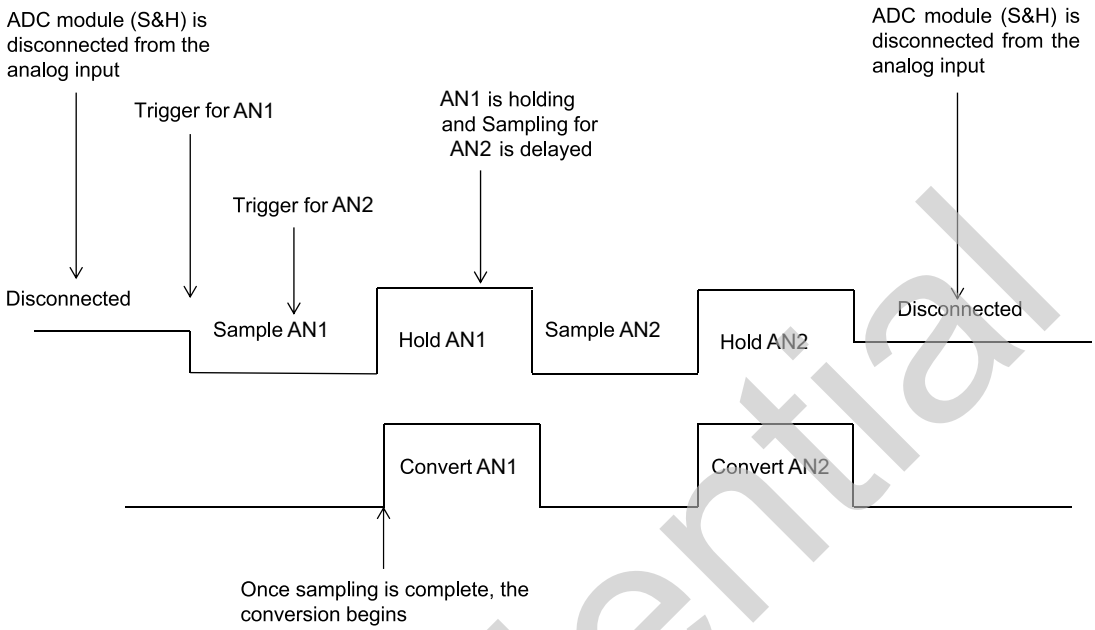


32.4.1 Class 2 Triggering

When a single Class 2 input is triggered, it is sampled and converted by the shared S&H using the sequence illustrated in Sample and Conversion Sequence for the Shared ADC Modules figure; see *Sample and Conversion Sequence for Shared ADC Modules* figure in the *ADC Operation* from Related Links. When multiple Class 2 inputs are triggered, it is important to understand the consequences of trigger timing. If a conversion is underway and

another Class 2 trigger occurs, then the sample-hold-conversion for the new trigger is stalled until the in-process, sample-hold cycle is complete, as shown in the following figure.

Figure 32-3. Multiple Independent Class 2 Trigger Conversion Sequence

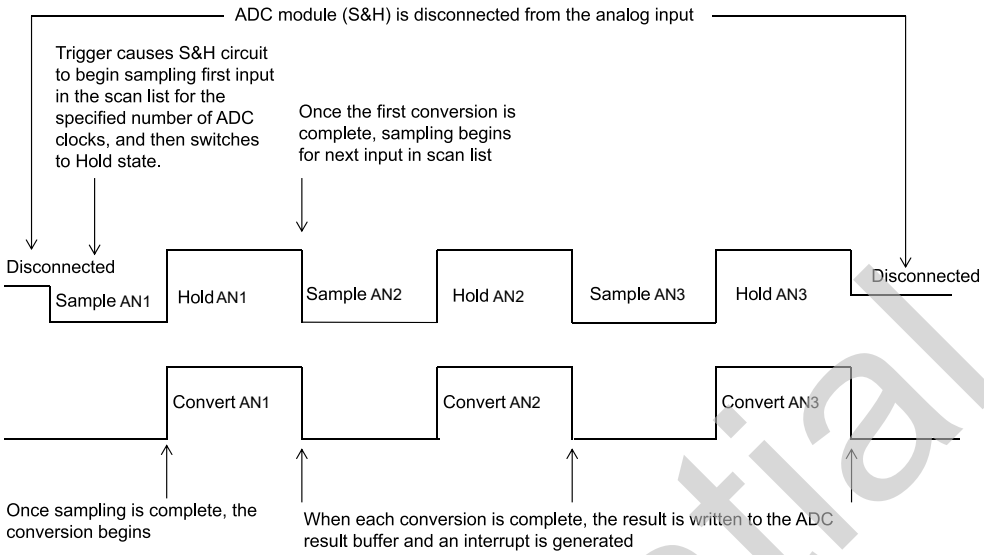


When multiple inputs to the shared S&H are triggered simultaneously, the processing order is determined by their natural priority (the lowest numbered input has the highest priority). As an example, if AN1, AN2 and AN3 are triggered simultaneously, AN1 is sampled and converted first, followed by AN2 and finally, AN3. When using the independent Class 2 triggering on the shared S&H, the SAMC[9:0] bits (ADCCON2[25:16]) determine the sample time for all inputs while the appropriate TRGSRC[4:0] bits in the ADCTRGx Register (see *ADCTRG1* register from Related Links) determine the trigger source for each input.

32.4.2 Input Scan

Input scanning is a feature that allows an automated scanning sequence of multiple Class 2 or Class 3 inputs. All Class 2 and Class 3 inputs are scanned using the single shared S&H. The selection of analog inputs for scanning is done with the CSSx bits of the ADCCSS1 registers. Class 2 inputs are triggered using STRIG selection in the ADCTRGx register, and Class 3 inputs are triggered using the STRGSRC[4:0] of the ADCCON1[20:16] register. When a trigger occurs for Class 2 or Class 3 inputs, the sampling and conversion occur in the natural input order is used; lower number inputs are sampled before higher number inputs.

Figure 32-4. Input Scan Conversion Sequence for Three Class 2 Inputs

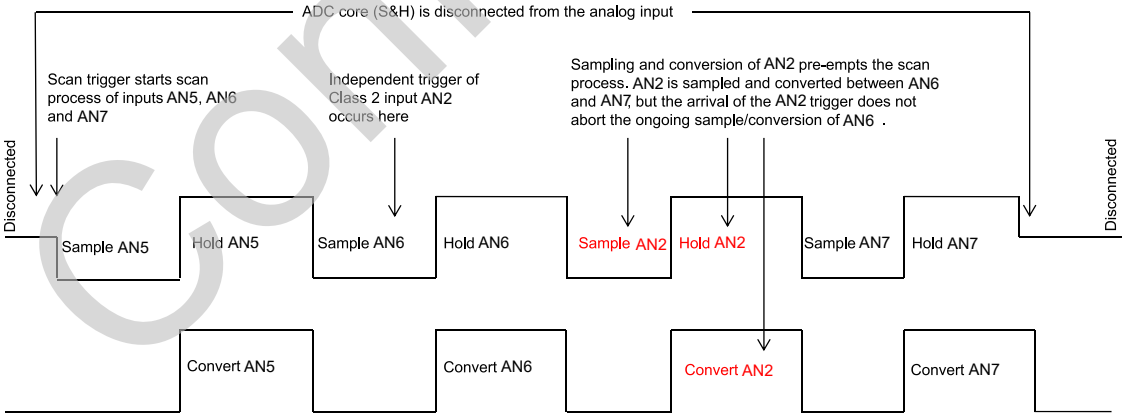


When using the shared analog inputs in scan mode, the SAMC[9:0] bits in the ADC Control Register 2 (ADCCON2[25:16]) determine the sample time for all inputs, while the Scan Trigger Source Selection bits (STRGSRC[4:0]) in the ADC Control Register 1 (ADCCON1[20:16]) determine the trigger source.

To ensure predictable results, a scan must not be retriggered until a sampling of all inputs is complete. Ensure system design to preclude retriggering a scan while a scan is in progress.

Individual Class 2 triggers that occur during a scan preempts the scan sequence if they are a higher priority than the sample currently being processed. In the following figure, a scan of AN5, AN6 and AN7 is underway when an independent trigger of Class 2 input AN2 takes place. The scan is interrupted for the sampling and conversion of AN2.

Figure 32-5. Scan Conversion Pre-empted by Class 2 Input Trigger



32.5 ADC Module Configuration

Operation of the ADC module is directed through bit settings in the specific registers. The following instructions summarize the actions and the settings. The options and details for each configuration step are provided in the subsequent sections.

To configure the ADC module, perform the following steps:

1. Configure the analog port pins as described in
2. Select the analog inputs to the ADC multiplexers as described in
3. Select the format of the ADC result as described in
4. Select the conversion trigger source as described in
5. Select the voltage reference source as described in
6. Select the scanned inputs as described in
7. Select the analog-to-digital conversion clock source and prescaler as described in
8. Specify any additional acquisition time (if required) as described in
9. Turn on the ADC module as described in
10. Poll (or wait for the interrupt) for the voltage reference to be ready as described in
11. Enable the analog and bias circuit for the required ADC modules, and, after the ADC module wakes up, enable the digital circuit as described in
12. Configure the ADC interrupts (if required) as described in

32.5.1 Configuring the Analog Port Pins

The ANSELx registers for the I/O ports associated with the analog inputs are used to configure the corresponding pin as an analog or a digital pin. A pin is configured as an analog input when the corresponding ANSELx bit = '1'. When the ANSELx bit = '0', the pin is set to digital control. The ANSELx registers are set when the device comes out of Reset, causing the ADC input pins to be configured as analog inputs by default.

The TRISx registers control the digital function of the port pins. The port pins that are required as analog inputs must have their corresponding bit set in the specific TRISx register, configuring the pin as an input. If the I/O pin associated with an ADC input is configured as an output by clearing the TRISx bit, the port's digital output level (V_{OH} or V_{OL}) is converted. After a device Reset, all of the TRISx bits are set. For more information on port pin configuration, see *I/O Ports and Peripheral Pin Select (PPS)* from Related Links.

Note: When reading a PORT register that shares pins with the ADC, any pin configured as an analog input reads as '0' when the PORT latch is read. Analog levels on any pin that is defined as a digital input but not configured as an analog input, may cause the input buffer to consume the current that exceeds the device specification.

32.5.2 Selecting the ADC Multiplexer Analog Inputs

The ADC module has two inputs, referred to as the positive and negative inputs. Input selection options vary as described in the following sections.

32.5.2.1 Selection of Positive Inputs

For the shared ADC module, the positive input is shared among all Class 2 and Class 3 inputs. Input connection of the analog input ANx to the shared ADC is automatic for either the Class 2 input trigger or during a scan of Class 2 and or Class 3 inputs. Selecting inputs for scanning is described in *Selecting the Scanned Inputs* from Related Links.

Related Links

32.5.6. Selecting the Scanned Inputs

32.5.2.2 Selection of Negative Inputs

Negative input selection is determined by the setting of the DIFFx bit of the ADCIMCON1 register. The DIFFx bit allows the inputs to be rail-to-rail and either single-ended or differential. The SIGNx and DIFFx bits in the ADCIMCON1 register scale the internal ADC analog inputs and reference voltages and configure the digital result to align with the expected full-scale output range.

For the shared ADC module, the analog inputs have individual settings for the DIFFx bit. Therefore, the user has the ability to select certain inputs as single-ended and others as differential while being connected to the same shared ADC module. While sampling, the signal changes on-the-fly as single-ended or differential according to its corresponding DIFFx bit setting.

Table 32-2. Negative Input Selection

ADCIMCON1		Input Configuration	Input Voltage		Output
DIFFx	SIGNx				
1	1	Differential 2's complement	Minimum input	$V_{IN}P - V_{IN}N = -V_{REF}$	-2048
			Maximum input	$V_{IN}P - V_{IN}N = V_{REF}$	+2047
1	0	Differential unipolar	Minimum input	$V_{IN}P - V_{IN}N = -V_{REF}$	0
			Maximum input	$V_{IN}P - V_{IN}N = V_{REF}$	+4095
0	1	Single-ended 2's complement	Minimum input	$V_{IN}P = V_{REF}$	-2048
			Maximum input	$V_{IN}P - V_{IN}N = V_{REF}$	+2047
0	0	Single-ended unipolar	Minimum input	$V_{IN}P = V_{REF}$	0
			Maximum input	$V_{IN}P - V_{IN}N = V_{REF}$	+4095

Legend:

- $V_{IN}P$ = Positive S&H input
- $V_{IN}N$ = Negative S&H input
- $V_{REF} = V_{REFH} - V_{REFL}$

Note: For proper operation and to prevent device damage, input voltage levels must not exceed the limits listed in the Electrical Specifications.

32.5.3 Selecting the Format of the ADC Result

The data in the ADC Result register can be read in any of the four supported data formats. The user can select from unsigned integer, signed integer, unsigned fractional or signed fractional. Integer data is right-justified and fractional data is left-justified.

- The integer or fractional data format selection is specified globally for all analog inputs using the Fractional Data Output Format bit, FRACT (ADCCON1[23]).
- The signed or unsigned data format selection can be independently specified for each individual analog input using the SIGNx bits in the ADCIMCONx registers
  .

The following table provides how a result is formatted.

Table 32-3. ADC Result Format

FRACT	SIGNx	Description	32-bit Output Data Format			
0	0	Unsigned integer	0000	0000	0000	0000
			0000	dddd	dddd	dddd
0	1	Signed integer	ssss	ssss	ssss	ssss
			ssss	sddd	dddd	dddd
1	0	Fractional	dddd	dddd	dddd	0000
			0000	0000	0000	0000
1	1	Signed fractional	sddd	dddd	dddd	dddd
			0000	0000	0000	0000

The following code is an example for ADC Class 2 configuration and fractional format.

```
int main(int argc, char** argv) {
int result[3];

/* Configure ADCCON1 */
ADCCON1bits.FRACT = 1;    // use Fractional output format ADCCON1bits.SELRES = 3;   // ADC
resolution is 12 bits ADCCON1bits.STRGSRC = 0;  // No scan trigger.

/* Configure ADCCON2 */
ADCCON2bits.SAMC = 5;         // ADC sampling time = 5 * TAD7
ADCCON2bits.ADCDIV = 1;       // ADC clock freq is half of control clock = TAD7

/* Initialize warm up time register */ ADCANCON = 0;
ADCANCONbits.WKUPCLKCNT = 5;  // Wakeup exponent = 32 * TAD7

/* Clock setting */ ADCCON3 = 0;
ADCCON3bits.ADCSEL = 0;       // Select input clock source
ADCCON3bits.CONCLKDIV = 1;    // Control clock frequency is half of input clock
ADCCON3bits.VREFSEL = 0;      // Select AVDD and AVSS as reference source

/* Select ADC input mode */
ADCIMCON1bits.SIGN7 = 0;  // unsigned data format ADCIMCON1bits.DIFF7 = 0;

/* Configure ADCGIRQENx */
ADCGIRQEN1 = 0;           // No interrupts are used
ADCGIRQEN2 = 0;

/* Configure ADCCSSx */
ADCCSS1 = 0;              // No scanning is used

/* Configure ADCCMPCONx */
ADCCMPCON1 = 0; // No digital comparators are used. Setting the ADCCMPCONx
ADCCMPCON2 = 0; // register to '0' ensures that the comparator is disabled.

/* Configure ADCFLTRx */
ADCFLTR1 = 0;             // No oversampling filters are used. ADCFLTR2 = 0;

/* Set up the trigger sources */
ADCTRGSNSbits.LVL7 = 0;
ADC1TRG2bits.TRGSRC7 = 1; // Set AN7 to trigger from software

/* Turn the ADC on */
ADCCON1bits.ON = 1;

/* Wait for voltage reference to be stable */
while(!ADCCON2bits.BGVRRDY); // Wait until the reference voltage is ready
while(ADCCON2bits.REFFLT);   // Wait if there is a fault with the reference voltage

/* Enable clock to analog circuit */
ADCANCONbits.ANEN7 = 1;       // Enable the clock to analog bias

/* Wait for ADC to be ready */
while(!ADCANCONbits.WKRDY7); // Wait until ADC7 is ready

/* Enable the ADC module */ ADCCON3bits.DIGEN7 = 1;       // Enable ADC7

while (1) {
/* Trigger a conversion */ ADCCON3bits.GSWTRG = 1;

/* Wait the conversions to complete */
while (ADCDSTAT1bits.ARDY7 == 0);
/* fetch the result */
result[0] = ADCDATA7;

/*
* Process results here
*
* Note 1: Loop time determines the sampling time since all inputs are Class 2.
* If the loop time happens is small and the next trigger happens before the
* completion of set sample time, the conversion will happen only after the
* sample time has elapsed.
*
* Note 2: Results are in fractional format
*
```

```
*/
}
return (1);
}
```

32.5.4 Selecting the Conversion Trigger Source

Class 2 inputs to the ADC module can be triggered for conversion either individually or as part of a scan sequence. Class 3 inputs can only be triggered as part of a scan sequence. Individual or scan triggers can originate from an event system (EVSYS), from external digital circuits connected to INT0, from external analog circuits connected to an analog comparator or through software by setting a trigger bit in an SFR.

Note: When conversion triggers for multiple Class 2 analog inputs occur simultaneously, they are prioritized according to a natural order priority scheme based on the analog input used. AN6 has the highest priority, AN7 has the next highest priority and so on.

32.5.4.1 Trigger Selection Class 2 Inputs

For each one of the Class 2 inputs, the user application can independently specify a conversion trigger source. The individual trigger source for an analog input 'x' is specified by the TRGSRC[4:0] bits located in registers ADCTRG1 through ADCTRG2. For example, these trigger sources may include:

- **Event System (EVSYS)**: The event system is tied with many peripherals. The peripheral can be an event generator and ADC can be an event user to trigger the ADC conversion. See *Event User m* from Related Links.
- **External INT0 Pin Trigger**: In this mode, the ADC module starts a conversion on an active transition on the INT0 pin. The INT0 pin may be programmed for either a rising edge input or a falling edge input to trigger the conversion process.
- **Global Software Trigger**: The ADC module can be configured for manually triggering a conversion for all inputs that have selected this trigger option. The user can manually trigger a conversion by setting the Global Software Trigger bit, GSWTRG (ADCCON3[6]).

Related Links

26.8.13. USERm

32.5.4.2 Conversion Trigger Sources and Control

The following are the possible sources for each trigger signal:

- External trigger selection through the TRGSRCx[4:0] bits in the ADCTRGx registers. This capability is supported only for Class 2 analog inputs. Typically, the user specifies a particular trigger source to initiate a conversion for specific input. All of the analog inputs may select the same trigger source if desired. In such an event, the result resembles a "scanned conversion", which has its order of completion enforced by the priority of the inputs associated with the same trigger source. The first trigger selection is `00000` (no trigger), which amounts to temporarily disabling that particular trigger and, consequently, temporarily disabling that analog input from being converted. The next two selections for trigger source (GSWTRG and GLSWTRG) are software-generated trigger sources. The second software-generated trigger selection is the Global Software Trigger (GSWTRG). This trigger links to the GSWTRG bit in the ADCCON3 register, which may be used to enable the user application to initiate a single conversion. GSWTRG is a self-clearing bit; therefore, it clears itself on the next ADC clock cycle after being set by the user application. The third software-generated trigger selection is the Global Level Software Trigger (GLSWTRG), which is linked to the GLSWTRG bit in the ADCCON3 register. This trigger may be used by the user application to initiate a burst of consecutive samples as the GLSWTRG bit is not self-clearing. The fourth trigger selection is a special selection, the Scan Trigger selection, which allows the Class 2 analog inputs to be included as members of a global scan of all inputs.
- Scanned trigger selection via the STRGSRC[4:0] bits in the ADCCON1 register and select bits in the ADCCSS1 registers. This mode is typically used to initiate the conversion of a group of analog inputs. This capability works for 2 and 3 analog inputs but is typically used for Class 3 inputs because they do not have individual associated TRGSRC bits. One of the trigger selections is the GSWTRG bit in the ADCCON3 register, which may be used to enable the user software to initiate a conversion.
- User initiated trigger via the ADINSEL[5:0] bits and the RQCNVRT bit in the ADCCON3 register. This mode enables the user application to create an individual conversion trigger request for a specified analog input. Using this mode enables the user application to trigger the conversion of an input without changing the trigger source configuration of the ADC. This is useful in handling error situations where another software module wants ADC

information without disrupting the normal operation of the ADC. This is also the preferred method to generate the initial trigger to start a digital filter sequence.

- User-controlled sampling of Class 2 and Class 3 inputs via the ADINSEL[5:0] bits and the SAMP bit in the ADCCON3 register. Setting the SAMP bit causes the Class 2 and Class 3 inputs to be in Sampling mode while ignoring the selection of the SAMC[9:0] bits. This mode is also useful in software conversion of ADC with software-selectable sample time.

32.5.4.3 User-Requested Individual Conversion Trigger (Software ADC Conversion)

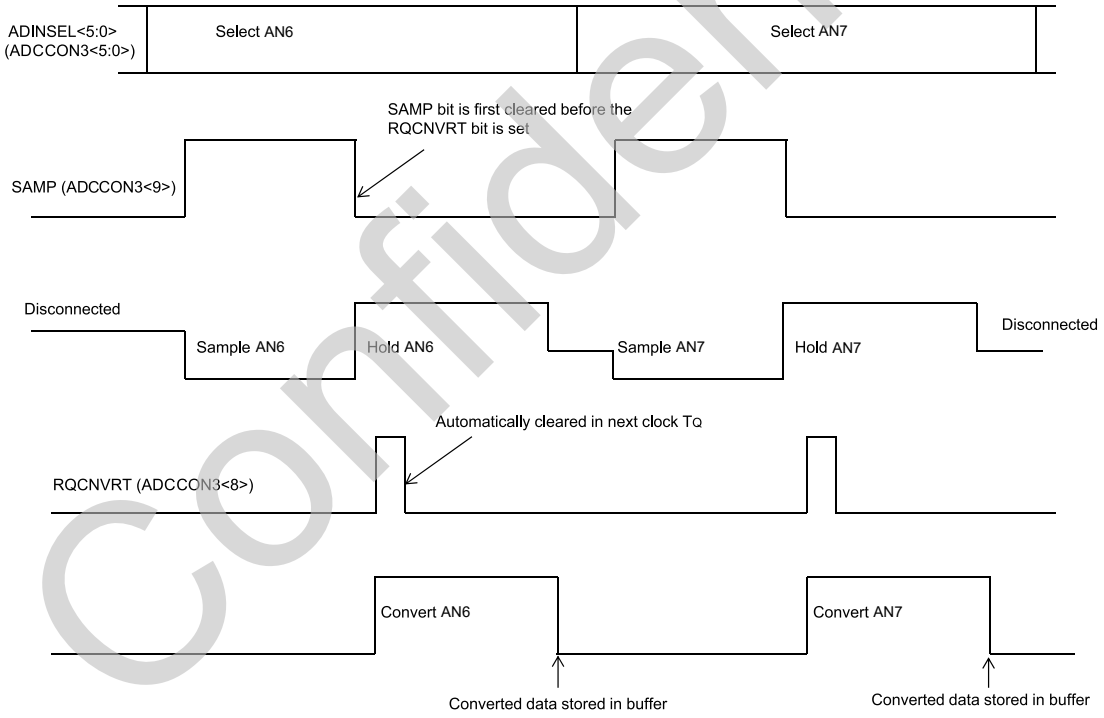
The user can explicitly request a single conversion (by software) of any selected analog input at any time during program execution without changing the trigger source configuration of the ADC.

The steps to be followed for conversion are as follows:

1. The analog input ID to be converted is specified by the ADC Input Select bits, ADINSEL[5:0] (ADCCON3[5:0]).
2. The sampling of analog input is started by setting the SAMP bit (ADCCON3[9]).
3. After the required sampling time (time delay), the SAMP bit is cleared.
4. The conversion of sampled signal is started by setting the RQCNVRT bit (ADCCON3[8]).
5. Once the conversion is complete, the ARDYx bit of the ADCDSTATx register is set. The data can be read from the ADCDATAx register.

The following figure illustrates the conversion process in graphical form.

Figure 32-6. Individual Conversion Trigger Process



32.5.5 Selecting the Voltage Reference Source

The user application can select the voltage reference for the ADC module, which can be internal or external. The Voltage Reference Input Selection bits, VREFSEL[2:0] (ADCCON3[15:13]), select the voltage reference for analog-to-digital conversions. The upper voltage reference (V_{REFH}) and the lower voltage reference (V_{REFL}) may be the internal AV_{DD} and AV_{SS} voltage rails or the band gap reference generator or the external V_{REFH}+ and V_{REF}- input pins. When the voltage reference and band gap reference are ready, the BGVRRDY (ADCCON2[31]) bit is set. If a Fault occurs in the voltage reference (such as a brown-out), the REFFLT bit (ADCCON2[30]) is set. The

BGVRRDY and REFFLT bits can also generate interrupts if the BGVRIEN bit (ADCCON2[15]) and REFFLTIEN bit (ADCCON2[14]) are set, respectively.

The voltages applied to the external reference pins must comply with certain specifications. See *Electrical Characteristics* from Related Links.

The Analog Input Charge Pump Enable bit, AICPMPEN (ADCCON1[12]), must be set when the difference between the selected reference voltages (V_{REFH} - V_{REFL}) is less than 0.65 * (AV_{DD} - AV_{SS}). Setting this bit does not increase the magnitude of the reference voltage; however, setting this bit reduces the series source resistance to the sampling capacitors. This maximizes the SNR for analog-to-digital conversions using small reference voltage rails.

Related Links

38. Electrical Characteristics

32.5.6 Selecting the Scanned Inputs

All available analog inputs can be configured for scanning. Class 2 and Class 3 inputs are sampled using the shared ADC module. A single conversion trigger source is selected for all of the inputs selected for scanning using the STRGSRC[4:0] bits (ADCCON1[20:16]). On each conversion trigger, the ADC module starts converting (in the natural priority) all inputs specified in the user-specified scan list (ADCCSS1). For Class 2 and Class 3 inputs, the trigger initiates a sequential sample/conversion process in the natural priority order.

An analog input belongs to the scan if it is:

- A Class 3 input. For Class 3 inputs, scan is the only mechanism for conversion.
- A Class 2 input that has the scan trigger selected as the trigger source by selecting the STRIG option in the TRGSRCx[4:0] bits located in the ADCTRG1 and ADCTRG2 registers.

The trigger options available for scan are identical to those available for independent triggering of Class 2 inputs. Any Class 2 inputs that are part of the scan must have the STRIG option selected as their trigger source in the TRGSRCx[4:0] bits.

Note: The end-of-scan (EOS) is generated only if the last shared input conversion has completed. Until this condition is met, the scan sequence is still in effect. Therefore, the EOS Interrupt can be used for any scan sequence with any combination of input types.

The following code is an example for ADC scanning multiple inputs.

```
int main(int argc, char** argv) {
int result[3];

/* Configure ADCCON1 */
ADCCON1 = 0;              // No ADCCON1 features are enabled including: Stop-in-Idle, turbo,
// CVD mode, Fractional mode and scan trigger source. ADCCON1bits.SELRES = 3;   // ADC7
resolution is 12 bits
ADCCON1bits.STRGSRC = 1;  // Select scan trigger.

/* Configure ADCCON2 */
ADCCON2bits.SAMC = 5;     // ADC7 sampling time = 5 * TAD7
ADCCON2bits.ADCDIV = 1;   // ADC7 clock freq is half of control clock = TAD7

/* Initialize warm up time register */ ADCANCON = 0;
ADCANCONbits.WKUPCLKCNT = 5; // Wakeup exponent = 32 * TAD7

/* Clock setting */
ADCCON3bits.ADCSEL = 0;       // Select input clock source
ADCCON3bits.CONCLKDIV = 1;    // Control clock frequency is half of input clock
ADCCON3bits.VREFSEL = 0;      // Select AVDD and AVSS as reference source

ADC0TIMEbits.ADCDIV = 1;      // ADC0 clock frequency is half of control clock = TAD7
ADC0TIMEbits.SAMC = 5;        // ADC0 sampling time = 5 * TAD7
ADC0TIMEbits.SELRES = 3;      // ADC0 resolution is 12 bits

/* Select ADC input mode */
ADCIMCON1bits.SIGN0 = 0;      // unsigned data format
ADCIMCON1bits.DIFF0 = 0;      // Single ended mode
ADCIMCON1bits.SIGN7 = 0;      // unsigned data format
ADCIMCON1bits.DIFF7 = 0;      // Single ended mode

/* Configure ADCGIRQENx */
ADCGIRQEN1 = 0;               // No interrupts are used. ADCGIRQEN2 = 0;
```

```
/* Configure ADCCSSx */
ADCCSS1 = 0;                 // Clear all bits
ADCCSS1bits.CSS0 = 1;         // AN0 set for scan ADCCSS1bits.CSS7 = 1;         // AN7 (Class
2) set for scan

/* Configure ADCCMPCONx */
ADCCMPCON1 = 0;               // No digital comparators are used. Setting the ADCCMPCONx
ADCCMPCON2 = 0;               // register to '0' ensures that the comparator is disabled.

/* Configure ADCFLTRx */
ADCFLTR1 = 0;                // No oversampling filters are used. ADCFLTR2 = 0;

/* Set up the trigger sources */
ADCTRG1bits.TRGSRC0 = 3;      // Set AN0 (Class 2) to trigger from scan source
ADCTRG2bits.TRGSRC7 = 3;      // Set AN7 (Class 2) to trigger from scan source

/* Turn the ADC on */ ADCCON1bits.ON = 1;

/* Wait for voltage reference to be stable */
while(!ADCCON2bits.BGVRRDY); // Wait until the reference voltage is ready
while(ADCCON2bits.REFFLT);   // Wait if there is a fault with the reference voltage

/* Enable clock to analog circuit */
ADCANCONbits.ANEN7 = 1;       // Enable, ADC7

/* Wait for ADC to be ready */
while(!ADCANCONbits.WKRDY0);     // Wait until ADC0 is ready while(!
ADCANCONbits.WKRDY7);     // Wait until ADC7 is ready

/* Enable the ADC module */
ADCCON3bits.DIGEN7 = 1;          // Enable ADC7

while (1) {
/* Trigger a conversion */ ADCCON3bits.GSWTRG = 1;

/* Wait the conversions to complete */
while (ADCDSTAT1bits.ARDY0 == 0);
/* fetch the result */
result[0] = ADCDATA0;

while (ADCDSTAT1bits.ARDY7 == 0);
/* fetch the result */
result[1] = ADCDATA7;

/*
* Process results here
*
*
*/
}
return (1);
}
```

32.5.7 Selecting the Analog-to-Digital Conversion Clock Source and Prescaler

The ADC module can use the internal Fast RC (FRC) oscillator output, system clock (SYS_CLK), reference clock (REFO3) or peripheral bus clock (PB1_CLK) as the conversion clock source (T_Q). See *ADCCON3* register from Related Links.

When the ADCSEL[1:0] bits (ADCCON2[31:30]) = '01', the internal FRC oscillator is used as the ADC clock source. When using the internal FRC oscillator, the ADC module can continue to function in Sleep and Idle modes.

Note: It is recommended that applications that require precise timing of ADC acquisitions use SYS_CLK as the clock source for the ADC.

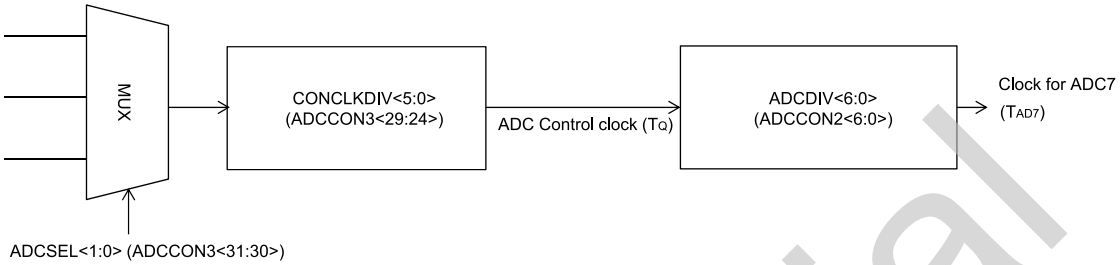
For correct analog-to-digital conversions, the conversion clock limits must not be exceeded. Clock frequencies from 1 MHz to 32 MHz are supported by the ADC module.

The maximum rate that analog-to-digital conversions may be completed by the ADC module (effective conversion throughput) is 2 Msps. However, the maximum rate that a single input can be converted is dependent on the sampling time requirements. In addition, the sampling time depends on the output impedance of the analog signal source. For more information on sampling time, see *ADC Sampling Requirements* from Related Links.

The input clock source for the ADC is selected using the ADCSEL[1:0] bits (ADCCON3[31:30]). The input clock is further divided by the control clock divider CONCLKDIV[5:0] bits (ADCCON3[29:24]). The output clock is called the "ADC control clock" with a time period of T_Q.

The ADC control clock is divided before it is used for the shared ADC by the ADCDIV[6:0] bits (ADCCON2[6:0]). The time period for this clock is denoted as T_{AD7}.

Figure 32-7. Clock Derivation for Shared ADC Modules



Equation 32-1. Sample Time for the Shared ADC Module

$$t_{SAMC} = ADCCON2 < 25:16 > T_{AD7}$$

$$t_{conversion} = 2 + ADCCON2 < 22:21 > T_{AD7}$$

Equation 32-2. ADC Throughput Rate

$$\text{FTP} = T_{AD7}/(T_{SAMC} + T_{CONV})$$

Where,

- T_{AD7} = The frequency of the individual ADC module.

32.5.8 Turning ON the ADC

Turning ON the ADC module involves the following procedure.

When the ADC module enable bit, ON (ADCCON1[15]), is set to '1', the module is in Active mode and is fully powered and functional. When the ON bit is '0', the ADC module is disabled. Once disabled, the digital and analog portions of the ADC are turned off for maximum current savings. In addition to setting the ON bit, the analog and digital circuits of ADC must be turned ON. See *Low-power Mode* from Related Links.

Note: Writing to the ADC control bits that control the ADC clock, input assignments, scanning, voltage reference selection, S&H circuit operating modes and interrupt configuration is not recommended while the ADC module is enabled.

Related Links

32.8.3. Low-Power Mode

32.5.9 ADC Status Bits

The ADC module includes the WKRDY7 status bit in the ADCANCON register, which indicates the current state of ADC Analog and bias circuit. The user application must not perform any ADC operations until this bit is set.

32.6 Additional ADC Functions

This section describes some additional features of the ADC module, which includes:

- Digital comparator
- Oversampling filter

32.6.1 Digital Comparator

The ADC module features digital comparators that can be used to monitor selected analog input conversion results and generate interrupts when a conversion result is within the user-specified limits. Conversion triggers are still required to initiate conversions. The comparison occurs automatically once the conversion is complete. This feature is enabled by setting the Digital Comparator Module Enable bit, ENDCMP (ADCCMPCONx[7]).

The user application makes use of an interrupt that is generated when the analog-to-digital conversion result is higher or lower than the specified high and low limit values in the ADCCMPx register. The high and low limit values are specified in the DCMPHI[15:0] bits (ADCCMPx[31:16]) and the DCMPLO[15:0] bits (ADCCMPx[15:0]).

The CMPEx bits ('x' = 0 through 11) in the ADCCMPENx registers are used to specify which analog inputs are monitored by the digital comparator (for 12 analog inputs, ANx, where 'x' = 0 through 11). The ADCCMPCONx register specifies the comparison conditions that generates an interrupt, as follows:

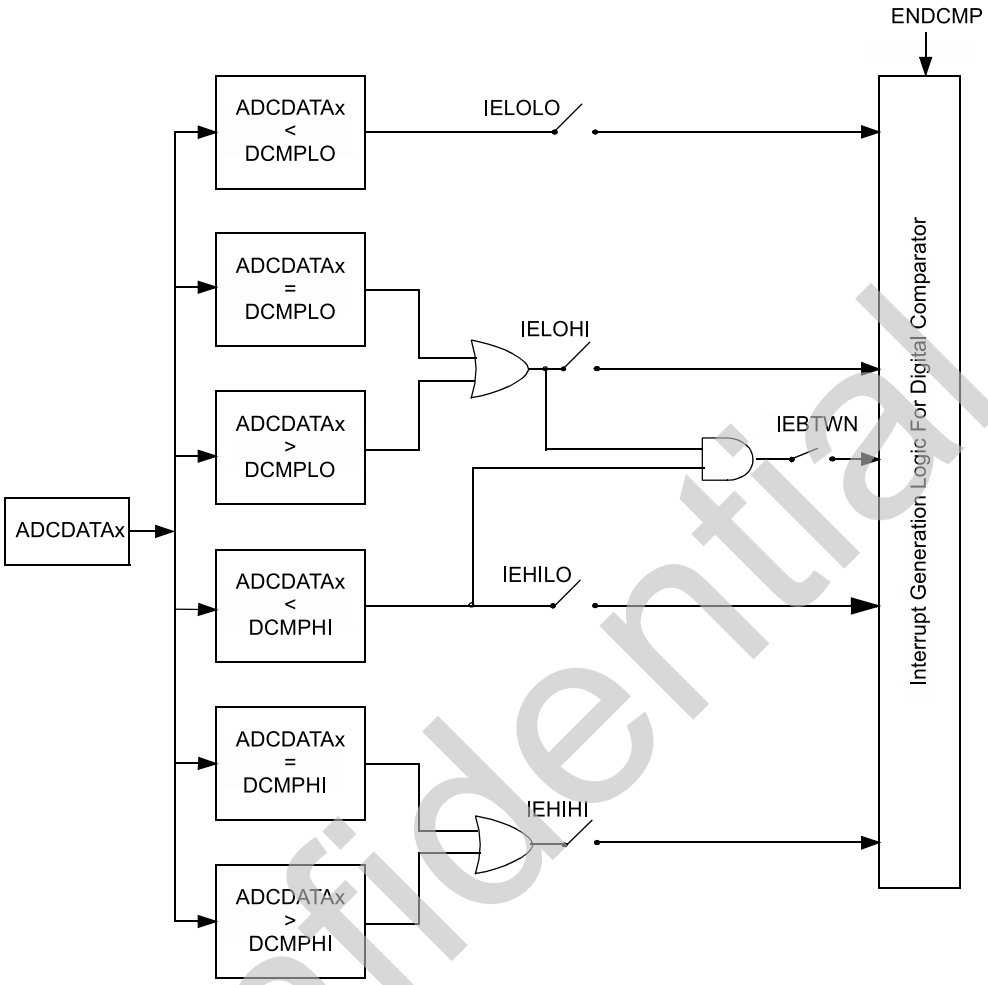
- When IEBTWN = 1, an interrupt is generated when DCMPLO ≤ ADCDATA < DCMPHI
- When IEHIHI = 1, an interrupt is generated when DCMPHI ≤ ADCDATA
- When IEHILO = 1, an interrupt is generated when ADCDATA < DCMPHI
- When IELOHI = 1, an interrupt is generated when DCMPLO ≤ ADCDATA
- When IELOLO = 1, an interrupt is generated when ADCDATA < DCMPLO

The comparator event generation is illustrated in the following figure. When the ADC module generates a conversion result, the conversion result is provided to the comparator. The comparator uses the DIFFx and SIGNx bits of the ADCIMCONx register (depending on the analog input used) to determine the data format used and to appropriately select whether the comparison must be signed or unsigned. The global ADC setting, which is specified by the FRACT bit (ADCCON1[23]), is also used to set the fractional or integer format. The digital comparator compares the ADC result with the high and low limit values (depending on the selected comparison criteria) in the ADCCMPx register.

Depending on the comparator results, a digital comparator interrupt event may be generated. If a comparator event occurs, the Digital Comparator Interrupt Event Detected status bit, DCMPED (ADCCMPCONx[5]), is set, and the Analog Input Identification (ID) bits, AINID[4:0] (ADCCMPCONx[12:8]), are automatically updated so that the user application knows which analog input generated the interrupt event.

Note: The user software must format the values contained in the ADCCMPx registers to match converted data format as either signed or unsigned, and fractional or integer.

Figure 32-8. Digital Comparator



The following code is an example for ADC digital comparator.

```
int main(int argc, char** argv) {
int result = 0, eventFlag = 0;

/* Configure ADCCON1 */
ADCCON1 = 0;                // No ADCCON1 features are enabled including: Stop-in-Idle,
// turbo, CVD mode, Fractional mode and scan trigger source. ADCCON1bits.SELRES = 3;       //
ADC resolution is 12 bits
ADCCON1bits.STRGSRC = 0;      // No scan trigger.

/* Configure ADCCON2 */
ADCCON2bits.SAMC = 5;         // ADC7 sampling time = 5 * TAD7
ADCCON2bits.ADCDIV = 1;       // ADC7 clock freq = TAD7

/* Initialize warm up time register */ ADCANCON = 0;


/* Select ADC input mode */
ADCIMCON1bits.SIGN7 = 0;      // unsigned data format
ADCIMCON1bits.DIFF7 = 0;      // Single ended mode

/* Configure ADCGIRQENx */
ADCGIRQEN1 = 0;               // No interrupts are used
ADCGIRQEN2 = 0;

/* Configure ADCCSSx */
ADCCSS1 = 0;                 // No scanning is used
```

```
/* Configure ADCCMPCONx */
ADCCMP1 = 0;                 // Clear the register ADCCMP1bits.DCMPHI = 0xC00;  // High
limit is a 3072 result. ADCCMP1bits.DCMPLO = 0x500;  // Low limit is a 1280 result.
ADCCMPCON1bits.IEBTWN = 1;    // Create an event when the measured result is
// >= low limits and < high limit. ADCCMPEN1 = 0;               // Clear all enable bits
ADCCMPEN1bits.CMPE8 = 1;      // set the bit corresponding to AN8
ADCCMPCON1bits.ENDCMP = 1;    // enable comparator

/* Configure ADCFLTRx */
ADCFLTR1 = 0;                // No oversampling filters are used. ADCFLTR2 = 0;
/* Set up the trigger sources */
ADCTRG2bits.TRGSRC7 = 3;         // Set AN7 (Class 2) to trigger from scan source

/* Turn the ADC on */ ADCCON1bits.ON = 1;

/* Wait for voltage reference to be stable */
while(!ADCCON2bits.BGVRRDY);     // Wait until the reference voltage is ready
while(ADCCON2bits.REFFLT);       // Wait if there is a fault with the reference voltage

/* Enable clock to analog circuit */
ADCANCONbits.ANEN7 = 1;          // Enable the clock to analog bias

/* Wait for ADC to be ready */
while(!ADCANCONbits.WKRDY7);     // Wait until ADC7 is ready

/* Enable the ADC module */
ADCCON3bits.DIGEN7 = 1;          // Enable ADC7

while (1) {
/* Trigger a conversion */ ADCCON3bits.GSWTRG = 1;

while (ADCDSTAT1bits.ARDY7 == 0);
/* fetch the result */
result = ADCDATA7;

/* Note: It is not necessary to fetch the result for the digital
* comparator to work. In this example we are triggering from
* software so we are using the ARDY8 to gate our loop. Reading the
* data clears the ARDY bit.
*/
/* See if we have a comparator event*/
if (ADCCMPCON1bits.DCMPED == 1) {
eventFlag = 1;
/*
* Process results here
*/
}
}
return (1);
}
```

32.6.2 Oversampling Digital Filter

The ADC module supports two oversampling digital filters. The oversampling digital filter consists of an accumulator and a decimator (down-sampler), which function together as a low-pass filter. By sampling an analog input at a higher-than-required sample rate, then processing the data through the oversampling digital filter, the effective resolution of the ADC module can be increased at the expense of decreased conversion throughput.

To obtain ‘x’ bits of extra resolution, the number of samples required (over and above the Nyquist rate) = $(2^x)^2$:

- 4x oversampling yields one extra bit of resolution (total 13 bits resolution)
- 16x oversampling yields two extra bits of resolution (total 14 bits resolution)
- 64x oversampling provides three extra bits of resolution (total 15 bits resolution)
- 256x oversampling provides four extra bits of resolution (total 16 bits resolution)

The digital filter also has an averaging mode, where it accumulates the samples and divides it by the number of samples.

Note:

1. Only Class 2 analog inputs can engage the digital filter. Therefore, the CHNLID[2:0] bits are 3 bits wide (0 to 7).
2. During the burst conversion process (repeated trigger until all required data for oversampling is obtained), in the case of filtering Class 2 input using the shared ADC module, higher priority ADC inputs may still process conversions; lower priority ADC conversion requests are held waiting until the filter burst sequence is completed.
3. If higher priority requests occur during the digital filter sequence, they delay the completion of the filtering process. This delay may affect the accuracy of the result because the multiple samples cannot be contiguous. The user must arrange the initiation trigger for the oversampling filters to occur while there are no expected interruptions from higher priority ADC conversion requests.

The user application must configure the following bits to perform an oversampling conversion:

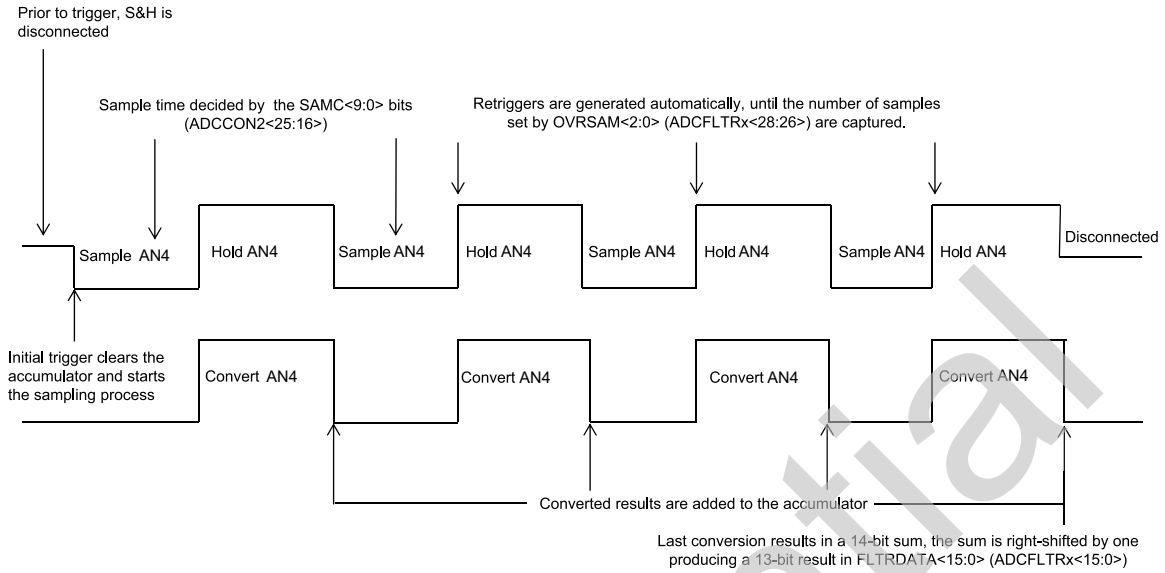
- Select the amount of oversampling through the Oversampling Filter Oversampling Ratio (OVRSAM[2:0]) bits in the ADC Filter register (ADCFLTRx[28:26]).
- Set the filter mode to either Oversampling mode or Averaging mode using the DFMODE bit(ADCFLTRx[29]).
- If the filter is set to Averaging mode and the data format is set to fractional (FRACT bit), set or clear the DATA16EN bit (ADCFLTRx[30]) to set the output resolution.
- Set the sample time for subsequent samples:
  - If using Class 2 inputs, select the sample time using the SAMC[9:0] bits (ADCCON2[25:16]).
- Select the specific analog input to be oversampled by configuring the Analog Input ID Selection bits, CHNLID[4:0] (ADCFLTRx[20:16]).
- If needed, include the oversampling filter interrupt event in the global ADC interrupt by setting the Accumulator Filter Global Interrupt Enable bit, AFGIEN (ADCFLTRx[25]).
- Enable the oversampling filter by setting the Oversampling Filter Accumulator Enable bit, AFEN (ADCFLTRx[31]).

When the digital filter module is configured, the filter's control logic waits for an external trigger to initiate the process. The trigger signal for the analog input to be oversampled causes the accumulator to be cleared and initiates the first conversion. The trigger also forces the trigger sensitivity into level mode and forces the trigger itself to 1 as long as the filter needs to acquire the user-specified number of samples via the OVRSAM[2:0] bits (ADCFLTRx[28:26]). The time delay between each acquired sample is decided by the set sample time in the SAMC[9:0] bits in the ADCCON2 register for Class 2 and the time for conversion. When the required number set by OVRSAM[2:0] are received and processed, the data stored in the FLTRDATA[15:0] bit (ADCFLTRx[15:0]) and the AFRDY bit (ADCFLTRx[24]) is set and the interrupt is generated (if enabled).

The following figure illustrates 4x oversampling using a Class 2 input. Triggering a Class 2 input initiates sampling for the length of time defined by the SAMC[9:0] bits. Retriggers generated by the oversampling logic use the SAMC[9:0] bits to set the sample time.

Class 2 inputs use the shared S&H; therefore, oversampling blocks lower priority Class 2 and Class 3 triggers. Higher priority Class 2 triggers completely disrupt the oversampling process; therefore, they must be avoided completely. The same priority rule applies to two Class 2 inputs that use two digital filters. In such a case, the higher priority input also uses the shared ADC module in Burst mode and prevents the lower priority input from using the shared ADC. Only after all required samples are obtained by the higher priority input can the lower priority input use the shared ADC to acquire samples for its own digital filtering.

Figure 32-9. 4x Oversampling of a Class 2 Input

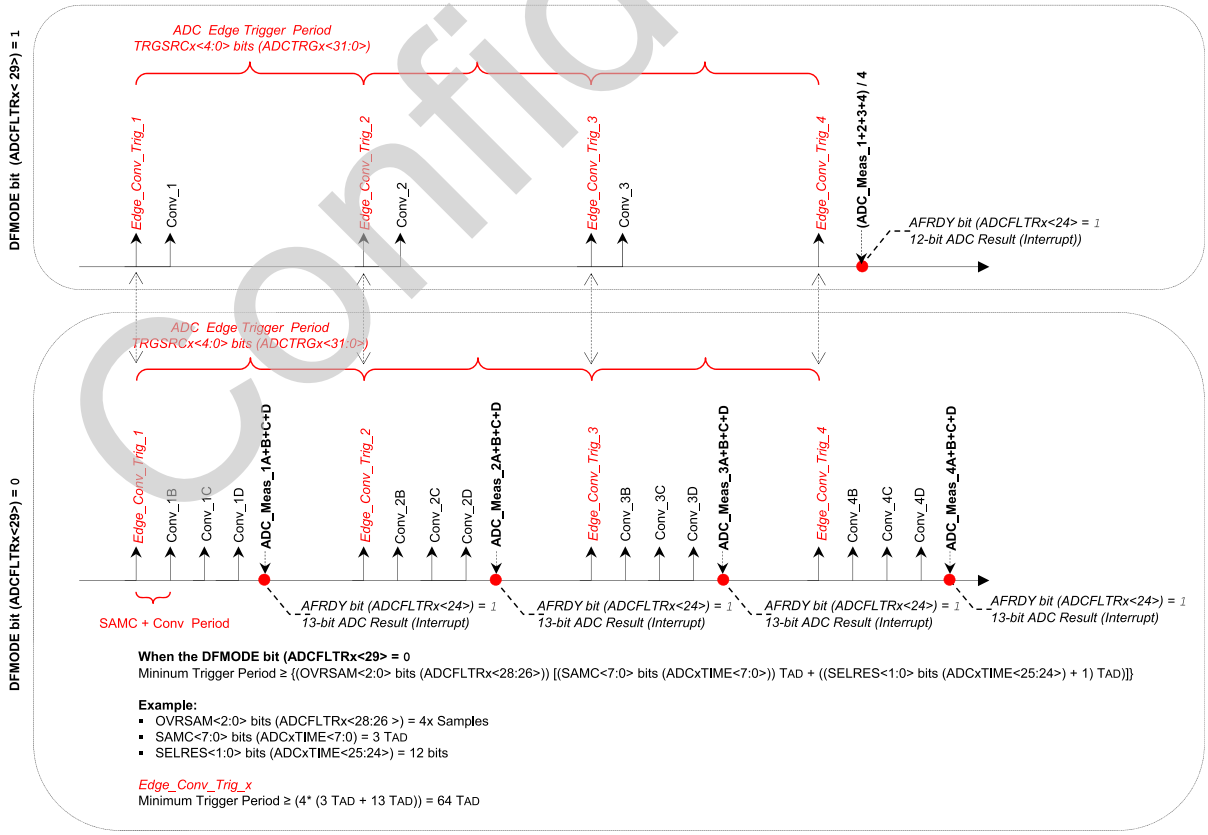


The following code is an example for ADC digital oversampling filter.

```
int main(int argc, char** argv) {
int result;

/* Configure ADCCON1 */
ADCCON1 = 0;          // No ADCCON1 features are enabled including: Stop-in-Idle, turbo,
// CVD mode, Fractional mode and scan trigger source.

/* Initialize warm up time register */ ADCANCON = 0;
ADCANCONbits.WKUPCLKCNT = 5;  // Wake-up exponent = 32 * TAD7

/* Clock setting */ ADCCON3 = 0;
ADCCON3bits.ADCSEL = 0;       // Select input clock source
ADCCON3bits.CONCLKDIV = 1;    // Control clock frequency is half of input clock
ADCCON3bits.VREFSEL = 0;      // Select AVDD and AVSS as reference source

ADCCON2bits.ADCDIV = 1;       // ADC7 clock frequency is half of control clock = TAD7
ADCCON2bits.SAMC = 5;         // ADC7 sampling time = 5 * TAD7
ADCCON1bits.SELRES = 3;       // ADC7 resolution is 12 bits


/* Select ADC input mode */
ADCIMCON1bits.SIGN0 = 0;      // unsigned data format
ADCIMCON1bits.DIFF0 = 0;      // Single ended mode

/* Configure ADCGIRQENx */
ADCGIRQEN1 = 0;               // No interrupts are used
ADCGIRQEN2 = 0;

/* Configure ADCCSSx */
ADCCSS1 = 0;                  // No scanning is used
/* Configure ADCCMPCONx */
ADCCMPCON1 = 0;               // No digital comparators are used. Setting the ADCCMPCONx

/* Configure ADCFLTRx */
ADCFLTR1 = 0;                 // Clear all bits ADCFLTR1bits.CHNLID = 0;      // Use AN0 as
the source ADCFLTR1bits.OVRSAM = 3;      // 16x oversampling ADCFLTR1bits.DFMODE = 0;      //
Oversampling mode ADCFLTR1bits.AFEN = 1;        // Enable filter 1
ADCFLTR2 = 0;                 // Clear all bits

/* Set up the trigger sources */ ADCTRGSNSbits.LVL0 = 0;        // Edge trigger
ADCTRG1bits.TRGSRC0 = 1;     // Set AN0 to trigger from software.

/* Turn the ADC on */
```

```
ADCCON1bits.ON = 1;

/* Wait for voltage reference to be stable */
while(!ADCCON2bits.BGVRRDY); // Wait until the reference voltage is ready
while(ADCCON2bits.REFFLT);   // Wait if there is a fault with the reference voltage

/* Enable clock to analog circuit */
ADCANCONbits.ANEN0 = 1;       // Enable the clock to analog bias and digital control

/* Wait for ADC to be ready */
while(!ADCANCONbits.WKRDY7); // Wait until ADC7 is ready

/* Enable the ADC module */ ADCCON3bits.DIGEN7 = 1;       // Enable ADC7

while (1) {
/* Trigger a conversion */ ADCCON3bits.GSWTRG = 1;

/* Wait for the oversampling process to complete */
while (ADCFLTR1bits.AFRDY == 0);
/* fetch the result */
result = ADCFLTR1bits.FLTRDATA;

/*
* Process result Here
*
* Note 1: Loop time determines the sampling time for the first sample.
* remaining samples sample time is determined by set sampling + conversion time.
*
* Note 2: The first 5 samples may have reduced accuracy.
*
*/
}
return (1);
}
```

Figure 32-10. ADC Filter Comparisons Example



32.7 Interrupts

The ADC module supports interrupts triggered from a variety of sources that can be processed individually or globally. An early interrupt feature is also available to compensate for interrupt servicing latency.

After an enabled interrupt is generated, the CPU jumps to the vector assigned to that interrupt. The CPU begins executing code at the vector address. The user software at this vector address must perform the required operations, such as processing the data results, clearing the interrupt flag, then exiting. See *Nested Vector Interrupt Controller (NVIC)* from Related Links for more information on interrupts and the vector address table details.

32.7.1 Interrupt Sources

The ADC is capable of generating interrupts from the events listed in the following table.

Table 32-4. ADC Interrupt Sources

Interrupt Event	Description	Interrupt Enable Bit	Interrupt Status Bit
ANx Data Ready Event (ADC_GIRQ)	Interrupt is generated upon a completion of a conversion from an analog input source (ANx). Each of the ARDYx bits is capable of generating a unique interrupt when set using the ADCBASE register.	AGIENx of ADCGIRQEN1	ARDYx of ADCDSTAT1 register
Digital Comparator Event (ADC_DIRQ)	When an conversion's comparison criteria are met by a configured and enabled digital comparator. Each of the digital comparators is capable of generating a unique interrupt when its DCMPED bit is set.	DCMPGIEN of ADCCMPCONx register	DCMPED of ADCCMPCONx register
Oversampling Filter Data Ready Event (ADC_AIRQ)	When an oversampling filter has completed the accumulation/ decimation process and has stored the result.	AFGIEN of ADCFLTRx register	AFRDY of ADCFLTRx register
Both Band Gap Voltage and ADC Reference Voltage Ready Event (ADC_BGVR_RDY)	Interrupt is generated when both band gap voltage and ADC reference voltage are ready.	BGVRIEN of ADCCON2 register	BGVRRDY of ADCCON2 register
Band Gap Fault/Reference Voltage Fault/AV_{DD} Brown-out Fault Event (ADC_FLT)	Interrupt is generated when Band Gap Fault/Reference Voltage Fault/ AV_{DD} Brown-out occurs.	REFFLTIEN of ADCCON2 register	REFFLT of ADCCON2 register
End of Scan Event (ADC_FCC)	Interrupt is generated when all the selected inputs have completed scan	EOSIEN of ADCCON2 register	EOSRDY of ADCCON2 register
ADC Module Wake-up Event	Interrupt is generated when ADC wakes up after being enabled.	WKIEN7 of ADCANCON register	WKRDY7 of ADCANCON register
Update Ready Event	Interrupt is generated when ADC SFRs are ready to be (and can be safely) updated with new values.	UPDIEN of ADCCON3 register	UPDRDY of ADCCON3 register

32.7.2 ADC Base Register (ADCBASE) Usage

After conversion of ADC is complete, if the interrupt is vectored to a function that is common to all analog inputs, it takes some significant time to find the ADC input by evaluating the ARDYx bits in the ADCDSTATx. To avoid this time spent, the ADCBASE register is provided, which contains the base address of the user's ADC ISR jump table. When read, the ADCBASE register provides a sum of the contents of the ADCBASE register plus an encoding of the

ARDYx bits set in the ADCDSTATx registers. This use of the ADCBASE register supports the creation of an interrupt vector address that can be used to improve the performance of an ISR.

The ARDYx bits are binary priority encoded with ARDY0 being the highest priority and ARDY11 being the lowest priority. The encoded priority result is, then, shifted left the amount specified by the number of bit positions specified by the IRQVS[2:0] bits in the ADCCON1 register, then added to the contents of the ADCBASE register. If there are no ARDYx bits set, then reading the ADCBASE register equals the value written into the ADCBASE register.

The ADCBASE register is typically loaded with the base address of a jump table that contains the address of the appropriate ISR. The k^{th} interrupt request is enabled via the AGIENx bit (0-11) in ADCGIRQEN1.

The following codes are examples for the ADCBASE register usage.

Case 1:

```
ADCBASE = 0x1234; // Set the address
ADCCON1bits.IRQVS = 2; // left shift by 2
ADCGIRQEN1bits.AGIEN0 = 1; // enable interrupt when AN0 completion is done.
```

When the ADC conversion for AN0 is complete, bit `0` of ADCDSTAT1 = ARDY0 is set.

Read value of ADCBASE = 0x1234 + (0 << 2) = 0x1234.

Therefore, the ISR must be placed at address `0x1234` for AN0.

Case 2:

```
ADCBASE = 0x1234; // Set the address
ADCCON1bits.IRQVS = 2; // left shift by 2
ADCGIRQEN1bits.AGIEN0 = 2; // enable interrupt when AN2 completion is done.
```

When the ADC conversion for AN2 is complete, bit `2` of ADCDSTAT1 = ARDY2 is set.

Read value of ADCBASE = 0x1234 + (2 << 2) = 0x123C.

Therefore, the ISR must be placed at address `0x123C` for AN2.

Note: The contents of the ADCBASE register are not altered. Summation is performed when the ADCBASE register is read and the summation result is the returned read value from the ADCBASE SFR.

32.7.3 Interrupt Enabling, Priority and Vectoring

Each of the ADC events previously mentioned generates an interrupt when its associate Interrupt Enable bit. Each of the ADC events previously listed also has an associated interrupt vector. See *Nested Vector Interrupt Controller* from Related Links for more information on the vector location and control/status bits associated with each individual interrupt.

32.7.4 Individual and Global Interrupts

The use of the individual interrupts previously listed can significantly optimize the servicing of multiple ADC events by keeping each ISR focused on efficiently handling a specific event. In addition, different ISRs can be easily segregated according to the tasks performed, thereby making user software easier to implement and maintain. There may be cases where it is desirable to have a single ISR service multiple interrupt events. To facilitate this, each ADC event can be logically “ORed” to create a single global ADC interrupt. When an ADC event is enabled for a global interrupt, it vectors to a single interrupt routine. It is the responsibility of this single global ISR to determine the source of the interrupt through polling and process it accordingly.

Use of the Global Interrupt requires configuration of its own unique ISER, IPR0, INTFLAG and configuration of its interrupt vector as described in Interrupt Enabling, Priority and Vectoring. See *Nested Vector Interrupt Controller (NVIC)* from Related Links.

Interrupts for the ADC can be configured as individual or global, or utilized as both where some are processed individually and others in the global ISR.

32.8 Power-Saving Modes of Operation

The Power-Saving, Standby Sleep and Idle modes are useful for reducing the conversion noise by minimizing the digital activity of the CPU, buses and other peripherals.

32.8.1 Standby Sleep Mode

When the device enters Standby Sleep mode, the system clock (SYS_CLK) is halted. If an ADC module selects SYS_CLK as its clock source or selects REFO3 as its clock source (REFO3 is generated from SYS-CLK) and Standby Sleep mode occurs during a conversion, the conversion is aborted. The converter cannot resume a partially completed conversion on exiting from Sleep mode. The ADC register contents are not affected by the device entering or leaving Sleep mode. The ADC module can operate during Sleep mode if the ADC clock source is derived from a source other than SYS_CLK that is active during Sleep mode. The FRC clock source is a logical choice for operation during Sleep; however, the REFO3 clock source can also be used, provided it has an input clock that is operational during Sleep mode.

ADC operation during Sleep mode reduces the digital switching noise from the conversion. When the conversion is completed, the ARDYx status bit for that analog input is set and the result is loaded into the corresponding ADC Result register (ADCDATAx).

If any of the ADC interrupts are enabled, the device is woken up from Sleep mode when the ADC interrupt occurs. The program execution resumes at the ADC ISR if the ADC interrupt is greater than the current CPU priority. Otherwise, execution continues from the instruction after the WFI instruction that placed the device in Sleep mode.

To minimize the effects of digital noise on the ADC module operation, the user must select a conversion trigger source that ensures that the analog-to-digital conversion take places in Sleep mode. For example, the external interrupt pin (INT0) conversion trigger option (TRGSRC[4:0] = `00100`) can be used for performing sampling and conversion while the device is in Sleep mode.

Note: For the ADC module to operate in Sleep mode, the ADC clock source must be set to Internal FRC (ADCSEL[1:0] bits (ADCCON2[31:30]) = `01`). Alternately, the REFO3 source can be used; however, the clock source used for REFO3 must operate during Sleep mode. Any changes to the ADC clock configuration require that the ADC be disabled.

32.8.2 Operation During Idle Mode

For the ADC, the stop in Idle Mode bit, SIDL (ADCCON1[13]), specifies whether the ADC module stops on Idle or continues on Idle. If SIDL = `0`, the ADC module continues normal operation when the device enters Idle mode. If any of the ADC interrupts are enabled, the device wakes up from Idle mode when the ADC interrupt occurs. The program execution resumes at the ADC ISR if the ADC interrupt is greater than the current CPU priority. Otherwise, execution continues from the instruction after the WAIT instruction that placed the device in Idle mode.

If SIDL = `1`, the ADC module stops in Idle mode. If the device enters Idle mode during a conversion, the conversion is aborted. The converter cannot resume a partially completed conversion on exiting from Idle mode.

32.8.3 Low-Power Mode

The ADC module can be placed in a low-power state by disabling the digital circuit for individual ADC modules that are not running. This is possible by clearing the DIGEN7 bit in the ADCCON3 register. (See *ADCCON3* register from Related Links.)

An even lower power state is possible by disabling the analog and bias circuit for ADC module that is not running. This is possible by clearing the ANEN7 bit in the ADCANCON register. (See *ADCANCON* register from Related Links.) Disabling the digital circuit to achieve Low-Power mode provides a significantly faster module restart compared to disabling and re-enabling the analog and bias circuit of the ADC module. This is because disabling and re-enabling the analog and bias circuit using the ANEN7 bit requires a wake-up time (typical minimum wake-up time of 20 µs) for the ADC module before it can be used. See *Electrical Characteristics* from Related Links for more information on the stabilization time.

When the analog and bias circuit for an ADC module is enabled, the wake-up must be polled (or through an interrupt) using the wake-up ready bits, WKRDY7, which must be equal to '`1`'.

Related Links

32.13.3. ADCCON3
32.13.20. ADCANCON
38. Electrical Characteristics

32.9 Effects of Reset

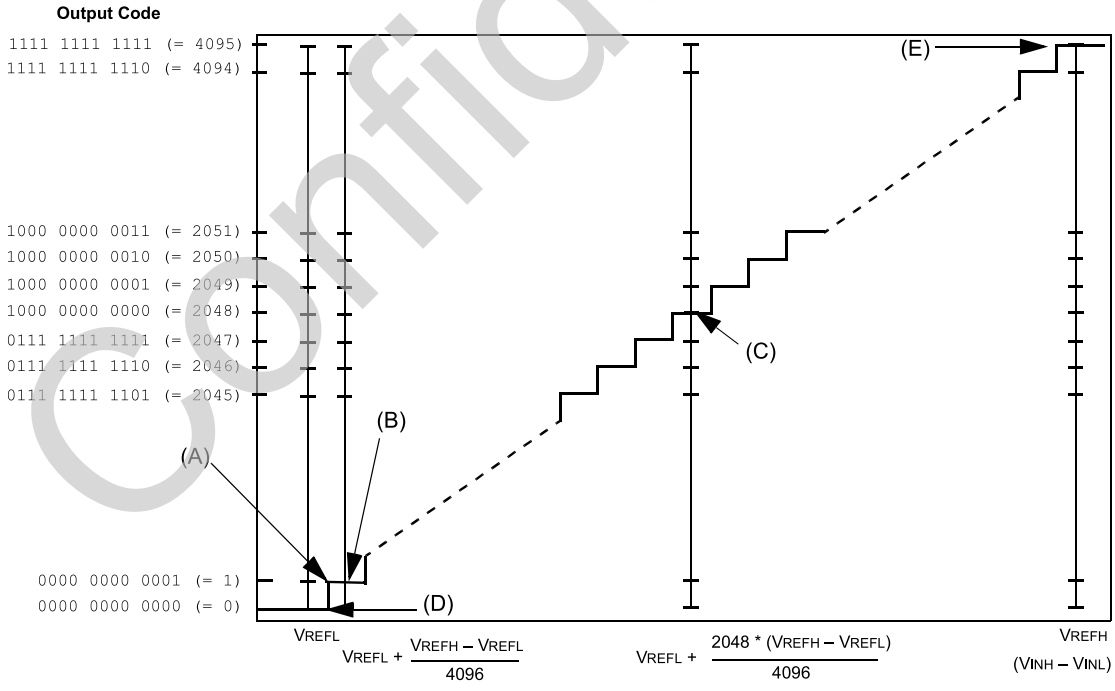
Following any Reset event, all the ADC control and status registers are reset to their default values with control bits in a non-active state. This disables the ADC module and sets the analog input pins to Analog Input mode. Any conversion that was in progress terminates, and the result cannot be written to the result buffer. The values in the ADCDATAx registers are initialized to `0x00000000` during a device Reset. The bias circuits are also turned off, so the ADC resuming operations wait for the bias circuits to stabilize by polling (or requesting to be interrupted by) the BGVRRDY bit (ADCCON2 register).

32.10 Transfer Function

A typical transfer function of the 12-bit ADC is illustrated in the following figure. The difference of the input voltages (V_{INH} - V_{INL}) is compared with the reference (V_{REFH} - V_{REFL}).

- The first code transition (A) occurs when the input voltage is (V_{REFH} - V_{REFL}/8192) or 0.5 LSb.
- The `0000 0000 0001` code is centered at (V_{REFH} - V_{REFL}/4096) or 1.0 LSb (B).
- The `1000 0000 0000` code is centered at (2048 * (V_{REFH} - V_{REFL})/4096) (C).
- An input voltage less than (1 * (V_{REFH} - V_{REFL})/8192) converts as `0000 0000 0000` (D).
- An input greater than (8192 * (V_{REFH} - V_{REFL})/8192) converts as `1111 1111 1111` (E).

Figure 32-11. Analog-to-Digital Transfer Function



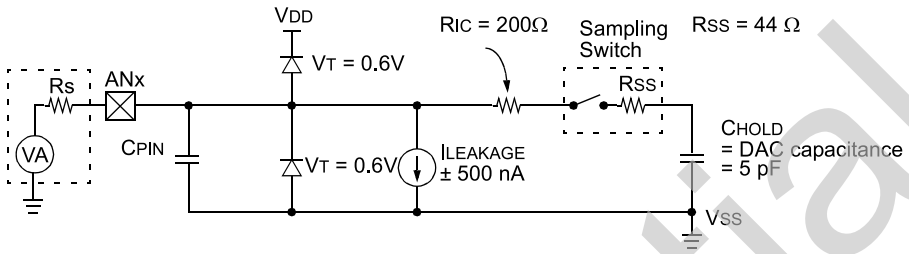
32.11 ADC Sampling Requirements

The analog input model of the 12-bit ADC is illustrated in the following figure. The total acquisition time for the analog-to-digital conversion is a function of the internal circuit settling time and the holding capacitor charge time.

For the ADC module to meet its specified accuracy, the charge holding capacitor (C_{HOLD}) must be allowed to fully charge to the voltage level on the analog input pin. The analog output source impedance (R_S), the interconnect impedance (R_{IC}) and the internal sampling switch (R_{SS}) impedance combine to directly affect the time required to charge the C_{HOLD}. The combined impedance of the analog sources must, therefore, be small enough to fully charge (to within one-fourth LSB of the desired voltage) the holding capacitor within the selected sample time. The internal holding capacitor is in the discharged state prior to each sample operation.

At least 1 T_{AD7} time period must be allowed between conversions for the acquisition time. See *Electrical Characteristics* from Related Links.

Figure 32-12. 12-bit ADC Analog Input Model



Note: The C_{PIN} value depends on the device package and is not tested. The effect of the C_{PIN} is negligible if Rs 5 k.

Legend:

- C_{PIN} = Input capacitance
- R_{SS} = Sampling switch resistance
- R_S = Source resistance
- $I_{LEAKAGE}$ = Leakage current at the pin due to various junctions
- V_T = Threshold voltage
- R_{IC} = Interconnect resistance
- C_{HOLD} = Sample/hold capacitance

32.11.1 Connection Considerations

Because the analog inputs employ Electrostatic Discharge (ESD) protection, they have diodes to V_{DD} and V_{SS}; therefore, the analog input must be between V_{DD} and V_{SS}. The presence of diodes is the reason why the analog pins cannot be 5V tolerant. If the input voltage exceeds this range by greater than 0.3V (either direction), one of the diodes becomes forward biased, and it may damage the device if the input current specification is exceeded.

An external RC filter is sometimes added for antialiasing of the input signal. The R (resistive) component must be selected to ensure that the acquisition time is met. Any external components connected (through high-impedance) to an analog input pin (capacitor, Zener diode and so on) must have very little leakage current at the pin.

32.12 Register Summary

The register offsets shown below are with respect to Base Address = 0x4400_0000.

The PIC32CX-BZ3 12-bit High Speed SAR ADC module has the following Special Function Registers (SFRs):

Note: All registers in this table have corresponding CLR, SET and INV registers at its virtual address, plus an offset of 0x4, 0x8 and 0xC, respectively. See *CLR, SET, and INV Registers* from Related Links.

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x00 ... 0x13FF	Reserved									
0x1400	ADCCON1	7:0		IRQVS[2:0]			STRGLVL			
		15:8	ON	FRZ	SIDL	AIPMPEN	CVD_EN		FSYUPB	SCANEN
		23:16	FRACT	SELRES[1:0]		STRGSRC[4:0]				
		31:24								
0x1404 ... 0x140F	Reserved									
0x1410	ADCCON2	7:0		ADCDIV[6:0]						
		15:8	BGVRIEN	REFFLTIEN	EOSIEN					
		23:16	SAMC[7:0]							
		31:24	BGVRRDY	REFFLT	EOSRDY	CVD_CPL[2:0]			SAMC[9:8]	
0x1414 ... 0x141F	Reserved									
0x1420	ADCCON3	7:0	GLSWTRG	GSWTRG	ADINSEL[5:0]					
		15:8	VREFSEL[2:0]			TRGSUSP	UPDIEN	UPDRDY	SAMP	RQCNVRT
		23:16	DIGEN7							
		31:24	ADCSEL[1:0]		CONCLKDIV[5:0]					
0x1424 ... 0x143F	Reserved									
0x1440	ADCIMCON1	7:0	DIFF3	SIGN3	DIFF2	SIGN2	DIFF1	SIGN1	DIFF0	SIGN0
		15:8	DIFF7	SIGN7	DIFF6	SIGN6	DIFF5	SIGN5	DIFF4	SIGN4
		23:16	DIFF11	SIGN11	DIFF10	SIGN10	DIFF9	SIGN9	DIFF8	SIGN8
		31:24								
0x1444 ... 0x147F	Reserved									
0x1480	ADCGIRQEN1	7:0	AGIEN7	AGIEN6	AGIEN5	AGIEN4	AGIEN3	AGIEN2	AGIEN1	AGIEN0
		15:8					AGIEN11	AGIEN10	AGIEN9	AGIEN8
		23:16								
		31:24								
0x1484 ... 0x149F	Reserved									
0x14A0	ADCCSS1	7:0	CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0
		15:8					CSS11	CSS10	CSS9	CSS8
		23:16								
		31:24								
0x14A4 ... 0x14BF	Reserved									
0x14C0	ADCDSTAT1	7:0	ARDY7	ARDY6	ARDY5	ARDY4	ARDY3	ARDY2	ARDY1	ARDY0
		15:8					ARDY11	ARDY10	ARDY9	ARDY8
		23:16								
		31:24								

.....continued

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x14C4 ... 0x14DF	Reserved									
0x14E0	ADCCMPEN1	7:0	CMPEx[7:0]							
		15:8					CMPEx[11:8]			
		23:16								
		31:24								
0x14E4 ... 0x14EF	Reserved									
0x14F0	ADCCMP1	7:0	DCMPLO[7:0]							
		15:8	DCMPLO[15:8]							
		23:16	DCMPHI[7:0]							
		31:24	DCMPHI[15:8]							
0x14F4 ... 0x14FF	Reserved									
0x1500	ADCCMPEN2	7:0	CMPEx[7:0]							
		15:8					CMPEx[11:8]			
		23:16								
		31:24								
0x1504 ... 0x150F	Reserved									
0x1510	ADCCMP2	7:0	DCMPLO[7:0]							
		15:8	DCMPLO[15:8]							
		23:16	DCMPHI[7:0]							
		31:24	DCMPHI[15:8]							
0x1514 ... 0x159F	Reserved									
0x15A0	ADCFLTR1	7:0	FLTRDATA[7:0]							
		15:8	FLTRDATA[15:8]							
		23:16				CHNLID[4:0]				
		31:24	AFEN	DATA16EN	DFMODE	OVRSAM[2:0]			AFGIEN	AFRDY
0x15A4 ... 0x15AF	Reserved									
0x15B0	ADCFLTR2	7:0	FLTRDATA[7:0]							
		15:8	FLTRDATA[15:8]							
		23:16				CHNLID[4:0]				
		31:24	AFEN	DATA16EN	DFMODE	OVRSAM[2:0]			AFGIEN	AFRDY
0x15B4 ... 0x15FF	Reserved									
0x1600	ADCTRG1	7:0				TRGSRC0[4:0]				
		15:8				TRGSRC1[4:0]				
		23:16				TRGSRC2[4:0]				
		31:24				TRGSRC3[4:0]				
0x1604 ... 0x160F	Reserved									
0x1610	ADCTRG2	7:0				TRGSRC4[4:0]				
		15:8				TRGSRC5[4:0]				
		23:16				TRGSRC6[4:0]				
		31:24				TRGSRC7[4:0]				
0x1614 ... 0x167F	Reserved									

.....continued

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x1680	ADCCMPCON1	7:0	ENDCMP	DCMPGIEN	DCMPED	IEBTWN	IEHIHI	IEHILO	IELOHI	IELOLO
		15:8				AINID[4:0]				
		23:16	CVD_DATA[7:0]							
		31:24	CVD_DATA[15:8]							
0x1684 ... 0x168F	Reserved									
0x1690	ADCCMPCON2	7:0	ENDCMP	DCMPGIEN	DCMPED	IEBTWN	IEHIHI	IEHILO	IELOHI	IELOLO
		15:8				AINID[4:0]				
		23:16								
		31:24								
0x1694 ... 0x16FF	Reserved									
0x1700	ADCBASE	7:0	ADCBASE[7:0]							
		15:8	ADCBASE[15:8]							
		23:16								
		31:24								
0x1704 ... 0x173F	Reserved									
0x1740	ADCTRGSNS	7:0	LVL7	LVL6	LVL5	LVL4	LVL3	LVL2	LVL1	LVL0
		15:8								
		23:16								
		31:24								
0x1744 ... 0x17FF	Reserved									
0x1800	ADCANCON	7:0	ANEN7							
		15:8	WKRDY7							
		23:16	WKIEN7							
		31:24					WKUPCLKCNT[3:0]			
0x1804 ... 0x1AFF	Reserved									
0x1B00	ADCSYSCFG0	7:0	AN[7:0]							
		15:8								
		23:16								
		31:24								
0x1B04 ... 0x1DFF	Reserved									
0x1E00	ADCDATA0	7:0	DATA[7:0]							
		15:8	DATA[15:8]							
		23:16	DATA[23:16]							
		31:24	DATA[31:24]							
...										
0x1E2C	ADCDATA11	7:0	DATA[7:0]							
		15:8	DATA[15:8]							
		23:16	DATA[23:16]							
		31:24	DATA[31:24]							

Related Links

5.4.1.9. CLR, SET and INV Registers

32.13 Register Description

Following conventions are used in the register description:

- – R = Readable bit
- – W = Writable bit
- – U = Unimplemented bit, read as ‘0’
- – -n = Value at POR
- – ‘1’ = Bit is set
- – ‘0’ = Bit is cleared
- – x = Bit is unknown
- HS = Hardware Set
- HC = Hardware Cleared

32.13.1 ADCCON1 – ADC Control Register 1

Name: ADCCON1
Offset: 0x1400
Reset: 0x00601000
Property: -

This register controls the basic operation of the ADC module, including behavior in Sleep and Idle modes, and data formatting. This register also specifies the vector shift amounts for the Interrupt Controller.

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	FRACT	SELRES[1:0]		STRGSRC[4:0]				
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	1	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	ON	FRZ	SIDL	AIPMPEN	CVD_EN		FSYUPB	SCANEN
Access	R/W	R/W	R/W	R/W	R/W		R/W	R/W
Reset	0	0	0	0	0		0	0

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

Bit 23 – FRACT Fractional Data Output Format bit

Value	Description
0	Integer
1	Fractional

Bits 22:21 – SELRES[1:0] Shared ADC Resolution bits

Note: Changing the resolution of the ADC does not shift the result in the corresponding ADCDATAx register. The result occupies 12 bits, with the corresponding lower unused bits set to '0'. For example, a resolution of 6 bits results in ADCDATAx[5:0] being set to '0' and ADCDATAx[11:6] holding the result.

Value	Description
11	12 bits (default)
10	10 bits
01	8 bits
00	6 bits

Bits 20:16 – STRGSRC[4:0] ScanTrigger Source Select bits

Value	Description
10001 - 11111	Reserved
10000	EVSYS_51
01111	EVSYS_50
01110	EVSYS_49
01101	EVSYS_48
01100	EVSYS_47
01011	EVSYS_46
01010	EVSYS_45
01001	EVSYS_44

Value	Description
01000	EVSYS_43
00111	EVSYS_42
00110	EVSYS_41
00101	EVSYS_40
00100	INT0 External interrupt
00011	Reserved
00010	Global level software trigger (GLSWTRG)
00001	Global software edge trigger (GSWTRG)
00000	No Trigger

Bit 15 – ON ADC Module Enable bit

Note: The ON bit must be set only after the ADC module is configured.

Value	Description
0	ADC module is disabled
1	ADC module is enabled

Bit 14 – FRZ Freeze in Debug Mode

Value	Description
0	Do not freeze in Debug mode
1	Freeze in Debug mode

Bit 13 – SIDL Stop in Idle Mode bit

Value	Description
0	Continue module operation in Idle mode
1	Discontinue module operation when device enters Idle mode

Bit 12 – AIPMPEN Analog Input charge Pump Enable

Note: The power-up/reset default value is 1'b0.

Bit 11 – CVD_EN CVD Enable ("Capacitive Voltage Division" Enable) is the bit which enables and starts the CVD operation. The software must ensure that prior to enable CVD_EN.

Notes:

1. The shared ADC core is enabled and ready for conversions, i.e.ADCANCON.ANENx = 1'b1 and ADCANCON.WKRDYx = 1'b1 and ADCCON3.CHN_EN_SHR = 1'b1.
2. The software must disable all external triggers for all second class channels by setting the corresponding ADCTRGx.TRGSRCx[4:0] = 5'h00 and also the third class scan trigger by setting ADCCONx.STRGSRC[4:0] = 5'h00.
3. The software must enable the ADCCSS1.CSSx channel scan select bits of all the channels to be included in the CVD scan.
4. The software must set up the Digital Comparator 1 with the necessary comparison values in ADCCMP1 and the required setup in ADCCMPCON1 (enabling the comparator itself as well as its interrupt if desired). The register ADCCMPEN1 is irrelevant for the CVD operation. The Digital Comparator 1 will update its status field ADCCMPCON1.AINID[5:0] with the channel ID just finished for CVD only upon issuing an ADCCMPCON1.DCMPED interrupt signifying a detected touch event. When the CVD has accomplished its purpose (usually after an interrupt request from the Digital Comparator 1 signaling a touch event), the software must clear first the ADCCON3.DIGEN7 bit before clearing CVD_EN. After that, the software may set again ADCCON3.DIGEN7 and start normal A/D conversions on the shared ADC core for all second and third class channels.

Bit 9 – FSYUPB Fast Synchronous UPB Clock bit

Value	Description
0	Fast synchronous UPB clock is disabled
1	Fast synchronous UPB clock is enabled

Bit 8 – SCANEN SCAN Enable bit

Value	Description
0	When this bit is cleared, the scan cycle is re-triggerable, which means that if an edge-sensitive scan trigger arrives in the middle of a current scan cycle, the current conversion completes but the rest of the current scan is aborted, the EOSRDY status bit is asserted and the scan cycle starts again from the beginning with the first channel included in the scan cycle.
1	By setting this bit, the user disallows the scan trigger to re-start the current scan cycle and the current scan cycle will complete with its last included channel before getting re-started.

Bits 6:4 – IRQVS[2:0] Interrupt Vector Shift bits

To determine the interrupt vector address, this bit specifies the amount of left-shift done to the ARDYx status bits in the ADCDSTAT1 and ADCDSTAT2 registers prior to adding with the ADCBASE register.
Interrupt Vector Address = Read Value of ADCBASE, and Read Value of ADCBASE = Value written to ADCBASE + x << IRQVS[2:0], where 'x' is the smallest active input ID from the ADCDSTAT1 or ADCDSTAT2 registers (which has highest priority).

Value	Description
111	Shift x left 7 bit position
110	Shift x left 6 bit position
101	Shift x left 5 bit position
100	Shift x left 4 bit position
011	Shift x left 3 bit position
010	Shift x left 2 bit position
001	Shift x left 1 bit position
000	Shift x left 0 bit position

Bit 3 – STRGLVL ScanTrigger High Level/Positive Edge Sensitivity bit

Value	Description
0	Scan trigger is positive edge sensitive. Once STRIG mode is selected (TRGSRCx[4:0] in the ADCTRGx register), only a single scan trigger is generated, which completes the scan of all selected analog inputs.
1	Scan trigger is high level sensitive. Once STRIG mode is selected (TRGSRCx[4:0] in the ADCTRGx register), the scan trigger continues for all selected analog inputs, until the STRIG option is removed.

32.13.2 ADCCON2 – ADC Control Register 2

Name: ADCCON2
Offset: 0x1410
Reset: 0x00000000
Property: -

This register controls the reference selection for the ADC module, the sample time for the shared ADC module, interrupt enable for reference, early interrupt selection and clock division selection for the shared ADC.

Bit	31	30	29	28	27	26	25	24
	BGVRRDY	REFFLT	EOSRDY	CVD_CPL[2:0]			SAMC[9:8]	
Access	R/HS/HC	R/HS/HC	R/HS/HC	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
	SAMC[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
	BGVRIEN	REFFLTIEN	EOSIEN					
Access	R/W	R/W	R/W					
Reset	0	0	0					

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

Bit 31 – BGVRRDY Band Gap Voltage/ADC Reference Voltage Status bit
Data processing is valid only after BGVRRDY is set by hardware, so the application code must check that the BGVRRDY bit is set to ensure data validity. This bit set to '0' when ON (ADCCON1[15]) = 0.

Value	Description
0	Either or both band gap voltage and ADC reference voltages (V_{REF}) are not ready
1	Both band gap voltage and ADC reference voltages (V_{REF}) are ready

Bit 30 – REFFLT Band Gap/V_{REF}/A_{VDD} BOR Fault Status bit
This bit is cleared when the ON bit (ADCCON1[15]) = 0 and the BGVRRDY bit = 1.

Value	Description
0	Band gap and V_{REF} voltage are working properly
1	Fault in band gap or the V_{REF} voltage while the ON bit (ADCCON1[15]) was set. Most likely a band gap or V_{REF} fault is caused by a BOR of the analog V_{DD}supply.

Bit 29 – EOSRDY End of Scan Interrupt Status bit
This bit is cleared when ADCCON2[31:24] are read in software.

Value	Description
0	Scanning has not completed
1	All analog inputs are considered for scanning through the scan trigger (all analog inputs specified in the ADCCSS1 register) have completed scanning

Bits 28:26 – CVD_CPL[2:0] CVD Partly Line Capacitor Setting; Cpline = CVD_CPL[2:0] * 2.5 pF = 0 to 17.5 pF

Bits 25:16 – SAMC[9:0] SampleTime for the Shared ADC bits
Where T_{AD7} = Period of the ADC conversion clock for the Shared ADC controlled by the ADCCON2.ADCDIV[6:0] bits.

Value	Description
1111111111	1025 T_{AD7}
...	
0000000001	3 T_{AD7}
0000000000	2 T_{AD7}

Bit 15 – BGVRIEN Band Gap/V_{REF} Voltage Ready Interrupt Enable bit

Value	Description
0	No interrupt is generated when the BGVRRDY bit is set
1	Interrupt is generated when the BGVRDDY bit is set

Bit 14 – REFFLTIEN Band Gap/V_{REF} Voltage Fault Interrupt Enable bit

Value	Description
0	No interrupt is generated when the REFFLT bit is set
1	Interrupt is generated when the REFFLT bit is set

Bit 13 – EOSIEN End of Scan Interrupt Enable bit

Value	Description
0	No interrupt is generated when the EOSRDY bit is set
1	Interrupt is generated when the EOSRDY bit is set

Bits 6:0 – ADCDIV[6:0] Division Ratio for the Shared SAR ADC Core Clock bits
The ADCDIV[6:0] bits divide the ADC control clock (T_Q) to generate the clock for the shared SAR ADC.

Value	Description
1111111	254 * T_Q = T_{AD7}
...	
0000011	6 * T_Q = T_{AD7}
0000010	4 * T_Q = T_{AD7}
0000001	2 * T_Q = T_{AD7}
0000000	Reserved

32.13.3 ADCCON3 – ADC Control Register 3

Name: ADCCON3
Offset: 0x1420
Reset: 0x00000000
Property: -

This register enables ADC clock selection, enables/disables the digital feature for the shared ADC module and controls the manual (software) sampling and conversion.

Note:

1. The SAMP bit has the highest priority and setting this bit keeps the S&H circuit in Sample mode until the bit is cleared. Also, usage of the SAMP bit causes settings of SAMC[9:0] bits (ADCCON2[25:16]) to be ignored.
2. The SAMP bit only connects Class 2 and Class 3 analog inputs to the shared ADC.
3. The SAMP bit is not a self-clearing bit and it is the responsibility of application software to first clear this bit and, only after setting the RQCNVRT bit, to start the analog-to-digital conversion.
4. Normally, when the SAMP and RQCNVRT bits are used by software routines, all TRGSRCx[4:0] bits and STRGSRC[4:0] bits must be set to '00000' to disable all external hardware triggers and prevent them from interfering with the software-controlled sampling command signal SAMP and with the software-controlled trigger RQCNVRT.

Bit	31	30	29	28	27	26	25	24
	ADCSEL[1:0]		CONCLKDIV[5:0]					
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	23	22	21	20	19	18	17	16
	DIGEN7							
Access	R/W							
Reset	0							

Bit	15	14	13	12	11	10	9	8
	VREFSEL[2:0]			TRGSUSP	UPDIEN	UPDRDY	SAMP	RQCNVRT
Access	R/W	R/W	R/W	R/W	R/W	R/HS/HC	R/W	R/HS/HC
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	GLSWTRG	GSWTRG	ADINSEL[5:0]					
Access	R/W	R/W, HC	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 31:30 – ADCSEL[1:0] Analog-to-Digital Clock Source (T_{CLK}) bits

Value	Description
00	Peripheral Bus Clock (PB1_CLK)
01	FRC Clock
10	REFO3 Clock Output
11	System Clock (SYS_CLK)

Bits 29:24 – CONCLKDIV[5:0] Analog-to-Digital Control Clock (T_Q) Divider bits

Value	Description
111111	64 * T_{CLK}= T_Q
...	
000011	4 * T_{CLK}= T_Q
000010	3 * T_{CLK}= T_Q
000001	2 * T_{CLK}= T_Q
000000	T_{CLK}= T_Q

Bit 23 – DIGEN7 Shared ADC Digital Enable bit

Value	Description
1	ADC is digital enabled
0	ADC is digital disabled

Bits 15:13 – VREFSEL[2:0] Voltage Reference (V_{REF}) Input Selection bits

Table 32-5.

VREFSEL[2:0]	AD_{REF+}	AD_{REF-}
000	AV_{DD}	AV_{SS}
001-111	RESERVED FOR FUTURE USE	

Bit 12 – TRGSUSP Trigger Suspend bit

Value	Description
1	Triggers are blocked from starting a new analog-to-digital conversion, but the ADC module is not disabled
0	Triggers are not blocked

Bit 11 – UPDIEN Update Ready Interrupt Enable bit

Value	Description
1	Interrupt is generated when the UPDRDY bit is set by hardware
0	No interrupt is generated

Bit 10 – UPDRDY ADC Update Ready Status bit

Note: This bit is only active while the TRGSUSP bit is set and there are no more running conversions of any ADC modules.

Value	Description
1	ADC SFRs can be updated
0	ADC SFRs cannot be updated

Bit 9 – SAMP Class 2 and Class 3 Analog Input Sampling Enable bit(1,2,3,4)

Value	Description
1	The ADC S&H amplifier is sampling
0	The ADC S&H amplifier is holding

Bit 8 – RQCNVRT Individual ADC Input Conversion Request bit

This bit and its associated ADINSEL[5:0] bits enable the user to individually request an analog-to-digital conversion of an analog input through software.

Note: This bit is automatically cleared in the next ADC clock cycle.

Value	Description
1	Trigger the conversion of the selected ADC input as specified by the ADINSEL[5:0] bits
0	Do not trigger the conversion

Bit 7 – GLSWTRG Global Level Software Trigger bit

Value	Description
1	Trigger conversion for ADC inputs that have selected the GLSWTRG bit as the trigger signal, either through the associated TRGSRC[4:0] bits in the ADCTRGx registers or through the STRGSRC[4:0]bits in the ADCCON1 register
0	Do not trigger an analog-to-digital conversion

Bit 6 – GSWTRG Global Software Trigger bit

This bit is automatically cleared in the next ADC clock cycle.

Value	Description
0	Trigger conversion for ADC inputs that have selected the GSWTRG bit as the trigger signal, either through the associated TRGSRC[4:0] bits in the ADCTRGx registers or through the STRGSRC[4:0]bits in the ADCCON1 register

Value	Description
1	Do not trigger an analog-to-digital conversion

Bits 5:0 – ADINSEL[5:0] Analog Input Select bits

These bits select the analog input to be converted when the RQCNVRT bit is set.

Value	Description
111111	Reserved
...	
001011	VDD33/2(AN11) (Internal)
001010	VDD 1.2V(VDD_1V2(AN10)) (Internal)
001001	ADC Charge-pump 1.2V(CP_1V2(AN9)) (Internal)
001000	BandGap Reference (BG_VREF (AN8)) (Internal)
000111	AN7 is being monitored
...	
000001	AN1 is being monitored
000000	AN0 is being monitored

32.13.4 ADCIMCON1 – ADC Input Mode Control Register 1

Name: ADCIMCON1
Offset: 0x1440
Reset: 0x00000000
Property: -

This register enables the user to select between single-ended and differential operation as well as select between signed and unsigned data format.

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	DIFF11	SIGN11	DIFF10	SIGN10	DIFF9	SIGN9	DIFF8	SIGN8
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
	DIFF7	SIGN7	DIFF6	SIGN6	DIFF5	SIGN5	DIFF4	SIGN4
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
	DIFF3	SIGN3	DIFF2	SIGN2	DIFF1	SIGN1	DIFF0	SIGN0
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 – DIFF11 AN11 Mode bit

Value	Description
1	AN11 is using Differential mode
0	AN11 is using Single-ended mode

Bit 22 – SIGN11 AN11 Signed Data Mode bit

Value	Description
1	AN11 is using Signed Data mode
0	AN11 is using Unsigned Data mode

Bit 21 – DIFF10 AN10 Mode bit

Value	Description
1	AN10 is using Differential mode
0	AN10 is using Single-ended mode

Bit 20 – SIGN10 AN10 Signed Data Mode bit

Value	Description
1	AN10 is using Signed Data mode
0	AN10 is using Unsigned Data mode

Bit 19 – DIFF9 AN9 Mode bit

Value	Description
1	AN9 is using Differential mode
0	AN9 is using Single-ended mode

Bit 18 – SIGN9 AN9 Signed Data Mode bit

Value	Description
1	AN9 is using Signed Data mode
0	AN9 is using Unsigned Data mode

Bit 17 – DIFF8 AN8 Mode bit

Value	Description
1	AN8 is using Differential mode
0	AN8 is using Single-ended mode

Bit 16 – SIGN8 AN8 Signed Data Mode bit

Value	Description
1	AN8 is using Signed Data mode
0	AN8 is using Unsigned Data mode

Bit 15 – DIFF7 AN7 Mode bit

Value	Description
1	AN7 is using Differential mode
0	AN7 is using Single-ended mode

Bit 14 – SIGN7 AN7 Signed Data Mode bit

Value	Description
1	AN7 is using Signed Data mode
0	AN7 is using Unsigned Data mode

Bit 13 – DIFF6 AN6 Mode bit

Value	Description
1	AN6 is using Differential mode
0	AN6 is using Single-ended mode

Bit 12 – SIGN6 AN6 Signed Data Mode bit

Value	Description
1	AN6 is using Signed Data mode
0	AN6 is using Unsigned Data mode

Bit 11 – DIFF5 AN5 Mode bit

Value	Description
1	AN5 is using Differential mode
0	AN5 is using Single-ended mode

Bit 10 – SIGN5 AN5 Signed Data Mode bit

Value	Description
1	AN5 is using Signed Data mode
0	AN5 is using Unsigned Data mode

Bit 9 – DIFF4 AN4 Mode bit

Value	Description
1	AN4 is using Differential mode
0	AN4 is using Single-ended mode

Bit 8 – SIGN4 AN4 Signed Data Mode bit

Value	Description
1	AN4 is using Signed Data mode
0	AN4 is using Unsigned Data mode

Bit 7 – DIFF3 AN3 Mode bit

Value	Description
1	AN3 is using Differential mode
0	AN3 is using Single-ended mode

Bit 6 – SIGN3 AN3 Signed Data Mode bit

Value	Description
1	AN3 is using Signed Data mode
0	AN3 is using Unsigned Data mode

Bit 5 – DIFF2 AN2 Mode bit

Value	Description
1	AN2 is using Differential mode
0	AN2 is using Single-ended mode

Bit 4 – SIGN2 AN2 Signed Data Mode bit

Value	Description
1	AN2 is using Signed Data mode
0	AN2 is using Unsigned Data mode

Bit 3 – DIFF1 AN1 Mode bit

Value	Description
1	AN1 is using Differential mode
0	AN1 is using Single-ended mode

Bit 2 – SIGN1 AN1 Signed Data Mode bit

Value	Description
1	AN1 is using Signed Data mode
0	AN1 is using Unsigned Data mode

Bit 1 – DIFF0 AN0 Mode bit

Value	Description
1	AN0 is using Differential mode
0	AN0 is using Single-ended mode

Bit 0 – SIGN0 AN0 Signed Data Mode bit

Value	Description
1	AN0 is using Signed Data mode
0	AN0 is using Unsigned Data mode

32.13.5 ADCGIRQEN1 – ADC Global Interrupt Enable Register 1

Name: ADCGIRQEN1
Offset: 0x1480
Reset: 0x00000000
Property: -

This register specifies which of the individual input conversion interrupts can generate the global ADC interrupt.

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
					AGIEN11	AGIEN10	AGIEN9	AGIEN8
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	AGIEN7	AGIEN6	AGIEN5	AGIEN4	AGIEN3	AGIEN2	AGIEN1	AGIEN0
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 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 – AGIEN ADC Global Interrupt Enable bits

Value	Description
1	Interrupts are enabled for the selected analog input. The interrupt is generated after the converted data is ready (indicated by the ARDYx bit ('x' = 8-1) of the ADCDSTAT1 register)
0	Interrupts are disabled

32.13.6 ADCCSS1 – ADC Common Scan Select Register 1

Name: ADCCSS1
Offset: 0x14A0
Reset: 0x00000000
Property: -

This register specifies the analog inputs to be scanned by the common scan trigger.

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
					CSS11	CSS10	CSS9	CSS8
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0
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 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 – CSS Analog Common Scan Select bits

Notes:

1. In addition to setting the appropriate bits in this register, Class 2 analog inputs must select the STRIG input as the trigger source if they are to be scanned through the CSSx bits. Refer to the bit descriptions in the ADCTRGx registers for selecting the STRIG option.
2. If a Class 2 input is included in the scan by setting the CSSx bit to '1' and by setting the TRGSRCx[4:0] bits to STRIG mode (0b11), the user application must ensure that no other triggers are generated for that input using the RQCNVRT bit in the ADCCON3 register or the hardware input or any digital filter. Otherwise, the scan behavior is unpredictable.

Value	Description
1	Select ANx for input scan
0	Skip ANx for input scan

32.13.7 ADCDSTAT1 – ADC Data Ready Status Register 1

Name: ADCDSTAT1
Offset: 0x14C0
Reset: 0x00000000
Property: -

This register contains the interrupt status of the individual analog input conversions. Each bit represents the data-ready status for its associated conversion result.

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
					ARDY11	ARDY10	ARDY9	ARDY8
Access					R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	ARDY7	ARDY6	ARDY5	ARDY4	ARDY3	ARDY2	ARDY1	ARDY0
Access	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC
Reset	0	0	0	0	0	0	0	0

Bits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 – ARDY Conversion Data Ready for Corresponding Analog Input Ready bits

Value	Description
1	This bit is set when converted data is ready in the data register
0	This bit is cleared when the associated data register is read

32.13.8 ADCCMPEN1 – ADC Digital Comparator 1 Enable Register

Name: ADCCMPEN1
Offset: 0x14E0
Reset: 0x00000000
Property: -

These registers select which analog input conversion results is processed by the digital comparator.

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
					CMPEx[11:8]			
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	CMPEx[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 11:0 – CMPEx[11:0] ADC Digital Comparator ‘x’ Enable bits
Note: CMPEx = where "x" stands for bit value from 0 to 11.

These bits enable conversion results corresponding to the analog input to be processed by the digital comparator. CMPE0 enables AN0, CMPE1 enables AN1 and so on.
Notes:

1. CMPEx = ANx, where ‘x’ = 0-11 (Digital Comparator inputs are limited to AN0 through AN11).
2. Changing the bits in this register while the Digital Comparator is enabled (ENDCMP = 1) can result in unpredictable behavior.

32.13.9 ADCCMPEN2 – ADC Digital Comparator 2 Enable Register

Name: ADCCMPEN2
Offset: 0x1500
Reset: 0x00000000
Property: -

These registers select which analog input conversion results is processed by the digital comparator.

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
					CMPEx[11:8]			
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
	CMPEx[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 11:0 – CMPEx[11:0] ADC Digital Comparator ‘x’ Enable bits
Note: CMPEx = where ‘x’ stands for bit value from 0 to 11.

These bits enable conversion results corresponding to the analog input to be processed by the digital comparator. CMPE0 enables AN0, CMPE1 enables AN1 and so on.
Notes:

1. CMPEx = ANx, where ‘x’ = 0-11 (Digital Comparator inputs are limited to AN0 through AN11).
2. Changing the bits in this register while the Digital Comparator is enabled (ENDCMP = 1) can result in unpredictable behavior.

32.13.10 ADCCMP1 – ADC Digital Comparator 1 Limit Value Register

Name: ADCCMP1
Offset: 0x14F0
Reset: 0x00000000
Property: -

These registers contain the high and low digital comparison values for use by the digital comparator.

Notes:

1. Changing theses bits while the Digital Comparator is enabled (ENDCMP = 1) can result in unpredictable behavior.
2. The format of the limit values must match the format of the ADC converted value in terms of sign and fractional settings.
3. For Digital Comparator 1 used in CVD mode, the DCMPHI[15:0] and DCMPLO[15:0] bits must always be specified in signed format as the CVD output data is differential and is always signed.

Bit	31	30	29	28	27	26	25	24
	DCMPHI[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	23	22	21	20	19	18	17	16
	DCMPHI[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
	DCMPLO[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
	DCMPLO[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 – DCMPHI[15:0] Digital Comparator 'x' High Limit Value bits(1,2,3)
These bits store the high limit value, which is used by digital comparator for comparisons with ADC converted data.

Bits 15:0 – DCMPLO[15:0] Digital Comparator 'x' Low Limit Value bits(1,2,3)
These bits store the low limit value, which is used by digital comparator for comparisons with ADC converted data.

32.13.11 ADCCMP2 – ADC Digital Comparator 2 Limit Value Register

Name: ADCCMP2
Offset: 0x1510
Reset: 0x00000000
Property: -

These registers contain the high and low digital comparison values for use by the digital comparator.

Notes:

1. Changing theses bits while the Digital Comparator is enabled (ENDCMP = 1) can result in unpredictable behavior.
2. The format of the limit values must match the format of the ADC converted value in terms of sign and fractional settings.

Bit	31	30	29	28	27	26	25	24
	DCMPHI[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	23	22	21	20	19	18	17	16
	DCMPHI[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
	DCMPLO[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
	DCMPLO[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 – DCMPHI[15:0] Digital Comparator 'x' High Limit Value bits(1,2,3)
These bits store the high limit value, which is used by digital comparator for comparisons with ADC converted data.

Bits 15:0 – DCMPLO[15:0] Digital Comparator 'x' Low Limit Value bits(1,2,3)
These bits store the low limit value, which is used by digital comparator for comparisons with ADC converted data.

32.13.12 ADCFLTR1 – ADC Digital Filter 1 Register

Name: ADCFLTR1
Offset: 0x15A0
Reset: 0x00000000
Property: -

These registers provide control and status bits for the oversampling filter accumulator, and also includes the 16-bit filter output data.

Bit	31	30	29	28	27	26	25	24
	AFEN	DATA16EN	DFMODE	OVRSAM[2:0]			AFGIEN	AFRDY
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
				CHNLID[4:0]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	FLTRDATA[15:8]							
Access	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	FLTRDATA[7:0]							
Access	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC
Reset	0	0	0	0	0	0	0	0

Bit 31 – AFEN Digital Filter 'x' Enable bit

Value	Description
1	Digital filter is enabled
0	Digital filter is disabled and the AFRDY status bit is cleared

Bit 30 – DATA16EN Filter Significant Data Length bit

Note: This bit is significant only if DFMODE = 1 (Averaging Mode) and FRACT (ADCCON1[23]) = 1 (Fractional Output Mode).

Value	Description
1	All 16 bits of the filter output data are significant
0	Only the first 12 bits are significant, followed by four zeros

Bit 29 – DFMODE ADC Filter Mode bit

Value	Description
1	Filter 'x' works in Averaging mode
0	Filter 'x' works in Oversampling Filter mode (default)

Bits 28:26 – OVRSAM[2:0] Oversampling Filter Ratio bits

Value	Description
	If DFMODE is '0'
111	128 samples (shift sum 3 bits to right, output data is in 15.1 format)
110	32 samples (shift sum 2 bits to right, output data is in 14.1 format)
101	8 samples (shift sum 1 bit to right, output data is in 13.1 format)
100	2 samples (shift sum 0 bits to right, output data is in 12.1 format)
011	256 samples (shift sum 4 bits to right, output data is 16 bits)
010	64 samples (shift sum 3 bits to right, output data is 15 bits)

Value	Description
001	16 samples (shift sum 2 bits to right, output data is 14 bits)
000	4 samples (shift sum 1 bit to right, output data is 13 bits)
	If DFMODE is '1'
111	256 samples (256 samples to be averaged)
110	128 samples (128 samples to be averaged)
101	64 samples (64 samples to be averaged)
100	32 samples (32 samples to be averaged)
011	16 samples (16 samples to be averaged)
010	8 samples (8 samples to be averaged)
001	4 samples (4 samples to be averaged)
000	2 samples (2 samples to be averaged)

Bit 25 – AFGIEN Digital Filter 'x' Interrupt Enable bit

Value	Description
1	Digital filter interrupt is enabled and is generated by the AFRDY status bit
0	Digital filter is disabled

Bit 24 – AFRDY Digital Filter 'x' Data Ready Status bit

Note: This bit is cleared by reading the FLTRDATA[15:0] bits or by disabling the Digital Filter module (by setting AFEN to '0').

Value	Description
1	Data is ready in the FLTRDATA[15:0] bits
0	Data is not ready

Bits 20:16 – CHNLID[4:0] Digital Filter Analog Input Selection bits

Note: Only the first 8 analog inputs, Class 2 (AN0 -AN7), can use a digital filter.

These bits specify the analog input to be used as the oversampling filter data source.

Value	Description
11111	Reserved
...	
...	
...	
00111	AN7
...	
...	
...	
00010	AN2
00001	AN1
00000	AN0

Bits 15:0 – FLTRDATA[15:0] Digital Filter 'x' Data Output Value bits

The filter output data is as per the fractional format set in the FRACT bit (ADCCON1[23]). The FRACT bit must not be changed while the filter is enabled. Changing the state of the FRACT bit after the operation of the filter ended must not update the value of the FLTRDATA[15:0] bits to reflect the new format.

32.13.13 ADCFLTR2 – ADC Digital Filter 2 Register

Name: ADCFLTR2
Offset: 0x15B0
Reset: 0x00000000
Property: -

These registers provide control and status bits for the oversampling filter accumulator, and also include the 16-bit filter output data.

Bit	31	30	29	28	27	26	25	24
	AFEN	DATA16EN	DFMODE	OVRSAM[2:0]			AFGIEN	AFRDY
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
				CHNLID[4:0]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	FLTRDATA[15:8]							
Access	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	FLTRDATA[7:0]							
Access	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC
Reset	0	0	0	0	0	0	0	0

Bit 31 – AFEN Digital Filter 'x' Enable bit

Value	Description
1	Digital filter is enabled
0	Digital filter is disabled and the AFRDY status bit is cleared

Bit 30 – DATA16EN Filter Significant Data Length bit

Note: This bit is significant only if DFMODE = 1 (Averaging Mode) and FRACT (ADCCON1[23]) = 1 (Fractional Output Mode).

Value	Description
1	All 16 bits of the filter output data are significant
0	Only the first 12 bits are significant, followed by four zeros

Bit 29 – DFMODE ADC Filter Mode bit

Value	Description
1	Filter 'x' works in Averaging mode
0	Filter 'x' works in Oversampling Filter mode (default)

Bits 28:26 – OVRSAM[2:0] Oversampling Filter Ratio bits

Value	Description
	If DFMODE is '0'
111	128 samples (shift sum 3 bits to right, output data is in 15.1 format)
110	32 samples (shift sum 2 bits to right, output data is in 14.1 format)
101	8 samples (shift sum 1 bit to right, output data is in 13.1 format)
100	2 samples (shift sum 0 bits to right, output data is in 12.1 format)
011	256 samples (shift sum 4 bits to right, output data is 16 bits)
010	64 samples (shift sum 3 bits to right, output data is 15 bits)

Value	Description
001	16 samples (shift sum 2 bits to right, output data is 14 bits)
000	4 samples (shift sum 1 bit to right, output data is 13 bits)
	If DFMODE is '1'
111	256 samples (256 samples to be averaged)
110	128 samples (128 samples to be averaged)
101	64 samples (64 samples to be averaged)
100	32 samples (32 samples to be averaged)
011	16 samples (16 samples to be averaged)
010	8 samples (8 samples to be averaged)
001	4 samples (4 samples to be averaged)
000	2 samples (2 samples to be averaged)

Bit 25 – AFGIEN Digital Filter 'x' Interrupt Enable bit

Value	Description
1	Digital filter interrupt is enabled and is generated by the AFRDY status bit
0	Digital filter is disabled

Bit 24 – AFRDY Digital Filter 'x' Data Ready Status bit

Note: This bit is cleared by reading the FLTRDATA[15:0] bits or by disabling the Digital Filter module (by setting AFEN to '0').

Value	Description
1	Data is ready in the FLTRDATA[15:0] bits
0	Data is not ready

Bits 20:16 – CHNLID[4:0] Digital Filter Analog Input Selection bits

Note: Only the first 8 analog inputs, Class 2 (AN0-AN7), can use a digital filter.

These bits specify the analog input to be used as the oversampling filter data source.

Value	Description
11111	Reserved
...	
...	
...	
01100	Reserved
0111	AN7
...	
...	
...	
00010	AN2
00001	AN1
00000	AN0

Bits 15:0 – FLTRDATA[15:0] Digital Filter 'x' Data Output Value bits

The filter output data is as per the fractional format set in the FRACT bit (ADCCON1[23]). The FRACT bit must not be changed while the filter is enabled. Changing the state of the FRACT bit after the operation of the filter ended must not update the value of the FLTRDATA[15:0] bits to reflect the new format.

32.13.14 ADCTRG1 – ADC Trigger Source 1 Register

Name: ADCTRG1
Offset: 0x1600
Reset: 0x00000000
Property: -

This register controls the trigger source selection for AN0 through AN3 analog inputs.

Bit	31	30	29	28	27	26	25	24
				TRGSRC3[4:0]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
				TRGSRC2[4:0]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
				TRGSRC1[4:0]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
				TRGSRC0[4:0]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bits 28:24 – TRGSRC3[4:0] Trigger Source for Conversion of Analog Input AN3 Select bits
Note: For STRIG, in addition to setting the trigger, it also requires programming of the STRGSRC[4:0] bits (ADCCON1[20:16]) to select the trigger source, and requires the appropriate CSS bits to be set in the ADCCSSx registers.

Value	Description
10001 - 11111	Reserved
10000	EVSYS_51
01111	EVSYS_50
01110	EVSYS_49
01101	EVSYS_48
01100	EVSYS_47
01011	EVSYS_46
01010	EVSYS_45
01001	EVSYS_44
01000	EVSYS_43
00111	EVSYS_42
00110	EVSYS_41
00101	EVSYS_40
00100	INT0 External interrupt
00011	STRIG
00010	Global level software trigger (GLSWTRG)
00001	Global software edge trigger (GSWTRG)
00000	No Trigger

Bits 20:16 – TRGSRC2[4:0] Trigger Source for Conversion of Analog Input AN2 Select bits
Note: See bits 28-24 for bit value definitions.

Bits 12:8 – TRGSRC1[4:0] Trigger Source for Conversion of Analog Input AN1 Select bits
Note: See bits 28-24 for bit value definitions.

Bits 4:0 – TRGSRC0[4:0] Trigger Source for Conversion of Analog Input AN0 Select bits
Note: See bits 28-24 for bit value definitions.

32.13.15 ADCTRG2 – ADC Trigger Source 2 Register

Name: ADCTRG2
Offset: 0x1610
Reset: 0x00000000
Property: -

This register controls the trigger source selection for AN4 through AN7 analog inputs.

Bit	31	30	29	28	27	26	25	24
				TRGSRC7[4:0]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
				TRGSRC6[4:0]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
				TRGSRC5[4:0]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
				TRGSRC4[4:0]				
Access				R/W	R/W	R/W	R/W	R/W
Reset				0	0	0	0	0

Bits 28:24 – TRGSRC7[4:0] Trigger Source for Conversion of Analog Input AN7 Select bits
Note: For STRIG, in addition to setting the trigger, it also requires programming of the STRGSRC[4:0] bits (ADCCON1[20:16]) to select the trigger source, and requires the appropriate CSS bits to be set in the ADCCSSx registers.

Value	Description
10001 - 11111	Reserved
10000	EVSYS_51
01111	EVSYS_50
01110	EVSYS_49
01101	EVSYS_48
01100	EVSYS_47
01011	EVSYS_46
01010	EVSYS_45
01001	EVSYS_44
01000	EVSYS_43
00111	EVSYS_42
00110	EVSYS_41
00101	EVSYS_40
00100	INT0 External interrupt
00011	STRIG
00010	Global level software trigger (GLSWTRG)
00001	Global software edge trigger (GSWTRG)
00000	No Trigger

Bits 20:16 – TRGSRC6[4:0] Trigger Source for Conversion of Analog Input AN6 Select bits
Note: See bits 28-24 for bit value definitions.

Bits 12:8 – TRGSRC5[4:0] Trigger Source for Conversion of Analog Input AN5 Select bits
Note: See bits 28-24 for bit value definitions.

Bits 4:0 – TRGSRC4[4:0] Trigger Source for Conversion of Analog Input AN4 Select bits
Note: See bits 28-24 for bit value definitions.

32.13.16 ADCCMPCON1 – ADC Digital Comparator 1 Control Register

Name: ADCCMPCON1
Offset: 0x1680
Reset: 0x00000000
Property: -

This register controls the operation of Digital Comparator 1, including the generation of interrupts, comparison criteria to be used and provides status when a comparator event occurs.

Bit	31	30	29	28	27	26	25	24
	CVD_DATA[15:8]							
Access	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	CVD_DATA[7:0]							
Access	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
				AINID[4:0]				
Access				R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC
Reset				0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	ENDCMP	DCMPGIEN	DCMPED	IEBTWN	IEHIHI	IEHILO	IELOHI	IELOLO
Access	R/W	R/W	R/HS/HC	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 31:16 – CVD_DATA[15:0] CVD Differential Output Data
In CVD mode this 16-bit field gets the CVD differential output data whenever a DCMPED interrupt is generated. The value in this field is ADCCON1.FRACT-compliant and ALWAYS signed, because it is the result of the subtraction between the CVD positive and negative measurements.

Bits 12:8 – AINID[4:0] Digital Comparator 1 Analog Input Identification (ID) bits
When a digital comparator event occurs (DCMPED = 1), these bits identify the analog input being monitored by digital comparator 1.

Value	Description
11111	Reserved
...	
...	
...	
01011	AN11 is being monitored
...	
01000	AN8 is being monitored
00111	AN7 is being monitored
...	
00001	AN1 is being monitored
00000	AN0 is being monitored

Bit 7 – ENDCMP Digital Comparator 1 Enable bit

Value	Description
1	Digital comparator 1 is enabled
0	Digital comparator 1 is not enabled, and the DCMPED status bit (ADCCMPCON[5]) is cleared

Bit 6 – DCMPGIEN Digital Comparator 1 Global Interrupt Enable bit

Value	Description
1	A Digital comparator 1 interrupt is generated when the DCMPED status bit (ADCCMPCON[5]) is set
0	A Digital comparator 1 interrupt is disabled

Bit 5 – DCMPED Digital Comparator 1 "Output True" Event Status bit

The logical conditions under which the digital comparator becomes "True" are defined by the IEBTWN, IEHIHI, IEHILO, IELOHI and IELOLO bits.

Note: This bit is cleared by reading the AINID[4:0] bits or by disabling the Digital Comparator module (by setting ENDCMP to '0').

Value	Description
1	Digital comparator 1 output true event has occurred (output of comparator is '1')
0	Digital comparator 1 output is false (output of comparator is '0')

Bit 4 – IEBTWN Between Low/High Digital Comparator 1 Event bit

Value	Description
1	Generate a digital comparator event when DATA[31:0] is less than DCMPHI[15:0] AND greater than DCMPLO[15:0]
0	Do not generate a digital comparator event

Bit 3 – IEHIHI High/High Digital Comparator 1 Event bit

Value	Description
1	Generate a digital comparator 1 event when DCMPHI[15:0] bits are less than or equal to DATA[31:0] bits
0	Do not generate an event

Bit 2 – IEHILO High/Low Digital Comparator 1 Event bit

Value	Description
1	Generate a digital comparator 1 event when DATA[31:0] bits are less than DCMPHI[15:0] bits
0	Do not generate an event

Bit 1 – IELOHI Low/High Digital Comparator 1 Event bit

Value	Description
1	Generate a digital comparator 1 event when DCMPLO[15:0] bits are less than or equal to DATA[31:0] bits
0	Do not generate an event

Bit 0 – IELOLO Low/Low Digital Comparator 1 Event bit

Value	Description
1	Generate a digital comparator 1 event when DATA[31:0] bits are less than DCMPLO[15:0] bits
0	Do not generate an event

32.13.17 ADCCMPCON2 – ADC Digital Comparator 2 Control Register

Name: ADCCMPCON2
Offset: 0x1690
Reset: 0x00000000
Property: -

These registers control the operation of Digital Comparator 2, including the generation of interrupts and the comparison criteria to be used. This register also provides the status when a comparator event occurs.

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
				AINID[4:0]				
Access				R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC	R/HS/HC
Reset				0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	ENDCMP	DCMPGIEN	DCMPED	IEBTWN	IEHIHI	IEHILO	IELOHI	IELOLO
Access	R/W	R/W	R/HS/HC	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 12:8 – AINID[4:0] Digital Comparator 2 Analog Input Identification (ID) bits
When a digital comparator event occurs (DCMPED = 1), these bits identify the analog input being monitored by the digital comparator.

Value	Description
11111	Reserved
01011	AN11 is being monitored
...	
...	
...	
00111	AN7 is being monitored
...	
00001	AN1 is being monitored
00000	AN0 is being monitored

Bit 7 – ENDCMP Digital Comparator 2 Enable bit

Value	Description
1	Digital comparator 2 is enabled
0	Digital comparator 2 is not enabled, and the DCMPED status bit is cleared

Bit 6 – DCMPGIEN Digital Comparator 2 Global Interrupt Enable bit

Value	Description
1	Digital comparator 2 interrupt is generated when the DCMPED status bit is set
0	Digital comparator 2 interrupt is disabled

Bit 5 – DCMPED Digital Comparator 2 “Output True” Event Status bit
The logical conditions where the digital comparator gets “True” are defined by the IEBTWN, IEHIHI, IEHILO, IELOHI and IELOLO bits.

Note: This bit is cleared by reading the AINID[4:0] bits or by disabling the Digital Comparator module (by setting ENDCMP to '0').

Value	Description
1	Digital comparator 2 output true event has occurred (output of comparator is '1')
0	Digital comparator 2 output is false (output of comparator is '0')

Bit 4 – IEBTWN Between Low/High Digital Comparator 2 Event bit

Value	Description
1	Generate a digital comparator event when DCMPLO[15:0] bits DATA[31:0] bits [DCMPHI[15:0] bits
0	Do not generate a digital comparator event

Bit 3 – IEHIHI High/High Digital Comparator 2 Event bit

Value	Description
1	Generate a digital comparator 2 event when DCMPHI[15:0] bits are less than or equal to DATA[31:0] bits
0	Do not generate an event

Bit 2 – IEHILO High/Low Digital Comparator 2 Event bit

Value	Description
1	Generate a digital comparator 2 event when DATA[31:0] bits are less than DCMPHI[15:0] bits
0	Do not generate an event

Bit 1 – IELOHI Low/High Digital Comparator 2 Event bit

Value	Description
1	Generate a digital comparator 2 event when DCMPLO[15:0] bits are less than or equal to DATA[31:0] bits
0	Do not generate an event

Bit 0 – IELOLO Low/Low Digital Comparator 2 Event bit

Value	Description
1	Generate a digital comparator 2 event when DATA[31:0] bits are less than DCMPLO[15:0] bits
0	Do not generate an event

32.13.18 ADCBASE – ADC Base Register

Name: ADCBASE
Offset: 0x1700
Reset: 0x00000000
Property: -

This register specifies the base address of the user ADC Interrupt Service Routine (ISR) jump table.

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
	ADCBASE[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
	ADCBASE[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 – ADCBASE[15:0] ADCISR Base Address bits
This register, when read, contains the base address of the user's ADC ISR jump table. The interrupt vector address is determined by the IRQVS[2:0] bits of the ADCCON1 register specifying the amount of left shift done to the ARDYx status bits in the ADCDSTAT1 register, prior to adding with ADCBASE register.
Interrupt vector address = Read value of ADCBASE
Read value of ADCBASE = Value written to ADCBASE + x << ADCCON1.IRQVS[2:0], where 'x' is the smallest active analog input ID from the ADCDSTAT1 register (which has the highest priority).

32.13.19 ADCTRGSNS – ADC Trigger Level/Edge Sensitivity Register

Name: ADCTRGSNS
Offset: 0x1740
Reset: 0x00000000
Property: -

This register contains the setting for trigger level for each ADC analog input.

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
	LVL7	LVL6	LVL5	LVL4	LVL3	LVL2	LVL1	LVL0
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 0, 1, 2, 3, 4, 5, 6, 7 – LVL Trigger Level and Edge Sensitivity bits

Notes:

1. This register specifies the trigger level for analog inputs 0 to 7.
2. The higher analog input ID belongs to Class 3, and, therefore, is only scan triggered. All Class 3 analog inputs use the scan trigger, for which the level/edge is defined by the STRGLVL bit (ADCCON1[3]).

Value	Description
1	Analog input is sensitive to the high level of its trigger (level sensitivity implies retriggering as long as the trigger signal remains high)
0	Analog input is sensitive to the positive edge of its trigger (this is the value after a reset)

32.13.20 ADCANCON – ADC Analog Warm-up Control Register

Name: ADCANCON
Offset: 0x1800
Reset: 0x00000000
Property: -

This register contains the warm-up control settings for the analog and bias circuit of the ADC module.

Bit	31	30	29	28	27	26	25	24
					WKUPCLKCNT[3:0]			
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	23	22	21	20	19	18	17	16
	WKIEN7							
Access	R/W							
Reset	0							

Bit	15	14	13	12	11	10	9	8
	WKRDY7							
Access	R/HS/HC							
Reset	0							

Bit	7	6	5	4	3	2	1	0
	ANEN7							
Access	R/W							
Reset	0							

Bits 27:24 – WKUPCLKCNT[3:0] Wake-up Clock Count bits
These bits represent the number of ADC clocks required to warm-up the ADC module before it can perform conversion. Although the clocks are specific to each ADC, the WKUPCLKCNT bit is common to all ADC modules.

Value	Description
1111	2^{15}= 32,768 clocks
...	
...	
...	
0110	2^{6}= 64 clocks
0101	2^{5}= 32 clocks
0100	2^{4}= 16 clocks
0011	2^{4}= 16 clocks
0010	2^{4}= 16 clocks
0001	2^{4}= 16 clocks
0000	2^{4}= 16 clocks

Bit 23 – WKIEN7 Shared ADC Wake-up Interrupt Enable bit

Value	Description
1	Enable interrupt and generate interrupt when the WKRDY7 status bit is set
0	Disable interrupt

Bit 15 – WKRDY7 Shared ADC Wake-up Status bit
Note: This bit is cleared by hardware when the ANEN7 bit is cleared.

Value	Description
1	ADC Analog and bias circuitry ready after the wake-up count number $2^{WKUPEXP}$ clocks after setting ANEN2 to '1'
0	ADC Analog and bias circuitry is not ready

Bit 7 – ANEN7 Shared ADC Analog and Bias Circuitry Enable bit

Value	Description
1	Analog and bias circuitry enabled. Once the analog and bias circuit is enabled, the ADC module needs a warm-up time, as defined by the ADCANCON.WKUPCLKCNT[3:0] bits.
0	Analog and bias circuitry disabled

32.13.21 ADCSYSCFG0 – ADC System Configuration Register 0

Name: ADCSYSCFG0
Offset: 0x1B00
Reset: 0x00000000
Property: -

This register contains read-only bits corresponding to the analog input.

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
	AN[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 7:0 – AN[7:0] ADC Analog Input bits
These bits reflect the system configuration and are updated during boot-up time. By reading these read-only bits, the user application can determine whether or not an analog input in the device is available.

32.13.22 ADCDATAx – ADC Output Data Register ('x' = 0 to 11)

Name: ADCDATAx
Offset: 0x1E00 + x*0x04 [x=0..11]
Reset: 0x00000000
Property: -

These registers are the analog-to-digital conversion output data registers. The ADCDATAx register is associated with each external analog input, 0-7, plus internal analog inputs 8-11.

Bit	31	30	29	28	27	26	25	24
	DATA[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
	DATA[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
	DATA[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
	DATA[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – DATA[31:0] ADC Converted Data Output bits

Note:

1. Reading the ADCDATAx register value after changing the ADCCON1.FRACT bit converts the data into the format specified by the ADCCON1.FRACT bit.

33. Capacitive Voltage Divider (CVD) Controller

33.1 Overview

The PIC32CX-BZ3 device family supports a hardware CVD controller, which supports enhanced CVD self, and self and mutual measurements. The enhanced CVD controller offloads the CPU by performing CVD scans with programmable phase timing and oversampling. Also, the enhanced CVD controller calculates the measurement deltas and detects the touches based on thresholds.

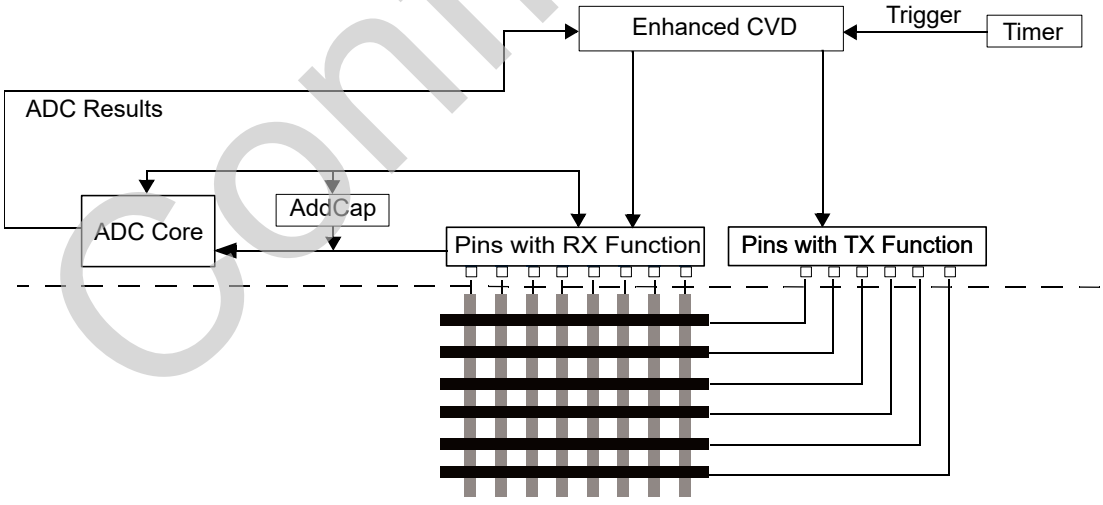
33.2 Features

The key features of the Capacitive Voltage Divider (CVD) Controller module are:

- Self measurement for basic touch detection.
- Supports upto eight RX/TX channels for touch measurements.
- AddCap control to optimize sensitivity on systems with small sample capacitors.
- Support for busing of multiple RX inputs and/or TX outputs for detecting touch over larger areas and algorithmically searching for touch location.
- Four Scan Descriptors control the scan settings to enable software controlled search algorithms by loading the next scan parameters while one is in progress.
- Oversampling of measurements to increase the signal-to-noise ratio.
- Ability to control the sequencing order of RX and TX scanning.
- Programmable thresholding to limit the data to the CPU to only those that exceed the set threshold.
- Supported in Standby Sleep and Idle mode.

33.3 Block Diagram

Figure 33-1. Capacitive Voltage Divider (CVD) Controller Block Diagram



33.4 CVD Module Operation

The enhanced CVD uses the shared ADC SARCORE (See *Analog-to-Digital Converter (ADC)* from Related Links) for its operations. The enhanced CVD controller controls the shared ADC SARCORE in a simplified mode that supports only the needs of CVD, performing the full sequence of an oversampled CVD touch scan and associated math for self, and combination of self and mutual mode.

The controller also controls pin functions needed for the CVD operations. Some of these pins provide RX functionality by having an analog. Some pins may have both RX and TX functions. The RX and TX pins connect to a matrix of button electrodes, a touch screen, or touch pad electrode grid with horizontal and vertical structures. To determine a touch or an approach, measure the capacitance of these electrodes.

An AddCap module connects to the analog input net to augment the internal capacitance of the chip to match the base capacitance of the touch panel. A timer module provides the trigger signals.

Related Links

32. Analog-to-Digital Converter (ADC)

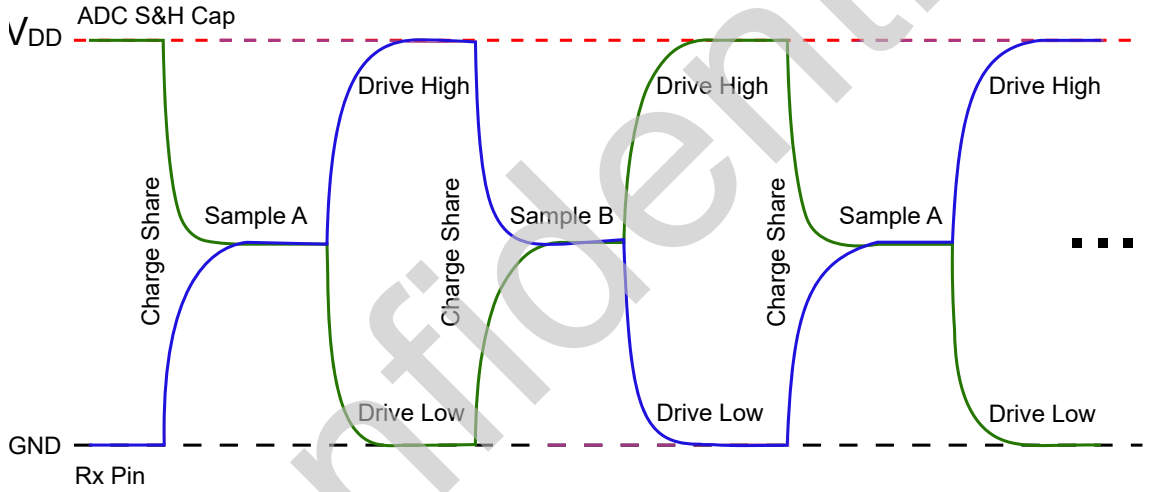
33.4.1 Theory of Operation

This section describes the full sequence of an oversampled CVD touch scan and associated math for self, and combination of self and mutual mode.

33.4.1.1 CVD Self Sensing

Connect a capacitive sensor to one of the PIC32CX-BZ3 device's analog pins and then perform the sampling exclusively by manipulating the input or output ports, and the ADC. The following figure shows the waveform of CVD self sensing configuration.

Figure 33-2. CVD Self Sensing Waveform



Step 1: Precharge the Capacitors

Charge the two capacitors to opposite voltages. First time charging of these capacitors are shown in the "Sample A" (see, Figure 33-2) and the second time (described in Steps 4-6) is "Sample B" (see, Figure 33-2). Configurations of these samples are:

- Sample A:
  - CVD Rx pin discharged to GND.
  - ADC S and H capacitor charged to V_{DD}.
- Sample B:
  - CVD Rx pin charged to V_{DD}.
  - ADC S and H capacitor discharged to GND.

Step 2: Connect the Capacitors and Settle

Connect the two capacitors in parallel and allow the charges to settle. As the external capacitance (CVD Rx Pin) increases, its initial charge (see, following equation) also increases. The internal capacitance (ADC S and H capacitor) does not change, therefore its charge remains constant.

Equation 33-1. Precharge Stage

$$Q_{base} = C_{base}V_{base}$$

$$Q_{hold} = C_{hold}V_{hold}$$

Where,

- Q_{base} is the total external charge.
- Q_{hold} is the total internal charge.
- V_{base} is the voltage provided to the external sensor during the precharge stage.
- V_{hold} is the voltage provided to the internal ADC hold capacitance during precharge.
- C_{base} is the base external capacitance in the released state.
- C_{hold} is the internal ADC capacitance for the sensor.

Step 3: ADC Conversion

Determine the final voltage on $C_{hold}$by the size of the external capacitance (CVD Rx pin) in relation to the size of the internal capacitance (ADC S and H capacitor) (see following equation).

Equation 33-2. Acquisition Stage

$$V_{settle} = \frac{C_{hold}V_{hold} + C_{base}V_{base}}{C_{hold} + C_{base}}$$

$$C_{total} = C_{hold} \parallel C_{base} = C_{hold} + C_{base}$$

$$Q_{total} = C_{hold}V_{hold} + C_{base}V_{base}$$

Where,

- Q_{total} is the total charge on both capacitors when connected during the Acquisition stage.
- C_{total} is the total capacitance of the circuit during the Acquisition stage when both internal and external capacitors are connected and sharing charge.
- V_{settle} is the final settling voltage during the Acquisition stage of a normal CVD waveform.

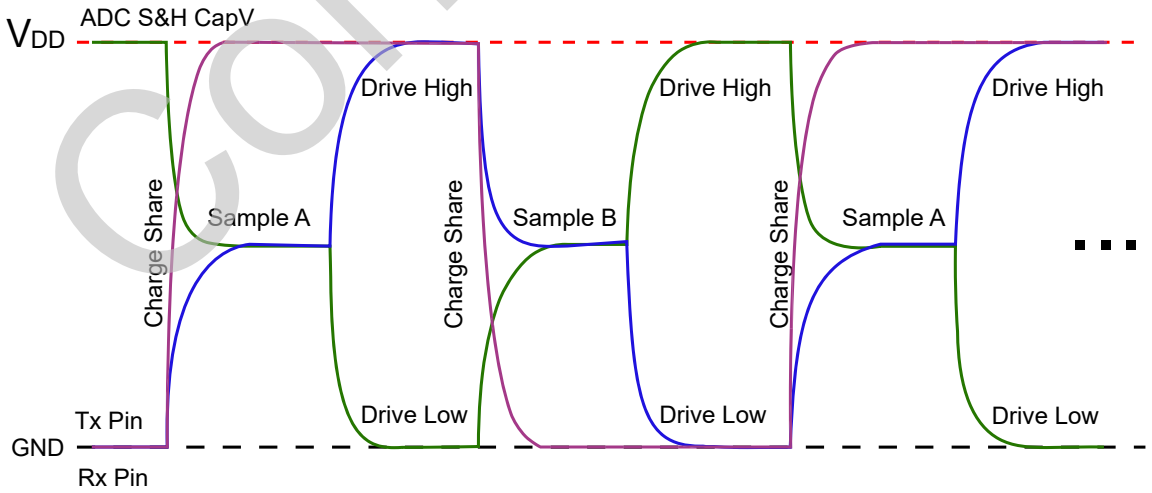
Steps 4-6: Reverse the Precharge Voltages and Repeat

Charge the capacitors, but during this time the precharge voltages are reversed. Use the difference between the two results as the current sensor reading.

33.4.1.2 CVD Self and Mutual Sensing

The mutual drive signal (Tx pin) is a square wave that is driven high during all of Sample A, and low during all of Sample B. This signal aligns in-phase with the external sensor's starting voltage for each sample. The following figure shows the waveform of CVD self and mutual sensing configuration.

Figure 33-3. CVD Self and Mutual Sensing Waveform



Using the capacitive sensor measure the two effects when generating the CVD waveform:

- Measure the capacitance of the sensor as normal due to the typical CVD waveform physics.
  - Sample A increases in value when the ADC S and H capacitance decreases.
  - Sample B decrease in value when the ADC S and H capacitance increases.

33.4.2 Channel Sets

The enhanced CVD module can control up to eight RX inputs and eight TX inputs. It enables the user to map up to eight specific CVDRx pins sequential RX indexes, and up to eight specific CVDTx pins to sequential TX indexes via Special Function Registers (SFRs). The actual pins used can therefore be in any order and with any gaps required. This enables the PCB designer to map I/O pins to panel RX/TX functions with greater ease.

Each scan can span multiple RX/TX indexes via the STRIDE setting, enabling the user to scan multiple RX inputs in parallel, or drive multiple TX outputs in parallel. This is useful for doing full-panel touch detection or for performing a touch search algorithm.

33.4.2.1 RX/TX Striding

The enhanced CVD employs the RX/TX striding to achieve the busing of multiple RX and/or TX pins together for a single measurement sequence. After programming the RX and TX indexes, user can configure a scan to run by simultaneously utilizing multiples of these indexes per measurement.

For example, if the user has eight RX and four TX pins, and needs to scan four RX and two TX simultaneously. Performing a scan for the panel in groups, programs the RX stride to indicate four per measurement and programs the TX stride to indicate two per measurement. The scan then measures as mentioned in the following steps:

1. RX0-3/TX0-1
2. RX4-7/TX0-1
3. RX0-3/TX2-3, and so on.

33.4.2.2 Scan Descriptors

The enhanced CVD module includes four scan descriptors to off-load the intervention requirements and timing dependencies of the CPU. Each scan descriptor includes the RX and TX indexes to be scanned and the strides for each, as well as the time for each measurement phase, the amount of oversampling, and a threshold at which to act on the data.

Each descriptor indicates:

- RX indexes to be scanned.
- TX indexes to be driven.
- Number of RX indexes to scan at one time.
- Number of TX indexes to scan at one time.
- Independent enable for Self (RX-drive), and combination of Self and Mutual modes.
- Channel timing control.
- Oversampling or threshold support.
- Provision to enable an interrupt when complete or when threshold exceeds.
- The Enable mode settings to enable the single, continuous, or continuous until threshold scanning.

The scan descriptors can be set up to run just one time, run continuously until software intervention, or run continuously until the threshold is met.

The four Scan Descriptors enables the software to maintain different scan settings for different regions or resolutions of the panel without having to swap previous description settings. For example: An application can set up Scan Descriptor Zero (SD0) to scan the panel in very large groups 16 RX and 16 TX (PIC32CX-BZ3 family of devices supports maximum 8) at a time. User can configure this Scan Descriptor (SD0) to scan continuously upon each external trigger until its threshold is met, then move on to the next enabled Scan Descriptor. The next scan descriptor can provide a higher resolution scan of only four RX and four TX at a time, with a different oversampling and different threshold. When the application code examines the result of the higher resolution scan, it can program third Scan Descriptor to scan at full resolution and oversampling a small area of the panel to zero-in on exactly where the touch was. Because it can zero-in with the third descriptor, it need not save the settings for the first and second ones, and does not need to re-load them later. The fourth descriptor enables the above to be done for a second simultaneous touch point. The Application code can ping-pong between SD2 and SD3 for as many simultaneous touch points as it supports.

These four descriptors are part of the SFR register:

- CVDSDnC1: CVD Scan Descriptor n Control 1, n = 0-3
- CVDSDnC2: CVD Scan Descriptor n Control 2, n = 0-3
- CVDSDnC3: CVD Scan Descriptor n Control 3, n = 0-3

- CVDSDnT2: CVD Scan Descriptor n Time 2, n = 0-3

33.4.3 Oversampling

A scan descriptor can program each measurement for any amount of oversampling from 0 to 127 additional samples. This accumulates all the samples into a 19-bit result. Do not perform division or averaging to prevent data loss and to enable non-power-of-two oversampling.

33.4.4 Phase Timing

The user can tune the timing of the main 6-phases of the CVD measurement as per application needs. The Scan Descriptor holds time values for each of the following phases of the measurement cycle.

- Charge
- Acquire
- Conversion
- Spacing between polarity measurements
- Spacing between oversampling of same point(s)
- Spacing between channels

33.4.5 Thresholds

Each Scan Descriptor specifies a threshold value. If the application wishes to receive all the accumulated oversample data, it can set the threshold to zero. Otherwise, the Scan Descriptor can be set to only store data that exceeds the requested threshold, or to store all data and assert an interrupt only when the threshold is met.

33.4.6 Interrupts

Configure the enhanced CVD to cause an interrupt when the FIFO exceeds a programmed threshold and/or when a scan descriptor exceeds a threshold and/or completes its function.
Note: FIFO threshold interrupt is not supported in the Standby Sleep mode.

33.4.7 Triggers

The enhanced CVD can be set to start the enabled Scan Descriptors upon two possible trigger events. One is a software trigger SFR bit and another trigger is from any of EVSYS generators. See *Event System (EVSYS)* from Related Links.

If a scan descriptor is set to run in the continuous mode, it runs when it receives a trigger and stays enabled. The CVD module then waits for the next trigger to run the next enabled scan descriptor. Alternatively, the scan descriptor may instruct the CVD module to continue scanning the one descriptor on every trigger until a threshold is met. These modes allow the CVD module to run autonomously being triggered by a system timer while the CPU sleeps (Idle, Standby Sleep), and interrupt the CPU to wake it if a tough threshold is reached. The first mode is useful for scanning various portions of the panel at regular intervals. The second mode is useful for scanning a low-resolution scan on every trigger, then moving to a higher resolution scan descriptor if a threshold is met.

Related Links
26. Event System (EVSYS)

33.4.8 FIFO

Results are stored in a FIFO with a depth of 8 word. The FIFO contains the accumulated result data and the TX and RX indexes for that data. The user can configure the CVD module to store all accumulated results into the FIFO, or only results that exceed the threshold specified in the scan descriptor. In either case, user can generate an interrupt when the FIFO exceeds a watermark value.

The application has the choice of reading out,

- Only a single 32-bit word that contains the RX and TX indexes and the delta of the positive and negative accumulated CVD measurements.
- A total of three 32-bit words to read the actual positive and negative CVD measurement along with delta.

The FIFO uses following registers:

- 33.6.4. CVDRESH: CVD Results POS FIFO Read Register

33.5 Register Summary

See *CVD* module in the *Product Memory Mapping Overview* from Related Links for base address.

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x00	CVDCON	7:0			CLKSEL[1:0]		TRIGSEL[3:0]			
		15:8	FIFOTH[7:0]							
		23:16	THSTR				CVDIEN	FIFOIEN	FIFOTH[9:8]	
		31:24	ON	FRZ	SIDL	ORDER	SDWREN		ABORT	SWTRIG
0x04	CVDADC	7:0					DIGEN	DIFFPEN	SELRES[1:0]	
		15:8								
		23:16								
		31:24								
0x08	CVDSTAT	7:0		SD2INT	SD2DONE	SD2BUSY		SD1INT	SD1DONE	SD1BUSY
		15:8		SD4INT	SD4DONE	SD4BUSY		SD3INT	SD3DONE	SD3BUSY
		23:16	FIFOCNT[7:0]							
		31:24	FIFOFULL	FIFOWM	FIFOMT				FIFOCNT[9:8]	
0x0C ... 0x0F	Reserved									
0x10	CVDRESH	7:0	POS[7:0]							
		15:8	POS[15:8]							
		23:16						POS[18:16]		
		31:24								
0x14	CVDRESL	7:0	NEG[7:0]							
		15:8	NEG[15:8]							
		23:16						NEG[18:16]		
		31:24								
0x18	CVDRESD	7:0	DELTA[7:0]							
		15:8	DELTA[15:8]							
		23:16	RXINDEX[4:0]						DELTA[17:16]	
		31:24	TXINDEX[4:0]						SDNUM[1:0]	
0x1C ... 0x7F	Reserved									
0x80	CVDRX0	7:0	RXAN0[5:0]							
		15:8	RXAN1[5:0]							
		23:16	RXAN2[5:0]							
		31:24	RXAN3[5:0]							
0x84	CVDRX1	7:0	RXAN4[5:0]							
		15:8	RXAN5[5:0]							
		23:16	RXAN6[5:0]							
		31:24	RXAN7[5:0]							
0x88 ... 0xBF	Reserved									
0xC0	CVDTX0	7:0	TXAN0[5:0]							
		15:8	TXAN1[5:0]							
		23:16	TXAN2[5:0]							
		31:24	TXAN3[5:0]							
0xC4	CVDTX1	7:0	TXAN4[5:0]							
		15:8	TXAN5[5:0]							
		23:16	TXAN6[5:0]							
		31:24	TXAN7[5:0]							
0xC8 ... 0xFF	Reserved									

.....continued

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x0100	CVDSD0C1	7:0		SD0OVRSAMP[6:0]						
		15:8	SD0TH[7:0]							
		23:16	SD0TH[15:8]							
		31:24	SD0TH[23:16]							
0x0104	CVDSD0C2	7:0	SD0RXSTRIDE	SD0RXSTRIDE	SD0RXBEG[5:0]					
		15:8	SD0RXSTRIDE	SD0RXSTRIDE	SD0RXEND[5:0]					
		23:16	SD0TXSTRIDE	SD0TXSTRIDE	SD0TXBEG[5:0]					
		31:24	SD0TXSTRIDE	SD0TXSTRIDE	SD0TXEND[5:0]					
0x0108	CVDSD0C3	7:0		SD0CHGTIME[6:0]						
		15:8		SD0ACQTIME[6:0]						
		23:16					CVDEN	CVDCPL[2:0]		
		31:24	SD0EN[1:0]				SD0BUF	SD0INTEN	SD0SELF	SD0MUT
0x010C	CVDSD0T2	7:0		SD0CONTIME[6:0]						
		15:8		SD0POLTIME[6:0]						
		23:16		SD0OVRTIME[6:0]						
		31:24		SD0CHNTIME[6:0]						
0x0110	CVDSD1C1	7:0		SD1OVRSAMP[6:0]						
		15:8	SD1TH[7:0]							
		23:16	SD1TH[15:8]							
		31:24	SD1TH[23:16]							
0x0114	CVDSD1C2	7:0	SD1RXSTRIDE	SD1RXSTRIDE	SD1RXBEG[5:0]					
		15:8	SD1RXSTRIDE	SD1RXSTRIDE	SD1RXEND[5:0]					
		23:16	SD1TXSTRIDE	SD1TXSTRIDE	SD1TXBEG[5:0]					
		31:24	SD1TXSTRIDE	SD1TXSTRIDE	SD1TXEND[5:0]					
0x0118	CVDSD1C3	7:0		SD1CHGTIME[6:0]						
		15:8		SD1ACQTIME[6:0]						
		23:16					CVDEN	CVDCPL[2:0]		
		31:24	SD1EN[1:0]				SD1BUF	SD1INTEN	SD1SELF	SD1MUT
0x011C	CVDSD1T2	7:0		SD1CONTIME[6:0]						
		15:8		SD1POLTIME[6:0]						
		23:16		SD1OVRTIME[6:0]						
		31:24		SD1CHNTIME[6:0]						
0x0120	CVDSD2C1	7:0		SD2OVRSAMP[6:0]						
		15:8	SD2TH[7:0]							
		23:16	SD2TH[15:8]							
		31:24	SD2TH[23:16]							
0x0124	CVDSD2C2	7:0	SD2RXSTRIDE	SD2RXSTRIDE	SD2RXBEG[5:0]					
		15:8	SD2RXSTRIDE	SD2RXSTRIDE	SD2RXEND[5:0]					
		23:16	SD2TXSTRIDE	SD2TXSTRIDE	SD2TXBEG[5:0]					
		31:24	SD2TXSTRIDE	SD2TXSTRIDE	SD2TXEND[5:0]					
0x0128	CVDSD2C3	7:0		SD2CHGTIME[6:0]						
		15:8		SD2ACQTIME[6:0]						
		23:16					CVDEN	CVDCPL[2:0]		
		31:24	SD2EN[1:0]				SD2BUF	SD2INTEN	SD2SELF	SD2MUT

.....continued

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x012C	CVDSD2T2	7:0		SD2CONTIME[6:0]						
		15:8		SD2POLTIME[6:0]						
		23:16		SD2OVRTIME[6:0]						
		31:24		SD2CHNTIME[6:0]						
0x0130	CVDSD3C1	7:0		SD3OVRSAMP[6:0]						
		15:8	SD3TH[7:0]							
		23:16	SD3TH[15:8]							
		31:24	SD3TH[23:16]							
0x0134	CVDSD3C2	7:0	SD3RXSTRIDE	SD3RXSTRIDE	SD3RXBEG[5:0]					
		15:8	SD3RXSTRIDE	SD3RXSTRIDE	SD3RXEND[5:0]					
		23:16	SD3TXSTRIDE	SD3TXSTRIDE	SD3TXBEG[5:0]					
		31:24	SD3TXSTRIDE	SD3TXSTRIDE	SD3TXEND[5:0]					
0x0138	CVDSD3C3	7:0		SD3CHGTIME[6:0]						
		15:8		SD3ACQTIME[6:0]						
		23:16					CVDEN	CVDCPL[2:0]		
		31:24	SD3EN[1:0]				SD3BUF	SD3INTEN	SD3SELF	SD3MUT
0x013C	CVDSD3T2	7:0		SD3CONTIME[6:0]						
		15:8		SD3POLTIME[6:0]						
		23:16		SD3OVRTIME[6:0]						
		31:24		SD3CHNTIME[6:0]						

Related Links

7. Product Memory Mapping Overview

33.6 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.

Optional write protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write Protection" property in each individual register description.

See *Peripheral Access Controller (PAC)* from Related Links.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable-protection is denoted by the Enable-protected property in each individual register description.

Following conventions are used in the register description:

- – R = Readable bit
- – W = Writable bit
- – U = Unimplemented bit, read as '0'
- – -n = Value at POR
- – '1' = Bit is set
- – '0' = Bit is cleared
- – x = Bit is unknown
- HS = Hardware Set
- HC = Hardware Cleared

Related Links

20. Peripheral Access Controller (PAC)

33.6.1 CVDCON - CVD Control Register

Name: CVDCON
Offset: 0x00
Reset: 0x00020000
Property: -

Bit	31	30	29	28	27	26	25	24
	ON	FRZ	SIDL	ORDER	SDWREN		ABORT	SWTRIG
Access	R/W	R/W	R/W	R/W	R/W		W/HC	W/HC
Reset	0	0	0	0	0		0	0

Bit	23	22	21	20	19	18	17	16
	THSTR				CVDIEN	FIFOIEN	FIFOTH[9:8]	
Access	R/W				R/W	R/W	R/W	R/W
Reset	0				0	0	1	0

Bit	15	14	13	12	11	10	9	8
	FIFOTH[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	7	6	5	4	3	2	1	0
			CLKSEL[1:0]		TRIGSEL[3:0]			
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit 31 – ON Enables the State Machine to scan the enabled Scan Descriptors upon next trigger
Before turning ON bit from 1'b1 to 1'b0, the Scan Enable bits of all descriptors must be cleared and the CVD controller must either be allowed to finish any scan in progress, or must be instructed to abort the scan with the ABORT bit.

Bit 30 – FRZ Freeze Mode

Value	Description
1	CVD controller stops in the debugger mode
0	CVD controller runs in debugger mode

Bit 29 – SIDL Stop in Idle Mode bit

Value	Description
1	CVD controller stops when device enters Idle mode
0	CVD controller continues running in Idle mode

Bit 28 – ORDER RX/TX Loop Order

Value	Description
1	Scans all the requested TX indexes, then increments RX index and continues operation
0	Scans all the requested RX indexes, then increments TX index and continues operation

Bit 27 – SDWREN Scan Descriptor Write Enable

Value	Description
1	Enables writes to the scan descriptors
0	Prevents writes to the scan descriptors

Bit 25 – ABORT Abort Current Scan
Note: The controller moves on to the next enabled Scan Descriptor if there is one, else it goes to idle state. Hardware clears this bit.

Value	Description
1	Aborts the current scan
0	CVD controller continues with the current scan

Bit 24 – SWTRIG Software Trigger control. Starts scan manually

Note: Hardware clears this bit.

Value	Description
1	Starts scan manually
0	Continues without the scan

Bit 23 – THSTR Threshold Store Mode

Value	Description
1	Stores only the results which exceed the programmed threshold for the Scan Descriptor
0	Stores all the results in FIFO

Bit 19 – CVDIEN Global Interrupt Enable

Value	Description
1	Enables the FIFO and scan descriptor interrupts
0	Disables the FIFO and scan descriptor interrupts

Bit 18 – FIFOIEN FIFO Threshold Interrupt Enable

Value	Description
1	Controller asserts an interrupt when the FIFO threshold is met
0	Controller does not assert an interrupt when the FIFO threshold is met

Bits 17:8 – FIFOTH[9:0] Threshold for the results FIFO

These bits contain threshold for the results FIFO that causes an interrupt and watermark FIFOWM status bit assertion.

Bits 5:4 – CLKSEL[1:0] Clock Select for CVD

Value	Description
00	PB1_CLK
01	POSC
02	LPRC
03	Reserved

Bits 3:0 – TRIGSEL[3:0] Trigger select for starting the scan

Value	Description
0000	SFR controlled software trigger
0001	EVSYS event. See *Event System (EVSYS)* from Related Links.
...	
...	
...	
1111	Reserved

Related Links

26. Event System (EVSYS)

33.6.2 CVD ADC Configuration Register

Name: CVDADC
Offset: 0x04
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
					DIGEN	DIFFPEN	SELRES[1:0]	
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit 3 – DIGEN

Shared ADC Digital Enable bit (Differential Mode Select from ADC controller)

Value	Description
1	Shared ADC is digital enabled
0	Shared ADC is digital disabled

Bit 2 – DIFFPEN

Controls differential mode operation of ANN0.

Value	Description
1	ANN0 (Differential) enabled
0	ANN0 (Differential) disabled

Bits 1:0 – SELRES[1:0] Shared ADC Resolution bits

Read as '0'.

Note: Changing the resolution of the ADC does not shift the result in the corresponding ADCDATAx register. The result will still occupy 12 bits, with the corresponding lower unused bits set to '0'. For example, a resolution of 6 bits will result in ADCDATAx[5:0] being set to '0', and ADCDATAx[11:6] holding the result.

Value	Description
00	6 bits
01	8 bits
10	10 bits
11	12 bits (default)

33.6.3 CVD Status Register

Name: CVDSTAT
Offset: 0x08
Reset: 0x20000000
Property: -

Bit	31	30	29	28	27	26	25	24
	FIFOFULL	FIFOWM	FIFOMT				FIFOCNT[9:8]	
Access	R	R	R				R	R
Reset	0	0	1				0	0

Bit	23	22	21	20	19	18	17	16
	FIFOCNT[7:0]							
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
		SD4INT	SD4DONE	SD4BUSY		SD3INT	SD3DONE	SD3BUSY
Access		R/W/HS	R	R		R/W/HS	R	R
Reset		0	0	0		0	0	0

Bit	7	6	5	4	3	2	1	0
		SD2INT	SD2DONE	SD2BUSY		SD1INT	SD1DONE	SD1BUSY
Access		R/W/HS	R	R		R/W/HS	R	R
Reset		0	0	0		0	0	0

Bit 31 – FIFOFULL Results FIFO is full

Value	Description
1	FIFO is full
0	Not full

Bit 30 – FIFOWM

Value	Description
1	FIFO has reached the programmed FIFOTHRESH threshold
0	FIFO has not reached the programmed FIFOTHRESH threshold

Bit 29 – FIFOMT

Value	Description
1	FIFO is empty
0	FIFO is not empty

Bits 25:16 – FIFOCNT[9:0]
These bits indicate the number of words in the Results FIFO.

Bit 14 – SD4INT

Value	Description
1	Scan Descriptor 4 has caused an interrupt
0	Scan Descriptor 4 has not caused an interrupt

Bit 13 – SD4DONE
Note: The hardware will clear this bit upon receiving next trigger for Scan Descriptor 4.

Value	Description
1	Scan Descriptor 4 has completed at least once
0	Scan Descriptor 4 has not completed

Bit 12 – SD4BUSY

Value	Description
1	Scan Descriptor 4 is in progress
0	Scan Descriptor 4 is not in progress

Bit 10 – SD3INT

Value	Description
1	Scan Descriptor 3 has caused an interrupt
0	Scan Descriptor 3 has not caused an interrupt

Bit 9 – SD3DONE

The controller sets this bit if Scan Descriptor 3 completes at least once. Core clears this bit upon receiving next trigger for Scan Descriptor 3.

Bit 8 – SD3BUSY

Value	Description
1	Scan Descriptor 3 is in progress
0	Scan Descriptor 3 is not in progress

Bit 6 – SD2INT

Value	Description
1	Scan Descriptor 2 has caused an interrupt
0	Scan Descriptor 2 has not caused an interrupt

Bit 5 – SD2DONE The controller sets this bit if Scan Descriptor 2 completes at least once. Core clears this bit upon receiving next trigger for Scan Descriptor 2.

Bit 4 – SD2BUSY

Value	Description
1	Scan Descriptor 2 has caused an interrupt
0	Scan Descriptor 2 has not caused an interrupt

Bit 2 – SD1INT Scan Descriptor 1 has caused an interrupt

Value	Description
1	Scan Descriptor 1 has caused an interrupt
0	Scan Descriptor 1 has not caused an interrupt

Bit 1 – SD1DONE The controller sets this bit if Scan Descriptor 1 has completed at least once. Core clears this bit upon receiving next trigger for Scan Descriptor 1.

Bit 0 – SD1BUSY Scan Descriptor 1 is in progress

Value	Description
1	Scan Descriptor 1 is in progress
0	Scan Descriptor 1 is not in progress

33.6.4 CVD Results POS FIFO Read Register

Name: CVDRESH
Offset: 0x010
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

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

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

Bits 18:0 – POS[18:0]

The accumulated result of the positive-side measurements since the controller supports up to 128x oversampling, each polarity can accumulate up to 19 bits when using 12-bit ADC (Actual number of bits = ADCBITS + 7). The accumulation is not shifted back down to create an average. Therefore, if oversampling was requested, the software will need to account for the left-shift of the result returned.

33.6.5 CVD Results NEG FIFO Read Register

Name: CVDRESL
Offset: 0x014
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

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

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

Bits 18:0 – NEG[18:0]

The accumulated result of the negative-side measurements since the controller supports up to 128x oversampling, each polarity can accumulate up to 18 bits when using 12-bit ADC (Actual number of bits = ADCBITS + 7). The accumulation is not shifted back down to create an average. Therefore, if oversampling was requested, the software will need to account for the left-shift of the result returned.

33.6.6 CVD Results Descriptor FIFO Read Register

Name: CVDRESD
Offset: 0x018
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
	TXINDEX[4:0]						SDNUM[1:0]	
Access	R	R	R	R	R		R	R
Reset	0	0	0	0	0		0	0

Bit	23	22	21	20	19	18	17	16
	RXINDEX[4:0]						DELTA[17:16]	
Access	R	R	R	R	R		R	R
Reset	0	0	0	0	0		0	0

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

Bits 31:27 – TXINDEX[4:0] Transmit Index of the result
If the Stride of the Scan Descriptor is more than one, the Transmit Index indicates the first one of the group.

Bits 25:24 – SDNUM[1:0] Scan Descriptor Number
These bits has the Scan Descriptor Number that generated the result.

Bits 23:19 – RXINDEX[4:0] Receive Index of the result
If the Stride of the Scan Descriptor is more than one, the Receive Index indicates the first one of the group.

Bits 17:0 – DELTA[17:0] Delta of the accumulated results of the negative-side and positive-side measurements
Since the controller supports up to 128x oversampling, each polarity can accumulate up to 19 bits when using 12-bit ADC. The accumulation is not shifted back down to create an average. Therefore, if oversampling was requested, the software will need to account for the results returned.
The DELTA presented is the 18 MSBs (including sign bit) of a signed subtraction of the two accumulators. The data will be in 2's complement form if the delta is negative.
Width Needed: ADCBITS + 7 + 1 (sign)
Width available: 18
delta_pre[ADCBITS+7:0] = signed'({1'b0, POS[ADCBITS+7-1:0]} - {1'b0, NEG[ADCBITS+7-1:0]})
DELTA[17:0] = delta_pre[ADC_BITS+7:ADC_BITS+7-17]
Note: Reading this register increments the FIFO read pointer, destroying the data in the previous two registers for NEG and POS absolute values. If the NEG and POS values are desired, those registers must be read before this register. If the absolute values are not required, bandwidth can be saved by reading only this descriptor register.

33.6.7 CVD Receive Index 0 Configuration

Name: CVDRX0
Offset: 0x080
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
			RXAN3[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
			RXAN2[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
			RXAN1[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
			RXAN0[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bits 29:24 – RXAN3[5:0] ANx/CVDR channel to use for RX Index 3

Bits 21:16 – RXAN2[5:0] ANx/CVDR channel to use for RX Index 2

Bits 13:8 – RXAN1[5:0] ANx/CVDR channel to use for RX Index 1

Bits 5:0 – RXAN0[5:0] ANx/CVDR channel to use for RX Index 0

33.6.8 CVD Receive Index 1 Configuration

Name: CVDRX1
Offset: 0x084
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
			RXAN7[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
			RXAN6[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
			RXAN5[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
			RXAN4[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bits 29:24 – RXAN7[5:0] ANx/CVDR channel to use for RX Index 7

Bits 21:16 – RXAN6[5:0] ANx/CVDR channel to use for RX Index 6

Bits 13:8 – RXAN5[5:0] ANx/CVDR channel to use for RX Index 5

Bits 5:0 – RXAN4[5:0] ANx/CVDR channel to use for RX Index 4

33.6.9 CVD Transmit Index 0 Configuration

Name: CVDTX0
Offset: 0x0C0
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
			TXAN3[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
			TXAN2[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
			TXAN1[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
			TXAN0[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bits 29:24 – TXAN3[5:0] ANx/CVDR channel to use for TX Index 3

Bits 21:16 – TXAN2[5:0] ANx/CVDR channel to use for TX Index 2

Bits 13:8 – TXAN1[5:0] ANx/CVDR channel to use for TX Index 1

Bits 5:0 – TXAN0[5:0] ANx/CVDR channel to use for TX Index 0

33.6.10 CVD Transmit Index 1 Configuration

Name: CVDTX1
Offset: 0x0C4
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
			TXAN7[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
			TXAN6[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
			TXAN5[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
			TXAN4[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bits 29:24 – TXAN7[5:0] ANx/CVDR channel to use for TX Index 7

Bits 21:16 – TXAN6[5:0] ANx/CVDR channel to use for TX Index 6

Bits 13:8 – TXAN5[5:0] ANx/CVDR channel to use for TX Index 5

Bits 5:0 – TXAN4[5:0] ANx/CVDR channel to use for TX Index 4

33.6.11 CVD Scan Descriptor 0 Control 1

Name: CVDSD0C1
Offset: 0x100
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
	SD0TH[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	23	22	21	20	19	18	17	16
	SD0TH[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	15	14	13	12	11	10	9	8
	SD0TH[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	7	6	5	4	3	2	1	0
		SD0OVRSAMP[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 31:8 – SD0TH[23:0] Scan Descriptor Threshold
The controller subtracts the accumulators after all oversampling is complete. The controller compares the result of subtraction to this threshold to determine if threshold exceed asserts internally, which is used to generate an interrupt and/or store data to the FIFO, depending on settings of other control bits.

Bits 6:0 – SD0OVRSAMP[6:0] Scan Descriptor Over Sampling
Determines the amount of oversampling done on each measurement.

Value	Description
0	Accumulates one measurement
1	Accumulates two measurements
...	
127	Accumulates 128 measurements

33.6.12 CVD Scan Descriptor 0 Control 2

Name: CVDSD0C2
Offset: 0x104
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
	SD0TXSTRIDE	SD0TXSTRIDE	SD0TXEND[5:0]					
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	23	22	21	20	19	18	17	16
	SD0TXSTRIDE	SD0TXSTRIDE	SD0TXBEG[5:0]					
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
	SD0RXSTRIDE	SD0RXSTRIDE	SD0RXEND[5:0]					
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	7	6	5	4	3	2	1	0
	SD0RXSTRIDE	SD0RXSTRIDE	SD0RXBEG[5:0]					
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 29:24 – SD0TXEND[5:0] Scan Descriptor TX Index End
Determines the last TX index to include in a scan. The SD0TXSTRIDE+1 value increments the TX index pointer, meets or exceeds this value, the TX loop of the scan is complete.

Bits 22,23,30,31 – SD0TXSTRIDE Scan Descriptor TX Index Stride
Determines the number of TX Indexes included in a single measurement.

Value	Description
4'b0	One TX Index
4'bF	16 TX Indexes

Bits 21:16 – SD0TXBEG[5:0] Scan Descriptor TX Index Start
Determines the first TX index to include in a scan.

Bits 13:8 – SD0RXEND[5:0] Scan Descriptor RX Index End
Determines the last RX index to include in a scan. The SD0RXSTRIDE+1 value increments the RX index pointer, meets or exceeds this value, the RX loop of the scan is complete.

Bits 6,7,14,15 – SD0RXSTRIDE Scan Descriptor RX Index Stride
Determines the number of RX indexes included in a single measurement.

Value	Description
4'b0	One RX Index
4'bF	16 RX Indexes

Bits 5:0 – SD0RXBEG[5:0] Scan Descriptor RX Index Start
Determines the first RX index to include in a scan.

33.6.13 CVD Scan Descriptor 0 Control 3

Name: CVDSD0C3
Offset: 0x108
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
	SD0EN[1:0]				SD0BUF	SD0INTEN	SD0SELF	SD0MUT
Access	R/W/HC	R/W/HC			R/W	R/W	R/W	R/W
Reset	0	0			0	0	0	0

Bit	23	22	21	20	19	18	17	16
					CVDEN	CVDCPL[2:0]		
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	15	14	13	12	11	10	9	8
		SD0ACQTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
		SD0CHGTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 31:30 – SD0EN[1:0] Scan Descriptor Enable Mode

Value	Description
00	Disables the Scan Descriptor.
01	Executes the Scan Descriptor one time only, then clears the enable.
10	Executes the Scan Descriptor, and keeps it enabled. Then, moves on to the next enabled descriptors.
11	Executes the Scan Descriptor in a loop until a threshold match is detected, then clears the enable mode and moves on to next enabled descriptors.

Bit 27 – SD0BUF Scan Descriptor CVD Buffer Enable

Value	Description
1	Uses CVD buffer output as shared ADC input.
0	Does not use CVD buffer output as shared ADC input.

Bit 26 – SD0INTEN Scan Descriptor Interrupt Enable

Value	Description
1	Scan Descriptor creates an interrupt if the accumulator threshold is met.
0	Scan descriptor does not create an interrupt.

Bit 25 – SD0SELF Scan Descriptor Self Measurement Mode

Value	Description
1	Enables the Self Measurement mode; RX outputs are part of CVD measurement and are driven.
0	Disables the Self Measurement mode; RX outputs are not part of CVD measurements.

Bit 24 – SD0MUT Scan Descriptor Mutual Mode

Value	Description
1	Enables the Mutual Measurement mode; TX outputs are part of CVD measurement and are driven.
0	Disables the Mutual Measurement mode; TX outputs are not part of CVD measurements.

Bit 19 – CVDEN Capacitive Voltage Division Enable bit

Value	Description
1	CVD operation is enabled
0	CVD operation is disabled

Bits 18:16 – CVDCPL[2:0] Capacitor Voltage Divider (CVD) Setting bits

Value	Description
111	7 * 2.5 pF = 17.5 pF
110	6 * 2.5 pF = 15 pF
101	5 * 2.5 pF = 12.5 pF
100	4 * 2.5 pF = 10 pF
011	3 * 2.5 pF = 7.5 pF
010	2 * 2.5 pF = 5 pF
001	1 * 2.5 pF = 2.5 pF
000	0 * 2.5 pF = 0 pF

Bits 14:8 – SD0ACQTIME[6:0] Scan Descriptor Acquire Time
Specifies the time for which CVD waits for ADC voltage to settle.

Bits 6:0 – SD0CHGTIME[6:0] Scan Descriptor Charge Time
Specifies the time for which CVD remains in the charging state for internal or external capacitors, and for TX outputs.

33.6.14 CVD Scan Descriptor 0 Time 2

Name: CVDSD0T2
Offset: 0x10C
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
		SD0CHNTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
		SD0OVRTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
		SD0POLTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
		SD0CONTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 30:24 – SD0CHNTIME[6:0] Scan Descriptor Channel Time
Controls the number of cycles the state machine waits in the RXCHAN or TXCHAN state before moving to the next RX/TX pair.

Bits 22:16 – SD0OVRTIME[6:0] Scan Descriptor Oversample Time
Controls the number of cycles the state machine waits in the OVERSAMP state before taking the next oversampling measurement of an RX/TX pair.

Bits 14:8 – SD0POLTIME[6:0] Scan Descriptor Polarity Time
Controls the number of cycles the state machine waits in the POLARITY state before taking the second polarity measurement of an RX/TX pair.

Bits 6:0 – SD0CONTIME[6:0] Scan Descriptor Convert Time
Controls the number of cycles the state machine waits in the CONVERT state for the ADC sample data. User must ensure that the ADC asserts End-Of-Convert (EOC) before the CONVERT state timer expires.

33.6.15 CVD Scan Descriptor 1 Control 1

Name: CVDSD1C1
Offset: 0x110
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
	SD1TH[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	23	22	21	20	19	18	17	16
	SD1TH[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	15	14	13	12	11	10	9	8
	SD1TH[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	7	6	5	4	3	2	1	0
		SD1OVRSAMP[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 31:8 – SD1TH[23:0] Scan Descriptor Threshold
The controller subtracts the accumulators after all oversampling is complete. The controller compares the result of subtraction to this threshold to determine if threshold exceed asserts internally, which is used to generate an interrupt and/or store data to the FIFO, depending on settings of other control bits.

Bits 6:0 – SD1OVRSAMP[6:0] Scan Descriptor Over Sampling
Determines the amount of oversampling done on each measurement.

Value	Description
0	Accumulates one measurement
1	Accumulates two measurements
...	
127	Accumulates 128 measurements

33.6.16 CVD Scan Descriptor 1 Control 2

Name: CVDSD1C2
Offset: 0x114
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
	SD1TXSTRIDE	SD1TXSTRIDE	SD1TXEND[5:0]					
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	23	22	21	20	19	18	17	16
	SD1TXSTRIDE	SD1TXSTRIDE	SD1TXBEG[5:0]					
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
	SD1RXSTRIDE	SD1RXSTRIDE	SD1RXEND[5:0]					
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	7	6	5	4	3	2	1	0
	SD1RXSTRIDE	SD1RXSTRIDE	SD1RXBEG[5:0]					
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 29:24 – SD1TXEND[5:0] Scan Descriptor TX Index End
Determines the last TX index to include in a scan. The SD1TXSTRIDE+1 value increments the TX index pointer, meets or exceeds this value, the TX loop of the scan is complete.

Bits 22,23,30,31 – SD1TXSTRIDE Scan Descriptor TX Index Stride
Determines the number of TX Indexes included in a single measurement.

Value	Description
4'b0	One TX Index
4'bF	16 TX Indexes

Bits 21:16 – SD1TXBEG[5:0] Scan Descriptor TX Index Start
Determines the first TX index to include in a scan.

Bits 13:8 – SD1RXEND[5:0] Scan Descriptor RX Index End
Determines the last RX index to include in a scan. The SD1RXSTRIDE+1 value increments the RX index pointer, meets or exceeds this value, the RX loop of the scan is complete.

Bits 6,7,14,15 – SD1RXSTRIDE Scan Descriptor RX Index Stride
Determines the number of RX indexes included in a single measurement.

Value	Description
4'b0	One RX Index
4'bF	16 RX Indexes

Bits 5:0 – SD1RXBEG[5:0] Scan Descriptor RX Index Start
Determines the first RX index to include in a scan.

33.6.17 CVD Scan Descriptor 1 Control 3

Name: CVDSD1C3
Offset: 0x118
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
	SD1EN[1:0]				SD1BUF	SD1INTEN	SD1SELF	SD1MUT
Access	R/W/HC	R/W/HC			R/W	R/W	R/W	R/W
Reset	0	0			0	0	0	0

Bit	23	22	21	20	19	18	17	16
					CVDEN	CVDCPL[2:0]		
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	15	14	13	12	11	10	9	8
		SD1ACQTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
		SD1CHGTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 31:30 – SD1EN[1:0] Scan Descriptor Enable Mode

Value	Description
00	Disables the Scan Descriptor.
01	Executes the Scan Descriptor one time only, then clears the enable.
10	Executes the Scan Descriptor, and keeps it enabled. Then, moves on to the next enabled descriptors.
11	Executes the Scan Descriptor in a loop until a threshold match is detected, then clears the enable mode and moves on to next enabled descriptors.

Bit 27 – SD1BUF Scan Descriptor CVD Buffer Enable

Value	Description
1	Uses CVD buffer output as shared ADC (ADC2) input.
0	Does not use CVD buffer output as shared ADC (ADC2) input.

Bit 26 – SD1INTEN Scan Descriptor Interrupt Enable

Value	Description
1	Scan Descriptor creates an interrupt if the accumulator threshold is met.
0	Scan descriptor does not create an interrupt.

Bit 25 – SD1SELF Scan Descriptor Self Measurement Mode

Value	Description
1	Enables the Self Measurement mode; RX outputs are part of CVD measurement and are driven.
0	Disables the Self Measurement mode; RX outputs are not part of CVD measurements.

Bit 24 – SD1MUT Scan Descriptor Mutual Mode

Value	Description
1	Enables the Mutual Measurement mode; TX outputs are part of CVD measurement and are driven.
0	Disables the Mutual Measurement mode; TX outputs are not part of CVD measurements.

Bit 19 – CVDEN Capacitive Voltage Division Enable bit

Value	Description
1	CVD operation is enabled
0	CVD operation is disabled

Bits 18:16 – CVDCPL[2:0] Capacitor Voltage Divider (CVD) Setting bits

Value	Description
111	7 * 2.5 pF = 17.5 pF
110	6 * 2.5 pF = 15 pF
101	5 * 2.5 pF = 12.5 pF
100	4 * 2.5 pF = 10 pF
011	3 * 2.5 pF = 7.5 pF
010	2 * 2.5 pF = 5 pF
001	1 * 2.5 pF = 2.5 pF
000	0 * 2.5 pF = 0 pF

Bits 14:8 – SD1ACQTIME[6:0] Scan Descriptor Acquire Time
Specifies the time for which CVD waits for ADC voltage to settle.

Bits 6:0 – SD1CHGTIME[6:0] Scan Descriptor Charge Time
Specifies the time for which CVD remains in the charging state for internal or external capacitors, and for TX outputs.

33.6.18 CVD Scan Descriptor 1 Time 2

Name: CVDSD1T2
Offset: 0x11C
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
		SD1CHNTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
		SD1OVRTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
		SD1POLTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
		SD1CONTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 30:24 – SD1CHNTIME[6:0] Scan Descriptor Channel Time
Controls the number of cycles the state machine waits in the RXCHAN or TXCHAN state before moving to the next RX/TX pair.

Bits 22:16 – SD1OVRTIME[6:0] Scan Descriptor Oversample Time
Controls the number of cycles the state machine waits in the OVERSAMP state before taking the next oversampling measurement of an RX/TX pair.

Bits 14:8 – SD1POLTIME[6:0] Scan Descriptor Polarity Time
Controls the number of cycles the state machine waits in the POLARITY state before taking the second polarity measurement of an RX/TX pair.

Bits 6:0 – SD1CONTIME[6:0] Scan Descriptor Convert Time
Controls the number of cycles the state machine waits in the CONVERT state for the ADC sample data. User must ensure that the ADC asserts End-Of-Convert (EOC) before the CONVERT state timer expires.

33.6.19 CVD Scan Descriptor 2 Control 1

Name: CVDSD2C1
Offset: 0x120
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
	SD2TH[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	23	22	21	20	19	18	17	16
	SD2TH[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	15	14	13	12	11	10	9	8
	SD2TH[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	7	6	5	4	3	2	1	0
		SD2OVRSAMP[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 31:8 – SD2TH[23:0] Scan Descriptor Threshold
The controller subtracts the accumulators after all oversampling is complete. The controller compares the result of subtraction to this threshold to determine if threshold exceed asserts internally, which is used to generate an interrupt and/or store data to the FIFO, depending on settings of other control bits.

Bits 6:0 – SD2OVRSAMP[6:0] Scan Descriptor Over Sampling
Determines the amount of oversampling done on each measurement.

Value	Description
0	Accumulates one measurement
1	Accumulates two measurements
...	
127	Accumulates 128 measurements

33.6.20 CVD Scan Descriptor 2 Control 2

Name: CVDSD2C2
Offset: 0x124
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
	SD2TXSTRIDE	SD2TXSTRIDE	SD2TXEND[5:0]					
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	23	22	21	20	19	18	17	16
	SD2TXSTRIDE	SD2TXSTRIDE	SD2TXBEG[5:0]					
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
	SD2RXSTRIDE	SD2RXSTRIDE	SD2RXEND[5:0]					
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	7	6	5	4	3	2	1	0
	SD2RXSTRIDE	SD2RXSTRIDE	SD2RXBEG[5:0]					
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 29:24 – SD2TXEND[5:0] Scan Descriptor TX Index End
Determines the last TX index to include in a scan. The SD2TXSTRIDE+1 value increments the TX index pointer, meets or exceeds this value, the TX loop of the scan is complete.

Bits 22,23,30,31 – SD2TXSTRIDE Scan Descriptor TX Index Stride
Determines the number of TX Indexes included in a single measurement.

Value	Description
4'b0	One TX Index
4'bF	16 TX Indexes

Bits 21:16 – SD2TXBEG[5:0] Scan Descriptor TX Index Start
Determines the first TX index to include in a scan.

Bits 13:8 – SD2RXEND[5:0] Scan Descriptor RX Index End
Determines the last RX index to include in a scan. The SD2RXSTRIDE+1 value increments the RX index pointer, meets or exceeds this value, the RX loop of the scan is complete.

Bits 6,7,14,15 – SD2RXSTRIDE Scan Descriptor RX Index Stride
Determines the number of RX indexes included in a single measurement.

Value	Description
4'b0	One RX Index
4'bF	16 RX Indexes

Bits 5:0 – SD2RXBEG[5:0] Scan Descriptor RX Index Start
Determines the first RX index to include in a scan.

33.6.21 CVD Scan Descriptor 2 Control 3

Name: CVDSD2C3
Offset: 0x128
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
	SD2EN[1:0]				SD2BUF	SD2INTEN	SD2SELF	SD2MUT
Access	R/W/HC	R/W/HC			R/W	R/W	R/W	R/W
Reset	0	0			0	0	0	0

Bit	23	22	21	20	19	18	17	16
					CVDEN	CVDCPL[2:0]		
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	15	14	13	12	11	10	9	8
		SD2ACQTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
		SD2CHGTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 31:30 – SD2EN[1:0] Scan Descriptor Enable Mode

Value	Description
00	Disables the Scan Descriptor.
01	Executes the Scan Descriptor one time only, then clears the enable.
10	Executes the Scan Descriptor, and keeps it enabled. Then, moves on to the next enabled descriptors.
11	Executes the Scan Descriptor in a loop until a threshold match is detected, then clears the enable mode and moves on to next enabled descriptors.

Bit 27 – SD2BUF Scan Descriptor CVD Buffer Enable

Value	Description
1	Uses CVD buffer output as shared ADC (ADC2) input.
0	Does not use CVD buffer output as shared ADC (ADC2) input.

Bit 26 – SD2INTEN Scan Descriptor Interrupt Enable

Value	Description
1	Scan Descriptor creates an interrupt if the accumulator threshold is met.
0	Scan descriptor does not create an interrupt.

Bit 25 – SD2SELF Scan Descriptor Self Measurement Mode

Value	Description
1	Enables the Self Measurement mode; RX outputs are part of CVD measurement and are driven.
0	Disables the Self Measurement mode; RX outputs are not part of CVD measurements.

Bit 24 – SD2MUT Scan Descriptor Mutual Mode

Value	Description
1	Enables the Mutual Measurement mode; TX outputs are part of CVD measurement and are driven.
0	Disables the Mutual Measurement mode; TX outputs are not part of CVD measurements.

Bit 19 – CVDEN Capacitive Voltage Division Enable bit

Value	Description
1	CVD operation is enabled
0	CVD operation is disabled

Bits 18:16 – CVDCPL[2:0] Capacitor Voltage Divider (CVD) Setting bits

Value	Description
111	7 * 2.5 pF = 17.5 pF
110	6 * 2.5 pF = 15 pF
101	5 * 2.5 pF = 12.5 pF
100	4 * 2.5 pF = 10 pF
011	3 * 2.5 pF = 7.5 pF
010	2 * 2.5 pF = 5 pF
001	1 * 2.5 pF = 2.5 pF
000	0 * 2.5 pF = 0 pF

Bits 14:8 – SD2ACQTIME[6:0] Scan Descriptor Acquire Time
Specifies the time for which CVD waits for ADC voltage to settle.

Bits 6:0 – SD2CHGTIME[6:0] Scan Descriptor Charge Time
Specifies the time for which CVD remains in the charging state for internal or external capacitors, and for TX outputs.

33.6.22 CVD Scan Descriptor 2 Time 2

Name: CVDSD2T2
Offset: 0x12C
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
		SD2CHNTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
		SD2OVRTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
		SD2POLTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
		SD2CONTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 30:24 – SD2CHNTIME[6:0] Scan Descriptor Channel Time
Controls the number of cycles the state machine waits in the RXCHAN or TXCHAN state before moving to the next RX/TX pair.

Bits 22:16 – SD2OVRTIME[6:0] Scan Descriptor Oversample Time
Controls the number of cycles the state machine waits in the OVERSAMP state before taking the next oversampling measurement of an RX/TX pair.

Bits 14:8 – SD2POLTIME[6:0] Scan Descriptor Polarity Time
Controls the number of cycles the state machine waits in the POLARITY state before taking the second polarity measurement of an RX/TX pair.

Bits 6:0 – SD2CONTIME[6:0] Scan Descriptor Convert Time
Controls the number of cycles the state machine waits in the CONVERT state for the ADC sample data. User must ensure that the ADC asserts End-Of-Convert (EOC) before the CONVERT state timer expires.

33.6.23 CVD Scan Descriptor 3 Control 1

Name: CVDSD3C1
Offset: 0x130
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
	SD3TH[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	23	22	21	20	19	18	17	16
	SD3TH[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	15	14	13	12	11	10	9	8
	SD3TH[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	7	6	5	4	3	2	1	0
		SD3OVRSAMP[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 31:8 – SD3TH[23:0] Scan Descriptor Threshold
The controller subtracts the accumulators after all oversampling is complete. The controller compares the result of subtraction to this threshold to determine if threshold exceed asserts internally, which is used to generate an interrupt and/or store data to the FIFO, depending on settings of other control bits.

Bits 6:0 – SD3OVRSAMP[6:0] Scan Descriptor Over Sampling
Determines the amount of oversampling done on each measurement.

Value	Description
0	Accumulates one measurement
1	Accumulates two measurements
...	
127	Accumulates 128 measurements

33.6.24 CVD Scan Descriptor 3 Control 2

Name: CVDSD3C2
Offset: 0x134
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
	SD3TXSTRIDE	SD3TXSTRIDE	SD3TXEND[5:0]					
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	23	22	21	20	19	18	17	16
	SD3TXSTRIDE	SD3TXSTRIDE	SD3TXBEG[5:0]					
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
	SD3RXSTRIDE	SD3RXSTRIDE	SD3RXEND[5:0]					
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	7	6	5	4	3	2	1	0
	SD3RXSTRIDE	SD3RXSTRIDE	SD3RXBEG[5:0]					
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 29:24 – SD3TXEND[5:0] Scan Descriptor TX Index End
Determines the last TX index to include in a scan. The SD3TXSTRIDE+1 value increments the TX index pointer, meets or exceeds this value, the TX loop of the scan is complete.

Bits 22,23,30,31 – SD3TXSTRIDE Scan Descriptor TX Index Stride
Determines the number of TX Indexes included in a single measurement.

Value	Description
4'b0	One TX Index
4'bF	16 TX Indexes

Bits 21:16 – SD3TXBEG[5:0] Scan Descriptor TX Index Start
Determines the first TX index to include in a scan.

Bits 13:8 – SD3RXEND[5:0] Scan Descriptor RX Index End
Determines the last RX index to include in a scan. The SD3RXSTRIDE+1 value increments the RX index pointer, meets or exceeds this value, the RX loop of the scan is complete.

Bits 6,7,14,15 – SD3RXSTRIDE Scan Descriptor RX Index Stride
Determines the number of RX indexes included in a single measurement.

Value	Description
4'b0	One RX Index
4'bF	16 RX Indexes

Bits 5:0 – SD3RXBEG[5:0] Scan Descriptor RX Index Start
Determines the first RX index to include in a scan.

33.6.25 CVD Scan Descriptor 3 Control 3

Name: CVDSD3C3
Offset: 0x138
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
	SD3EN[1:0]				SD3BUF	SD3INTEN	SD3SELF	SD3MUT
Access	R/W/HC	R/W/HC			R/W	R/W	R/W	R/W
Reset	0	0			0	0	0	0

Bit	23	22	21	20	19	18	17	16
					CVDEN	CVDCPL[2:0]		
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	15	14	13	12	11	10	9	8
		SD3ACQTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
		SD3CHGTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 31:30 – SD3EN[1:0] Scan Descriptor Enable Mode

Value	Description
00	Disables the Scan Descriptor.
01	Executes the Scan Descriptor one time only, then clears the enable.
10	Executes the Scan Descriptor, and keeps it enabled. Then, moves on to the next enabled descriptors.
11	Executes the Scan Descriptor in a loop until a threshold match is detected, then clears the enable mode and moves on to next enabled descriptors.

Bit 27 – SD3BUF Scan Descriptor CVD Buffer Enable

Value	Description
1	Uses CVD buffer output as shared ADC (ADC2) input.
0	Does not use CVD buffer output as shared ADC (ADC2) input.

Bit 26 – SD3INTEN Scan Descriptor Interrupt Enable

Value	Description
1	Scan Descriptor creates an interrupt if the accumulator threshold is met.
0	Scan descriptor does not create an interrupt.

Bit 25 – SD3SELF Scan Descriptor Self Measurement Mode

Value	Description
1	Enables the Self Measurement mode; RX outputs are part of CVD measurement and are driven.
0	Disables the Self Measurement mode; RX outputs are not part of CVD measurements.

Bit 24 – SD3MUT Scan Descriptor Mutual Mode

Value	Description
1	Enables the Mutual Measurement mode; TX outputs are part of CVD measurement and are driven.
0	Disables the Mutual Measurement mode; TX outputs are not part of CVD measurements.

Bit 19 – CVDEN Capacitive Voltage Division Enable bit

Value	Description
1	CVD operation is enabled
0	CVD operation is disabled

Bits 18:16 – CVDCPL[2:0] Capacitor Voltage Divider (CVD) Setting bits

Value	Description
111	7 * 2.5 pF = 17.5 pF
110	6 * 2.5 pF = 15 pF
101	5 * 2.5 pF = 12.5 pF
100	4 * 2.5 pF = 10 pF
011	3 * 2.5 pF = 7.5 pF
010	2 * 2.5 pF = 5 pF
001	1 * 2.5 pF = 2.5 pF
000	0 * 2.5 pF = 0 pF

Bits 14:8 – SD3ACQTIME[6:0] Scan Descriptor Acquire Time
Specifies the time for which CVD waits for ADC voltage to settle.

Bits 6:0 – SD3CHGTIME[6:0] Scan Descriptor Charge Time
Specifies the time for which CVD remains in the charging state for internal or external capacitors, and for TX outputs.

33.6.26 CVD Scan Descriptor 3 Time 2

Name: CVDSD3T2
Offset: 0x13C
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
		SD3CHNTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
		SD3OVRTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
		SD3POLTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
		SD3CONTIME[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 30:24 – SD3CHNTIME[6:0] Scan Descriptor Channel Time
Controls the number of cycles the state machine waits in the RXCHAN or TXCHAN state before moving to the next RX/TX pair.

Bits 22:16 – SD3OVRTIME[6:0] Scan Descriptor Oversample Time
Controls the number of cycles the state machine waits in the OVERSAMP state before taking the next oversampling measurement of an RX/TX pair.

Bits 14:8 – SD3POLTIME[6:0] Scan Descriptor Polarity Time
Controls the number of cycles the state machine waits in the POLARITY state before taking the second polarity measurement of an RX/TX pair.

Bits 6:0 – SD3CONTIME[6:0] Scan Descriptor Convert Time
Controls the number of cycles the state machine waits in the CONVERT state for the ADC sample data. User must ensure that the ADC asserts End-Of-Convert (EOC) before the CONVERT state timer expires.

34. Analog Comparators (AC)

34.1 Overview

The Analog Comparator (AC) supports two individual comparators; one shared (with built-in supply voltage monitor (MVREF)) AC_CMP1, and one dedicated AC_CMP0.

Each comparator (COMP) compares the voltage levels on two inputs, and provides a digital output based on the comparison. Each comparator may be configured to generate interrupt requests and/or peripheral events upon several different combinations of input change.

The input selection includes up to four shared analog port pins and several internal signals. Each Comparator Output state can also be output on a pin for use by external devices.

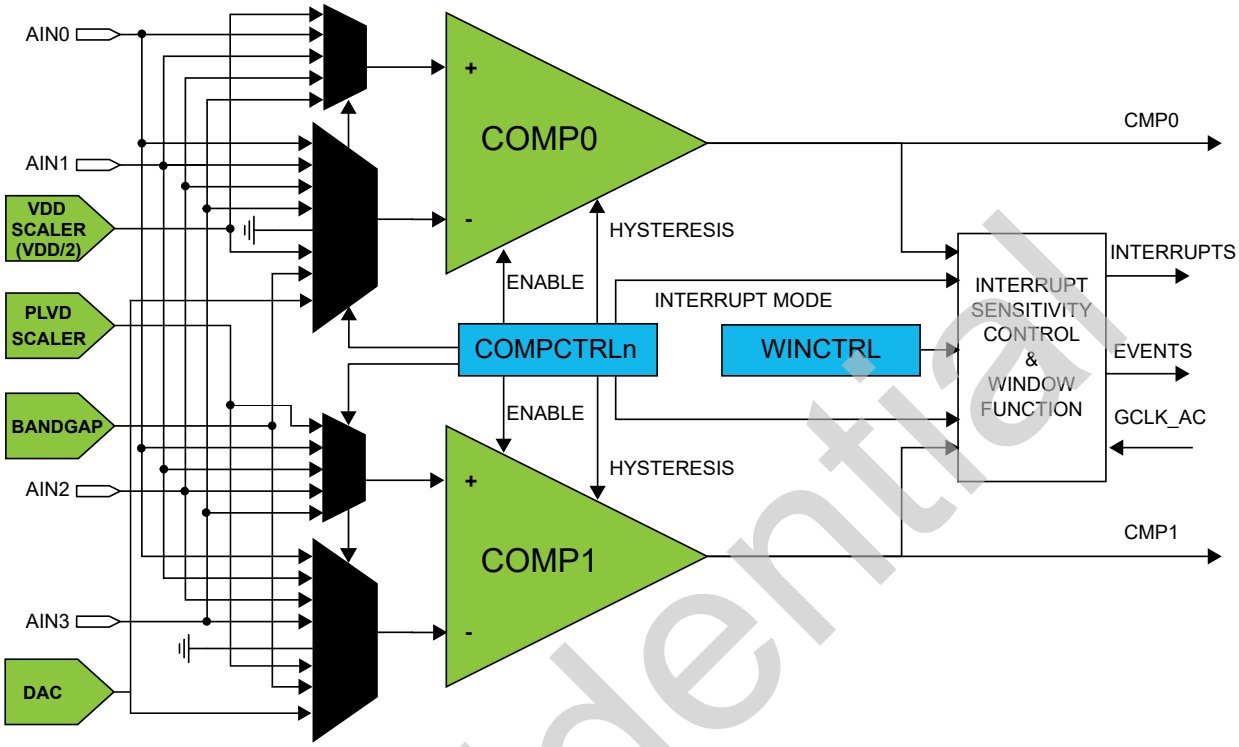
The comparators are grouped in pairs on each port. The AC peripheral implements pairtwo pairs of comparators . These are called Comparator 0 (COMP0) and Comparator 1 (COMP1) . for the first pair and Comparator 2 (COMP2) and Comparator 3 (COMP3) for the second pair. They have identical behaviors, but separate Control registers. TheEach pair can be set in Window mode to compare a signal to a voltage range instead of a single voltage level.

34.2 Features

- Two individual comparators (Single pair configuration)
- Analog comparator outputs available on pins
  - Asynchronous or synchronous
- Flexible input selection:
  - Up to four pins selectable for positive or negative inputs
  - Ground (for zero crossing)
  - Bandgap reference voltage
  - Programmable VDD scaler for AC_CMP1 (Shared with Programmable Low Voltage Detector (PLVD)) and fixed VDD/2 for AC_CMP0
  - DAC output
- Interrupt generation on:
  - Rising or falling edge
  - Toggle
  - End of comparison
- Window function interrupt generation on:
  - Signal above window
  - Signal inside window
  - Signal below window
  - Signal outside window
- Event generation on:
  - Comparator output
  - Window function inside/outside window
- Optional digital filter on comparator output

34.3 Block Diagram

Figure 34-1. Analog Comparator Block Diagram



34.4 Product Dependencies

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

34.4.1 I/O Lines

Using the AC's I/O lines requires the I/O pins to be configured as analog pins for AC_AINx inputs. See *I/O Ports and Peripheral Pin Select (PPS)* from Related Links.

Table 34-1. I/O LINES

Signal	Peripheral Function
AC_AIN0	Comparator input
AC_AIN1	Comparator input
AC_AIN2	Comparator input
AC_AIN3	Comparator input
AC_CMP0	Comparator output
AC_CMP1	Comparator output

Note:
1. To get the analog comparator output on I/O line, CFGCON1.CMP0_OE/CFGCON1.CMP1_OE needs to be set/enabled. See *System Configuration and Register Locking (CFG)* from Related Links.

Related Links

5. I/O Ports and Peripheral Pin Select (PPS)
18. System Configuration and Register Locking (CFG)

34.4.2 Power Management

The AC will continue to operate in any Sleep mode (Idle, Standby Sleep) where the selected source clock is running. The AC's interrupts can be used to wake-up the device from Sleep modes. Events connected to the Event System can trigger other operations in the system without exiting Standby Sleep mode.

34.4.3 Clocks

The AC bus clock (PB2_CLK) can be enabled and disabled in the Main Clock module, CRU (see *Clock and Reset Unit (CRU)* from Related Links), and the Analog Comparator module can be enabled or disabled via the PMD1 register. See *Peripheral Module Disable Register (PMD)* from Related Links.

A generic clock (GCLK_AC) is required to clock the AC. This clock must be configured and enabled in the generic clock controller before using the AC.

This generic clock is asynchronous to the bus clock (PB2_CLK). Due to this asynchronicity, writes to certain registers will require synchronization between the clock domains. See *Synchronization* from Related Links.

Related Links

13. Clock and Reset Unit (CRU)
34.5.13. Synchronization

34.4.4 DMA

Not applicable.

34.4.5 Interrupts

The interrupt request lines are connected to the interrupt controller. Using the AC interrupts requires the interrupt controller to be configured first. See *Nested Vector Interrupt Controller (NVIC)* from Related Links.

Related Links

8.2. Nested Vector Interrupt Controller (NVIC)

34.4.6 Events

The events are connected to the Event System. See *Event System (EVSYS)* from Related Links for details on how to configure the Event System.

Related Links

26. Event System (EVSYS)

34.4.7 Debug Operation

When the CPU is halted in debug mode, the AC will halt normal operation after any ongoing comparison is completed. The AC can be forced to continue normal operation during debugging. See *DBGCTRL* register from Related Links. If the AC is configured in a way that requires it to be periodically serviced by the CPU through interrupts or similar, improper operation or data loss may result during debugging.

Related Links

34.7.9. DBGCTRL

34.4.8 Register Access Protection

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

- Control B register (CTRLB)
- Interrupt Flag register (INTFLAG)

Optional write protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write Protection" property in each individual register description.

PAC write protection does not apply to accesses through an external debugger.

34.4.9 Analog Connections

Each comparator has up to four I/O pins that can be used as analog inputs. The pair of comparators shares the same four pins. These pins must be configured for analog operation before using them as comparator inputs.

Any internal reference source, such as a bandgap voltage reference, DAC must be configured and enabled prior to its use as a comparator input.

34.5 Functional Description

34.5.1 Principle of Operation

Each comparator has one positive input and one negative input. Each positive input may be chosen from a selection of analog input pins. Each negative input may be chosen from a selection of both analog input pins and internal inputs, such as bandgap voltage reference.

The digital output from the comparator is '1' when the difference between the positive and the negative input voltage is positive, and '0'otherwise.

The individual comparators can be used independently (Normal mode) or paired to form a window comparison (Window mode).

34.5.2 Basic Operation

34.5.2.1 Initialization

Some registers are enable-protected, meaning they can only be written when the module is disabled.

The following register is enable-protected:

- Event Control register (EVCTRL)

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

34.5.2.2 Enabling, Disabling and Resetting

The AC is enabled by writing a '1' to the Enable bit in the Control A register (CTRLA.ENABLE). The AC is disabled writing a '0' to CTRLA.ENABLE. In addition to this AC module enable configuration, AC must also be enabled in the PMD module.

The AC is reset by writing a '1' to the Software Reset bit in the Control A register (CTRLA.SWRST). All registers in the AC will be reset to their initial state, and the AC will be disabled. See *CTRLA* register from Related Links.

Related Links

34.7.1. CTRLA

34.5.2.3 Comparator Configuration

Each individual comparator must be configured by its respective Comparator Control register (COMPCTRLx) before that comparator is enabled. These settings cannot be changed while the comparator is enabled.

- Select the desired measurement mode with COMPCTRLx.SINGLE. See *Starting a Comparison* from Related Links.
- Fixed hysteresis does not support programmable Hysteresis.
- Fixed speed of operation
- Select the interrupt source with COMPCTRLx.INTSEL.
- Select the positive and negative input sources with the COMPCTRLx.MUXPOS and COMPCTRLx.MUXNEG bits. See *Selecting Comparator Inputs* from Related Links.
- Select the filtering option with COMPCTRLx.FLEN.
- Select the standby operation with the Run in Standby bit (COMPCTRLx.RUNSTDBY).

The individual comparators are enabled by writing a '1' to the Enable bit in the Comparator x Control registers (COMPCTRLx.ENABLE). The individual comparators are disabled by writing a '0' to COMPCTRLx.ENABLE. Writing a '0' to CTRLA.ENABLE will also disable all the comparators but will not clear their COMPCTRLx.ENABLE bits.

Related Links

34.5.2.4. Starting a Comparison
34.5.3. Selecting Comparator Inputs

34.5.2.4 Starting a Comparison

Each comparator channel can be in one of two different measurement modes, which is determined by the COMPCTRLx.SINGLE bit:

- Continuous measurement
- Single-shot

After being enabled, a start-up delay is required before the result of the comparison is ready. This start-up time is measured automatically to account for environmental changes, such as temperature or voltage supply level, and is specified in the Electrical Specifications. During the start-up time, the COMP output is not available.

The comparator can be configured to generate interrupts when the output toggles, when the output changes from '0' to '1' (rising edge), when the output changes from '1' to '0' (falling edge) or at the end of the comparison. An end-of-comparison interrupt can be used with the Single-Shot mode to chain further events in the system, regardless of the state of the comparator outputs. The Interrupt mode is set by the Interrupt Selection bit group in the Comparator Control register (COMPCTRLx.INTSEL). Events are generated using the comparator output state regardless of whether the interrupt is enabled or not.

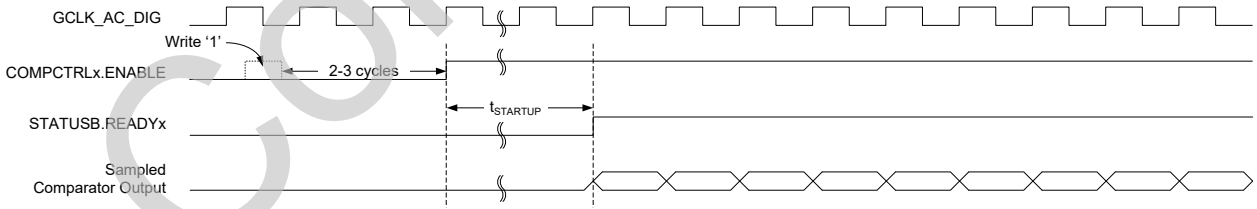
34.5.2.4.1 Continuous Measurement

Continuous measurement is selected by writing COMPCTRLx.SINGLE to zero. In continuous mode, the comparator is continuously enabled and performing comparisons. This ensures that the result of the latest comparison is always available in the Current State bit in the Status A register (STATUSA.STATEx).

After the start-up time has passed, a comparison is done and STATUSA is updated. The Comparator x Ready bit in the Status B register (STATUSB.READYx) is set, and the appropriate peripheral events and interrupts are also generated. New comparisons are performed continuously until the COMPCTRLx.ENABLE bit is written to zero. The start-up time applies only to the first comparison.

In the continuous operation, edge detection of the comparator output for interrupts is done by comparing the current and previous sample. The sampling rate is the GCLK_AC frequency. An example of continuous measurement is shown in the following figure.

Figure 34-2. Continuous Measurement Example



For low-power operation, comparisons can be performed during sleep mode without a clock. The comparator is enabled continuously, and changes of the comparator state are detected asynchronously. When a toggle occurs, the CRU will start GCLK_AC to register the appropriate peripheral events and interrupts. The GCLK_AC clock is, then, disabled again automatically unless configured to wake up the system from sleep.

34.5.2.4.2 Single-Shot

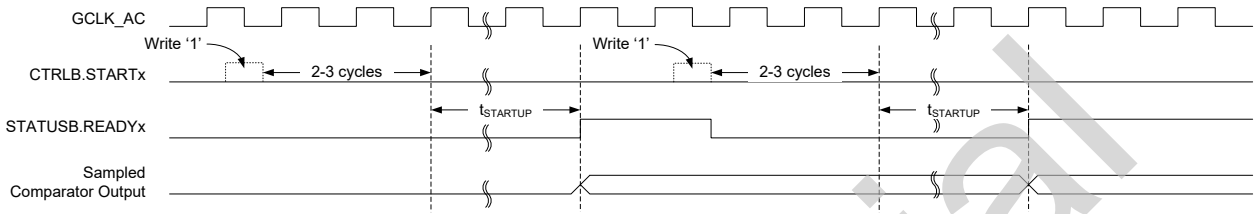
Single-shot operation is selected by writing COMPCTRLx.SINGLE to '1'. During single-shot operation, the comparator is normally idle. The user starts a single comparison by writing '1' to the respective Start Comparison bit in the write-only Control B register (CTRLB.STARTx). The comparator is enabled and, after the start-up time has passed, a single comparison is done and STATUSA is updated. Appropriate peripheral events and interrupts are also generated. No new comparisons will be performed.

Writing '1' to CTRLB.STARTx also clears the Comparator x Ready bit in the Status B register (STATUSB.READYx). STATUSB.READYx is set automatically by hardware when the single comparison has completed.

A single-shot measurement can also be triggered by the Event System. Setting the Comparator x Event Input bit in the Event Control Register (EVCTRL.COMPEIx) enables triggering on incoming peripheral events. Each comparator can be triggered independently by separate events. Event-triggered operation is similar to user-triggered operation; the difference is that a peripheral event from another hardware module causes the hardware to automatically start the comparison and will not clear STATUSB.READYx.

To detect an edge of the comparator output in single-shot operation for the purpose of interrupts, the result of the current measurement is compared with the result of the previous measurement (one sampling period earlier). An example of single-shot operation is shown in the following figure.

Figure 34-3. Single-Shot Example



For low-power operation, event-triggered measurements can be performed during sleep modes. When the event occurs, the CRU will start GCLK_AC. The comparator is enabled and, after the start-up time has passed, a comparison is done and appropriate peripheral events and interrupts are also generated. The comparator and GCLK_AC are, then, disabled again automatically unless configured to wake up the system from sleep.

34.5.3 Selecting Comparator Inputs

Each comparator has one positive and one negative input. The positive input is one of the external input pins (AINx). The negative input can be fed either from an external input pin (AINx) or from one of the internal reference voltage sources common to all comparators. The user selects the input source as follows:

- The positive input is selected by the Positive Input MUX Select bit group in the Comparator Control register (COMPCTRLx.MUXPOS).
- The negative input is selected by the Negative Input MUX Select bit group in the Comparator Control register (COMPCTRLx.MUXNEG).

In the case of using an external I/O pin, the selected pin must be configured for analog use in the GPIO by disabling the digital input and output. The switching of the analog input multiplexers is controlled to minimize crosstalk between the channels. The input selection must be changed only while the individual comparator is disabled.

Note: For internal use of the comparison results by the CCL module (see *Configurable Custom Logic (CCL)* from Related Links), COMPCTRLx.OUT must be `0x1` or `0x2`.

Related Links

24. Configurable Custom Logic (CCL)

34.5.4 Window Operation

The comparator pair can be configured to work together in Window mode. In this mode, a voltage range is defined, and the comparators give information about whether an input signal is within this range or not. Window mode is enabled by the Window Enable x bit in the Window Control register (WINCTRL.WENx). Both comparators must have the same measurement mode setting in their respective Comparator Control Registers (COMPCTRLx.SINGLE).

Notes:

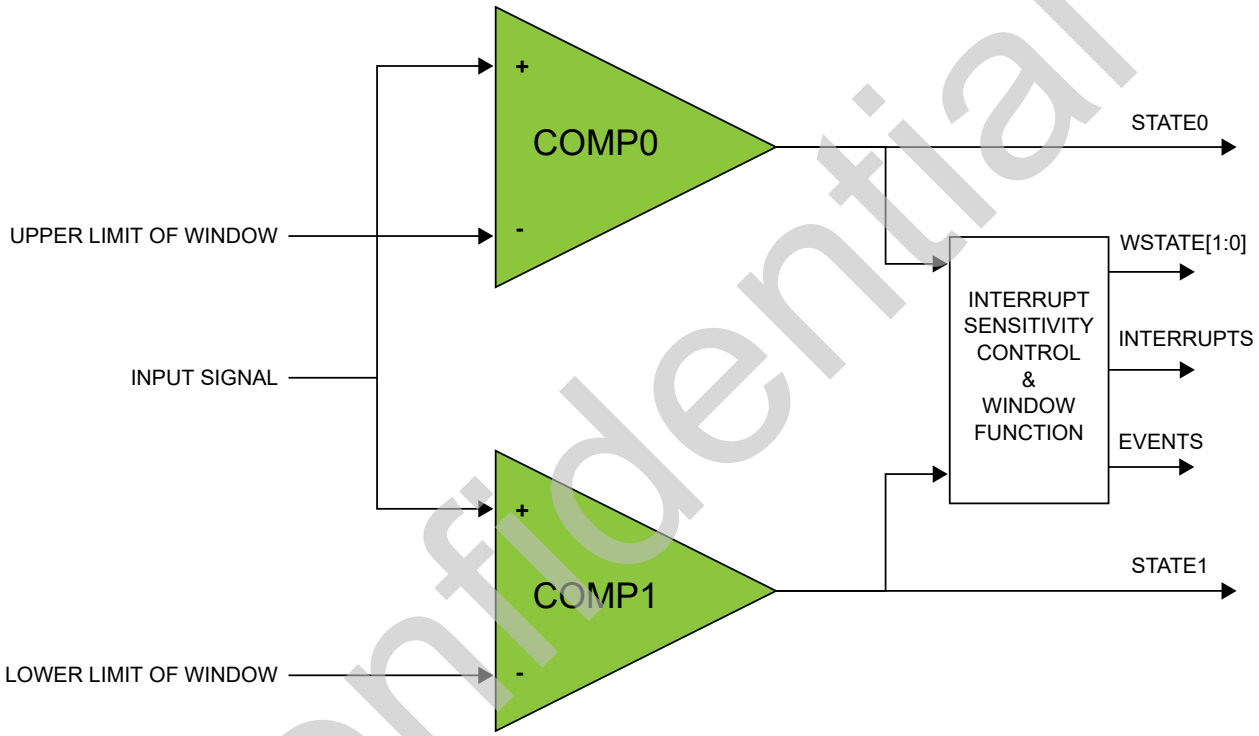
1. Both comparators must be enabled (COMPCTRLx.ENABLE = 1) before the Window mode is enabled (WINCTRL.WEN0 = 1).
2. Window mode must be first disabled (WINCTRL.WEN0 = 0) before both comparators are disabled (COMPCTRLx.ENABLE = 0).

To physically configure the comparators for Window mode, the same I/O pin must be chosen as positive input for each comparator, providing a shared input signal. The negative inputs define the range for the window. In the figure below, COMP0 defines the upper limit and COMP1 defines the lower limit of the window, as shown but the window will also work in the opposite configuration with COMP0 lower and COMP1 higher. The current state of the window function is available in the Window x State bit group of the Status register (STATUS.WSTATEx).

Window mode can be configured to generate interrupts when the input voltage changes to below the window, when the input voltage changes to above the window, when the input voltage changes into the window or when the input voltage changes outside the window. The interrupt selections are set by the Window Interrupt Selection bit field in the Window Control register (WINCTRL.WINTSEL). Events are generated using the inside/outside state of the window, regardless of whether the interrupt is enabled or not. Note that the individual comparator outputs, interrupts and events continue to function normally during Window mode.

When the comparators are configured for Window mode and Single-shot mode, measurements are performed simultaneously on both comparators. Writing '1' to either Start Comparison bit in the Control B register (CTRLB.STARTx) will start a measurement. Likewise either peripheral event can start a measurement. STATUSA.WSTATE bit field must be read only after STATUSB.READY bits of both the related comparators are set.

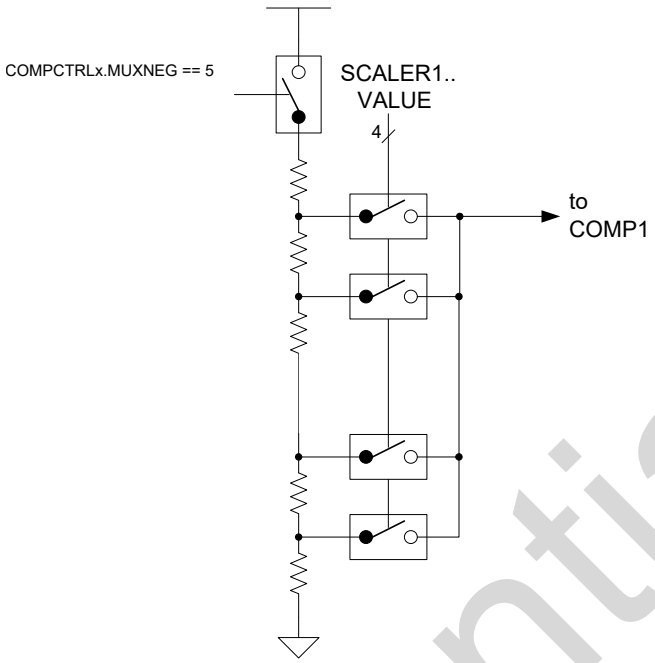
Figure 34-4. Comparators in Window Mode



34.5.5 VDD Scaler

The VDD scaler generates a reference voltage that is a fraction of the device's supply voltage with 16 levels. The programmable VDD scaler (PLVD Scaler) is available for AC_CMP1 only. AC_CMP0 uses a fixed VDD/2 reference. The scaler of a comparator is enabled when the Negative Input Mux bit field or the Positive Input Mux in the respective Comparator Control register (COMPCTRLx.MUXNEG) is set to 0x5 and the comparator is enabled. The voltage of each channel is selected by the Value bit field in the SCALER1 registers (SCALER1.VALUE) using the VDD resistor ladder.

Figure 34-5. PLVD Scaler



34.5.6 Filtering

The output of the comparators can be filtered digitally to reduce noise. The filtering is determined by the Filter Length bits in the Comparator Control x register (COMPCTRLx.FLEN), and is independent for each comparator. Filtering is selectable from none, 3-bit majority (N=3) or 5-bit majority (N=5) functions. Any change in the comparator output is considered valid only if N/2+1 out of the last N samples agree. The filter sampling rate is the GCLK_AC frequency.

Note that filtering creates an additional delay of N-1 sampling cycles from when a comparison is started until the comparator output is validated. For Continuous mode, the first valid output will occur when the required number of filter samples is taken. Subsequent outputs will be generated every cycle based on the current sample plus the previous N-1 samples, as shown in Figure 34-6. For Single-shot mode, the comparison completes after the Nth filter sample, as shown in Figure 34-7.

Figure 34-6. Continuous Mode Filtering

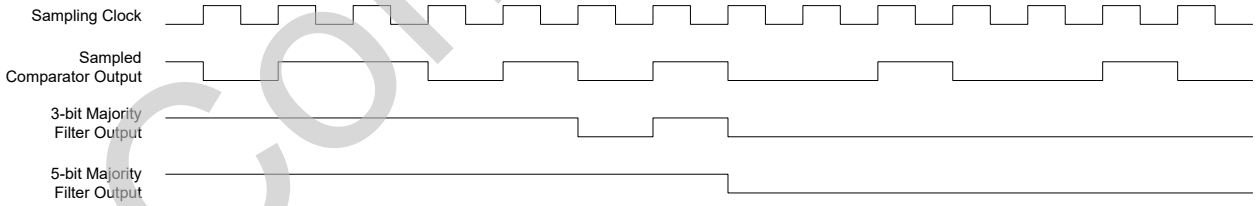
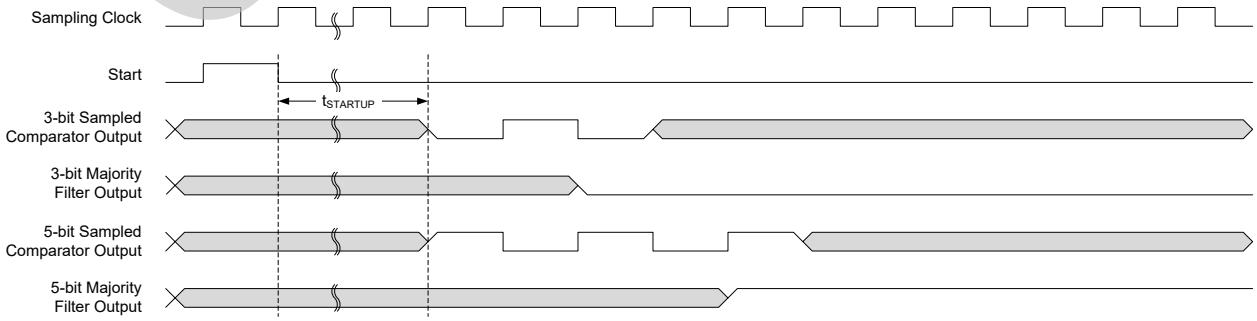


Figure 34-7. Single-Shot Filtering



During Sleep modes, filtering is supported only for single-shot measurements. Filtering must be disabled if continuous measurements will be done during Sleep modes, or the resulting interrupt/event may be generated incorrectly.

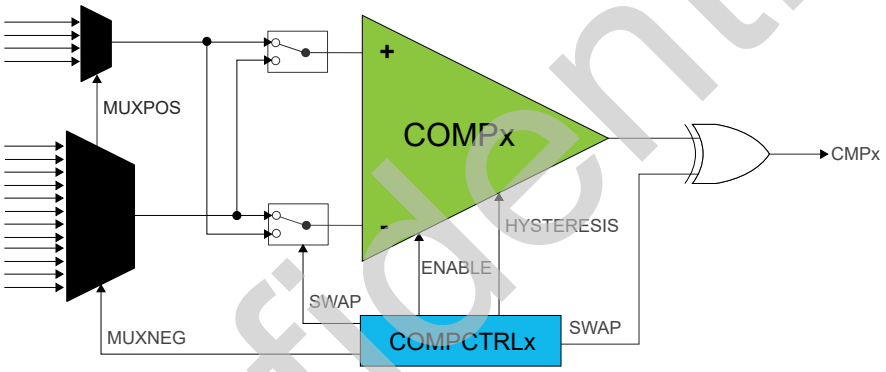
34.5.7 Comparator Output

The output of each comparator can be routed to an I/O pin by setting the Output bit group in the Comparator Control x register (COMPCTRLx.OUT). To get the analog comparator output on the I/O line, CFGCON1.CMP0_OE/ CFGCON1.CMP1_OE also needs to be set or enabled. This allows the comparator to be used by external circuitry. Either the raw, non-synchronized output of the comparator or the GCLK_AC-synchronized version, including filtering, can be used as the I/O signal source. The output appears on the corresponding AC_CMPx pin. The AC_CMP1 can be output on an alternate pin by configuring the CFGCON0.ACCMP1_ALTEN configuration.

34.5.8 Offset Compensation

The Swap bit in the Comparator Control registers (COMPCTRLx.SWAP) controls switching of the input signals to a comparator's positive and negative terminals. When the comparator terminals are swapped, the output signal from the comparator is also inverted, as shown in Figure 34-8. This allows the user to measure or compensate for the comparator input offset voltage. As part of the input selection, COMPCTRLx.SWAP can be changed only while the comparator is disabled.

Figure 34-8. Input Swapping for Offset Compensation



34.5.9 DMA Operation

Not applicable.

34.5.10 Interrupts

The AC has the following interrupt sources:

- Comparator (COMP0, COMP1, COMP2, COMP3): Indicates a change in comparator status
- Window (WIN0, WIN1): Indicates a change in the window status

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

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear (INTFLAG) register is set when the interrupt condition occurs. Each interrupt can be individually enabled by writing a one to the corresponding bit in the Interrupt Enable Set (INTENSET) register and disabled by writing a one to the corresponding bit in the Interrupt Enable Clear (INTENCLR) register.

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled or the AC is Reset. See *INTFLAG* register from Related Links for details on how to clear interrupt flags. All interrupt requests from the peripheral are OR'ed together on the system level to generate one combined interrupt request to the NVIC. The user must read the INTFLAG register to determine which interrupt condition is present. See *Nested Vector Interrupt Controller (NVIC)* from Related Links.

Note: Interrupts must be globally enabled for interrupt requests to be generated.

Related Links

8.2. Nested Vector Interrupt Controller (NVIC)

34.7.6. INTFLAG

34.5.11 Events

The AC can generate the following output events:

- Comparator (COMP0, COMP1, COMP2, COMP3): Generated as a copy of the comparator status
- Window (WIN0, WIN1): Generated as a copy of the window inside/outside status

Writing a one to an Event Output bit in the Event Control Register (EVCTRL.xxEO) enables the corresponding output event. Writing a zero to this bit disables the corresponding output event. See *Event System (EVSYS)* from Related Links for details on how to configure the Event System.

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

- Start comparison (START0, START1): Start a comparison

Writing a one to an Event Input bit into the Event Control register (EVCTRL.COMPEIx) enables the corresponding action on an input event. Writing a zero to this bit disables the corresponding action on an input event. Note that if several events are connected to the AC, the enabled action will be taken on any of the incoming events. See *Event System (EVSYS)* from Related Links for details on how to configure the Event System.

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

Related Links

26. Event System (EVSYS)

34.5.12 Sleep Mode Operation

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

For Window Mode operation, both comparators in a pair must have the same RUNSTDBY configuration.

When RUNSTDBY is one, any enabled AC interrupt source can wake up the CPU. The AC can also be used during sleep modes where the clock used by the AC is disabled, provided that the AC is still powered (not in shutdown). In this case, the behavior is slightly different and depends on the measurement mode, as listed in Table 34-2.

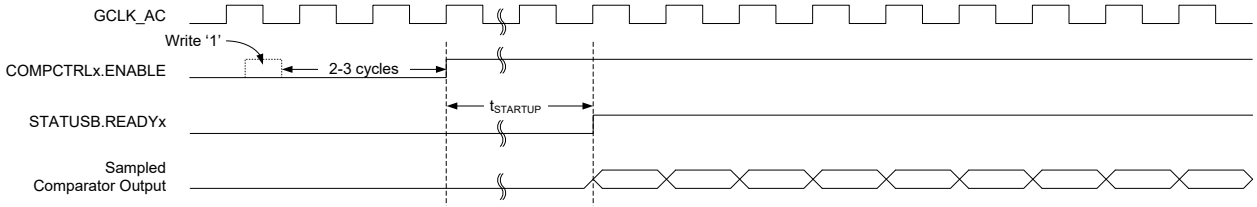
Table 34-2. Sleep Mode Operation

COMPCTRLx.MODE	RUNSTDBY=0	RUNSTDBY=1
0 (Continuous)	COMPx disabled	GCLK_AC stopped, COMPx enabled
1 (Single-shot)	COMPx disabled	GCLK_AC stopped, COMPx enabled only when triggered by an input event

34.5.12.1 Continuous Measurement during Sleep

When a comparator is enabled in continuous measurement mode and GCLK_AC is disabled during sleep, the comparator will remain continuously enabled and will function asynchronously. The current state of the comparator is asynchronously monitored for changes. If an edge matching the interrupt condition is found, GCLK_AC is started to register the interrupt condition and generate events. If the interrupt is enabled in the Interrupt Enable registers (INTENCLR/SET), the AC can wake up the device; otherwise GCLK_AC is disabled until the next edge detection. Filtering is not possible with this configuration.

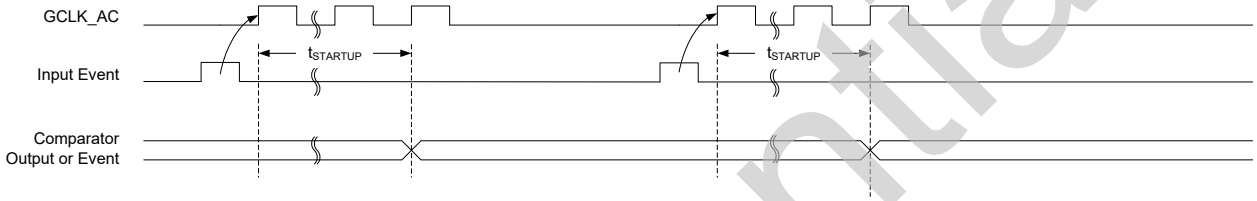
Figure 34-9. Continuous Mode SleepWalking



34.5.12.2 Single-Shot Measurement during Sleep

For low-power operation, event-triggered measurements can be performed during sleep modes. When the event occurs, the CRU will start GCLK_AC. The comparator is enabled and, after the start-up time has passed, a comparison is done, with filtering if desired, and the appropriate peripheral events and interrupts are also generated as shown in the following figure. The comparator and GCLK_AC are, then, disabled again automatically unless configured to wake the system from sleep. Filtering is allowed with this configuration.

Figure 34-10. Single-Shot SleepWalking



34.5.13 Synchronization

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

The following bits are synchronized when written:

- Software Reset bit in Control register (CTRLA.SWRST)
- Enable bit in Control register (CTRLA.ENABLE)
- Enable bit in Comparator Control register (COMPCTRLn.ENABLE)

The following registers are synchronized when written:

- Window Control register (WINCTRL)

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

34.6 Register Summary

See *AC* module in the *Product Memory Mapping Overview* from Related Links for base address.

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x00	CTRLA	7:0							ENABLE	SWRST
0x01	CTRLB	7:0					START3	START2	START1	START0
0x02	EVCTRL	7:0			WINEO1	WINEO0	COMPEO3	COMPEO2	COMPEO1	COMPEO0
		15:8	INVEI3	INVEI2	INVEI1	INVEI0	COMPEI3	COMPEI2	COMPEI1	COMPEI0
0x04	INTENCLR	7:0			WIN1	WIN0	COMP3	COMP2	COMP1	COMP0
0x05	INTENSET	7:0			WIN1	WIN0	COMP3	COMP2	COMP1	COMP0
0x06	INTFLAG	7:0			WIN1	WIN0	COMP3	COMP2	COMP1	COMP0
0x07	STATUSA	7:0	WSTATE1[1:0]		WSTATE0[1:0]		STATEx	STATEx	STATE1	STATE0
0x08	STATUSB	7:0							READY1	READY0
0x09	DBGCTRL	7:0								DBGRUN
0x0A	WINCTRL	7:0		WINTSEL1[1:0]		WEN1		WINTSEL0[1:0]		WEN0
0x0B ... 0x0C	Reserved									
0x0D	SCALER1	7:0					VALUE[3:0]			
0x0E ... 0x0F	Reserved									
0x10	COMPCTRL0	7:0		RUNSTDBY		INTSEL[1:0]		SINGLE	ENABLE	
		15:8	SWAP	MUXPOS[2:0]				MUXNEG[2:0]		
		23:16								
		31:24			OUT[1:0]			FLEN[2:0]		
0x14	COMPCTRL1	7:0		RUNSTDBY		INTSEL[1:0]		SINGLE	ENABLE	
		15:8	SWAP	MUXPOS[2:0]				MUXNEG[2:0]		
		23:16								
		31:24			OUT[1:0]			FLEN[2:0]		
0x18 ... 0x1F	Reserved									
0x20	SYNCBUSY	7:0		COMPCTRL3	COMPCTRL2	COMPCTRL1	COMPCTRL0	WINCTRL	ENABLE	SWRST
		15:8								
		23:16								
		31:24								

Related Links

7. Product Memory Mapping Overview

34.7 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. See *Register Access Protection* from Related Links.

Some registers are synchronized when read and/or written. Synchronization is denoted by the “Write-Synchronized” or the “Read-Synchronized” property in each individual register description. See *Synchronization* from Related Links.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.

Related Links

34.4.8. Register Access Protection
34.5.13. Synchronization

34.7.1 Control A

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

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

Bit 1 – ENABLE Enable

Due to synchronization, there is delay from updating the register until the peripheral is enabled/disabled. The value written to CTRL.ENABLE will read back immediately and the corresponding bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE is cleared when the peripheral is enabled/disabled.
Note: To avoid spurious interrupts from multiple enable/disable cycles of AC, use the SWRST bit to reset the comparator module.

Value	Description
0	The AC is disabled.
1	The AC is enabled. Each comparator must also be enabled individually by the Enable bit in the Comparator Control register (COMPCTRLn.ENABLE).

Bit 0 – SWRST Software Reset

Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets all registers in the AC to their initial state, and the AC will be disabled.
Writing a '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded.
Due to synchronization, there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the reset is complete.

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

34.7.2 Control B

Name: CTRLB
Offset: 0x01
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
					START3	START2	START1	START0
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bits 0, 1 – STARTx Comparator x Start Comparison
Writing a '0' to this field has no effect.
Writing a '1' to STARTx starts a single-shot comparison on COMPx if both the Single-Shot and Enable bits in the Comparator x Control Register are '1' (COMPCTRLx.SINGLE and COMPCTRLx.ENABLE). If comparator x is not implemented, or if it is not enabled in single-shot mode, Writing a '1' has no effect.
This bit always reads as zero.

Bits 0, 1, 2, 3 – STARTx Comparator x Start Comparison
Writing a '0' to this field has no effect.
Writing a '1' to STARTx starts a single-shot comparison on COMPx if both the Single-Shot and Enable bits in the Comparator x Control Register are '1' (COMPCTRLx.SINGLE and COMPCTRLx.ENABLE). If comparator x is not enabled in single-shot mode, writing a '1' has no effect.
This bit always reads as zero.

34.7.3 Event Control

Name: EVCTRL
Offset: 0x02
Reset: 0x0000
Property: PAC Write-Protection, Enable-Protected

Bit	15	14	13	12	11	10	9	8
	INVEI3	INVEI2	INVEI1	INVEI0	COMPEI3	COMPEI2	COMPEI1	COMPEI0
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
			WINEO1	WINEO0	COMPEO3	COMPEO2	COMPEO1	COMPEO0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bits 12, 13 – INVEIx Inverted Event Input Enable x

Value	Description
0	Incoming event is not inverted for comparator x.
1	Incoming event is inverted for comparator x.

Bits 12, 13, 14, 15 – INVEIx Inverted Event Input Enable x

Value	Description
0	Incoming event is not inverted for comparator x.
1	Incoming event is inverted for comparator x.

Bits 8, 9 – COMPEIx Comparator x Event Input

Note that several actions can be enabled for incoming events. If several events are connected to the peripheral, the enabled action will be taken for any of the incoming events. There is no way to tell which of the incoming events caused the action.

These bits indicate whether a comparison will start or not on any incoming event.

Value	Description
0	Comparison will not start on any incoming event.
1	Comparison will start on any incoming event.

Bits 8, 9, 10, 11 – COMPEIx Comparator x Event Input

Note that several actions can be enabled for incoming events. If several events are connected to the peripheral, the enabled action will be taken for any of the incoming events. There is no way to tell which of the incoming events caused the action.

These bits indicate whether a comparison will start or not on any incoming event.

Value	Description
0	Comparison will not start on any incoming event.
1	Comparison will start on any incoming event.

Bit 4 – WINEO0 Window 0 Event Output Enable

These bits indicate whether the window 0 function can generate a peripheral event or not.

Value	Description
0	Window 0 Event is disabled.
1	Window 0 Event is enabled.

Bits 4, 5 – WINEOx Window x Event Output Enable

These bits indicate whether the window x function can generate a peripheral event or not.

Value	Description
0	Window x Event is disabled.
1	Window x Event is enabled.

Bits 0, 1 – COMPEOx Comparator x Event Output Enable

These bits indicate whether the comparator x output can generate a peripheral event or not.

Value	Description
0	COMPx event generation is disabled.
1	COMPx event generation is enabled.

Bits 0, 1, 2, 3 – COMPEOx Comparator x Event Output Enable

These bits indicate whether the comparator x output can generate a peripheral event or not.

Value	Description
0	COMPx event generation is disabled.
1	COMPx event generation is enabled.

34.7.4 Interrupt Enable Clear

Name: INTENCLR
Offset: 0x04
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
			WIN1	WIN0	COMP3	COMP2	COMP1	COMP0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit 4 – WIN0 Window 0 Interrupt Enable
Reading this bit returns the state of the Window 0 interrupt enable.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit disables the Window 0 interrupt.

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

Bits 4, 5 – WINx Window x Interrupt Enable
Reading this bit returns the state of the Window x interrupt enable.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit disables the Window x interrupt.

Value	Description
0	The Window x interrupt is disabled.
1	The Window x interrupt is enabled.

Bits 0, 1 – COMPx Comparator x Interrupt Enable
Reading this bit returns the state of the Comparator x interrupt enable.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit disables the Comparator x interrupt.

Value	Description
0	The Comparator x interrupt is disabled.
1	The Comparator x interrupt is enabled.

Bits 0, 1, 2, 3 – COMPx Comparator x Interrupt Enable
Reading this bit returns the state of the Comparator x interrupt enable.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit disables the Comparator x interrupt.

Value	Description
0	The Comparator x interrupt is disabled.
1	The Comparator x interrupt is enabled.

34.7.5 Interrupt Enable Set

Name: INTENSET
Offset: 0x05
Reset: 0x00
Property: PAC Write-Protection

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

Bit	7	6	5	4	3	2	1	0
			WIN1	WIN0	COMP3	COMP2	COMP1	COMP0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit 4 – WIN0 Window 0 Interrupt Enable
Reading this bit returns the state of the Window 0 interrupt enable.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit enables the Window 0 interrupt.

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

Bits 4, 5 – WINx Window x Interrupt Enable
Reading this bit returns the state of the Window x interrupt enable.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit enables the Window x interrupt.

Value	Description
0	The Window x interrupt is disabled.
1	The Window x interrupt is enabled.

Bits 0, 1 – COMPx Comparator x Interrupt Enable
Reading this bit returns the state of the Comparator x interrupt enable.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Ready interrupt bit and enable the Ready interrupt.

Value	Description
0	The Comparator x interrupt is disabled.
1	The Comparator x interrupt is enabled.

Bits 0, 1, 2, 3 – COMPx Comparator x Interrupt Enable
Reading this bit returns the state of the Comparator x interrupt enable.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Ready interrupt bit and enable the Ready interrupt.

Value	Description
0	The Comparator x interrupt is disabled.
1	The Comparator x interrupt is enabled.

34.7.6 Interrupt Flag Status and Clear

Name: INTFLAG
Offset: 0x06
Reset: 0x00
Property: –

Bit	7	6	5	4	3	2	1	0
			WIN1	WIN0	COMP3	COMP2	COMP1	COMP0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit 4 – WIN0 Window 0
This flag is set according to the Window 0 Interrupt Selection bit group in the WINCTRL register (WINCTRL.WINTSELx) and will generate an interrupt if INTENCLR/SET.WINx is also one.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Window 0 interrupt flag.

Bits 4, 5 – WINx Window x
This flag is set according to the Window x Interrupt Selection bit group in the WINCTRL register (WINCTRL.WINTSELx) and will generate an interrupt if INTENCLR/SET.WINx is also one.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Window x interrupt flag.

Bits 0, 1 – COMPx Comparator x
Reading this bit returns the status of the Comparator x interrupt flag. If comparator x is not implemented, COMPx always reads as zero.
This flag is set according to the Interrupt Selection bit group in the Comparator x Control register (COMPCTRLx.INTSEL) and will generate an interrupt if INTENCLR/SET.COMPx is also one.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Comparator x interrupt flag.

Bits 0, 1, 2, 3 – COMPx Comparator x
Reading this bit returns the status of the Comparator x interrupt flag. If comparator x is not implemented, COMPx always reads as zero.
This flag is set according to the Interrupt Selection bit group in the Comparator x Control register (COMPCTRLx.INTSEL) and will generate an interrupt if INTENCLR/SET.COMPx is also one.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Comparator x interrupt flag.

Bit 0 – SWRST Software Reset
Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets all registers in the AC to their initial state, and the AC will be disabled.
Writing a '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded.
Due to synchronization, there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the reset is complete.

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

34.7.7 Status A

Name: STATUSA
Offset: 0x07
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	WSTATE1[1:0]		WSTATE0[1:0]		STATEx	STATEx	STATE1	STATE0
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 7:6 – WSTATE1[1:0] Window 1 Current State
These bits show the current state of the signal if the window 1 mode is enabled.

Value	Name	Description
0x0	ABOVE	Signal is above window
0x1	INSIDE	Signal is inside window
0x2	BELOW	Signal is below window
0x3		Reserved

Bits 5:4 – WSTATE0[1:0] Window 0 Current State
These bits show the current state of the signal if the window 0 mode is enabled.
These values may change in during startup and measurement cycles. When polling for sample completion use both the STATUSB.READYx bits to signal completion.

Value	Name	Description
0x0	ABOVE	Signal is above window
0x1	INSIDE	Signal is inside window
0x2	BELOW	Signal is below window
0x3	-	Reserved

Bits 3,2,1,0 – STATEx Comparator x Current State
This bit shows the current state of the output signal from COMPx. STATEx is valid only when the STATUSB.READYx bit is one.

Bits 0, 1 – STATEx Comparator x Current State
This bit shows the current state of the output signal from COMPx. STATEx is valid only when the STATUSB.READYx bit is one.

34.7.8 Status B

Name: STATUSB
Offset: 0x08
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
							READY1	READY0
Access							R	R
Reset							0	0

Bits 0, 1 – READYx Comparator x Ready

Value	Description
0	This bit is cleared when the comparator x output is not ready.
1	This bit is set when the comparator x output is ready.

34.7.9 Debug Control

Name: DBGCTRL
Offset: 0x09
Reset: 0x00
Property: PAC Write-Protection

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

Bit 0 – DBGRUN Debug Run
This bit is not reset by a software reset.
This bits controls the functionality when the CPU is halted by an external debugger.

Value	Description
0	The AC is halted when the CPU is halted by an external debugger. Any on-going comparison will complete before AC is halted.
1	The AC continues normal operation when the CPU is halted by an external debugger.

34.7.10 Window Control

Name: WINCTRL
Offset: 0x0A
Reset: 0x00
Property: PAC Write-Protection, Write-Synchronized

Bit	7	6	5	4	3	2	1	0
		WINTSEL1[1:0]		WEN1		WINTSEL0[1:0]		WEN0
Access		R/W	R/W	R/W		R/W	R/W	R/W
Reset		0	0	0		0	0	0

Bits 6:5 – WINTSEL1[1:0] Window 1 Interrupt Selection
These bits configure the interrupt mode for the comparator window 1 mode.

Value	Name	Description
0x0	ABOVE	Interrupt on signal above window
0x1	INSIDE	Interrupt on signal inside window
0x2	BELOW	Interrupt on signal below window
0x3	OUTSIDE	Interrupt on signal outside window

Bit 4 – WEN1 Window 1 Mode Enable

Value	Description
0	Window mode is disabled for comparators 2 and 3.
1	Window mode is enabled for comparators 2 and 3.

Bits 2:1 – WINTSEL0[1:0] Window 0 Interrupt Selection
These bits configure the interrupt mode for the comparator window 0 mode.

Value	Name	Description
0x0	ABOVE	Interrupt on signal above window
0x1	INSIDE	Interrupt on signal inside window
0x2	BELOW	Interrupt on signal below window
0x3	OUTSIDE	Interrupt on signal outside window

Bit 0 – WEN0 Window 0 Mode Enable

Value	Description
0	Window mode is disabled.
1	Window mode is enabled.

34.7.11 Scaler 1

Name: SCALER1
Offset: 0x0D
Reset: 0x00
Property: PAC Write-Protection

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

Bits 3:0 – VALUE[3:0] Scaler Value
These bits define the scaling factor for channel 1 of the VDD voltage scaler. The output voltage, V_{SCALE}, is:

$$V_{\mathrm{SCALE}} = V_{DD} \times \left(\frac{R\_Bottom}{R\_Total}\right)$$

Where, R_Total = 900.
Refer to the following table for R_Bottom for different VALUE[3:0]
For example, V_{SCALE} for VALUE[3:0] = 0x02 at VDD = 3.3V

$$V_{\mathrm{SCALE}} = 3.3 \times \left(\frac{598.5}{900}\right) = 2.1945V$$

Table 34-3. Scaler Value

Value[3:0]	R_Bottom
0x0	Reserved
0x1	Reserved
0x2	598.5
0x3	634.5
0x4	306
0x5	324
0x6	328.5
0x7	360
0x8	387
0x9	400.5
0xA	432
0xB	450
0xC	468
0xD	490.5
0xE	481.5
0xF	External Reference on LVDIN pin

34.7.12 Comparator Control n

Name: COMPCTRL
Offset: 0x10 + n*0x04 [n=0..1]
Reset: 0x00000000
Property: PAC Write-Protection

Bit	31	30	29	28	27	26	25	24
			OUT[1:0]			FLEN[2:0]		
Access			R/W	R/W		R/W	R/W	R/W
Reset			0	0		0	0	0

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	SWAP	MUXPOS[2:0]				MUXNEG[2:0]		
Access	R/W	R/W	R/W	R/W		R/W	R/W	R/W
Reset	0	0	0	0		0	0	0

Bit	7	6	5	4	3	2	1	0
		RUNSTDBY		INTSEL[1:0]		SINGLE	ENABLE	
Access		R/W		R/W	R/W	R/W	R/W	
Reset		0		0	0	0	0	

Bits 29:28 – OUT[1:0] Output
These bits configure the output selection for comparator n. COMPCTRLn.OUT can be written only while COMPCTRLn.ENABLE is zero.
Note: For internal use of the comparison results by the CCL, this must be 0x1 or 0x2.

These bits are not synchronized.

Value	Name	Description
0x0	OFF	The output of COMPn is not routed to the COMPn I/O port
0x1	ASYNC	The asynchronous output of COMPn is routed to the COMPn I/O port
0x2	SYNC	The synchronous output (including filtering) of COMPn is routed to the COMPn I/O port
0x3	N/A	Reserved

Bits 26:24 – FLEN[2:0] Filter Length
These bits configure the filtering for comparator n. COMPCTRLn.FLEN can only be written while COMPCTRLn.ENABLE is zero.
These bits are not synchronized.

Value	Name	Description
0x0	OFF	No filtering
0x1	MAJ3	3-bit majority function (2 of 3)
0x2	MAJ5	5-bit majority function (3 of 5)
0x3-0x7	N/A	Reserved

Bit 15 – SWAP Swap Inputs and Invert
This bit swaps the positive and negative inputs to COMPn and inverts the output. This function can be used for offset cancellation. COMPCTRLn.SWAP can be written only while COMPCTRLn.ENABLE is zero.
These bits are not synchronized.

Value	Description
0	The output of MUXPOS connects to the positive input, and the output of MUXNEG connects to the negative input.

Value	Description
1	The output of MUXNEG connects to the positive input, and the output of MUXPOS connects to the negative input.

Bits 14:12 – MUXPOS[2:0] Positive Input Mux Selection

These bits select which input will be connected to the positive input of comparator n. COMPCTRLn.MUXPOS can be written only while COMPCTRLn.ENABLE is zero.

These bits are not synchronized.

Value	Name	Description
0x0	PIN0	AC_AIN0
0x1	PIN1	AC_AIN1
0x2	PIN2	AC_AIN2
0x3	PIN3	AC_AIN3
0x4	VSCALE	VDD scaler
0x5-0x7	-	Reserved

Bits 10:8 – MUXNEG[2:0] Negative Input Mux Selection

These bits select which input will be connected to the negative input of comparator n. COMPCTRLn.MUXNEG can only be written while COMPCTRLn.ENABLE is zero.

These bits are not synchronized.

Value	Name	Description
0x0	PIN0	I/O pin 0
0x1	PIN1	I/O pin 1
0x2	PIN2	I/O pin 2
0x3	PIN3	I/O pin 3
0x4	GND	Ground
0x5	VSCALE	VDD scaler
0x6	BANDGAP	Internal bandgap voltage
0x7	DAC	DAC output

Bit 6 – RUNSTDBY Run in Standby

This bit controls the behavior of the comparator during standby sleep mode.

This bit is not synchronized

Value	Description
0	The comparator is disabled during sleep.
1	The comparator continues to operate during sleep.

Bits 4:3 – INTSEL[1:0] Interrupt Selection

These bits select the condition for comparator n to generate an interrupt or event. COMPCTRLn.INTSEL can be written only while COMPCTRLn.ENABLE is zero.

These bits are not synchronized.

Value	Name	Description
0x0	TOGGLE	Interrupt on comparator output toggle
0x1	RISING	Interrupt on comparator output rising
0x2	FALLING	Interrupt on comparator output falling
0x3	EOC	Interrupt on end of comparison (single-shot mode only)

Bit 2 – SINGLE Single-Shot Mode

This bit determines the operation of comparator n. COMPCTRLn.SINGLE can be written only while COMPCTRLn.ENABLE is zero.

These bits are not synchronized.

Value	Description
0	Comparator n operates in continuous measurement mode.
1	Comparator n operates in single-shot mode.

Bit 1 – ENABLE Enable

Writing a zero to this bit disables comparator n.
Writing a one to this bit enables comparator n.
Due to synchronization, there is delay from updating the register until the comparator is enabled/disabled. The value written to COMPCTRLn.ENABLE will read back immediately after being written. SYNCBUSY.COMPCTRLn is set. SYNCBUSY.COMPCTRLn is cleared when the peripheral is enabled/disabled.
Writing a one to COMPCTRLn.ENABLE will prevent further changes to the other bits in COMPCTRLn. These bits remain protected until COMPCTRLn.ENABLE is written to zero and the write is synchronized.

34.7.13 Synchronization Busy

Name: SYNCBUSY
Offset: 0x20
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
		COMPCTRL3	COMPCTRL2	COMPCTRL1	COMPCTRL0	WINCTRL	ENABLE	SWRST
Access		R	R	R	R	R	R	R
Reset		0	0	0	0	0	0	0

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

Bits 3, 4, 5, 6 – COMPCTRLx COMPCTRLx Synchronization Busy
This bit is cleared when the synchronization of the COMPCTRLx register between the clock domains is complete.
This bit is set when the synchronization of the COMPCTRLx register between clock domains is started.

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

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

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

35. Timer/Counter (TC)

35.1 Overview

There are up to eight TC peripheral instances. Up to four TCs (TC[3:0]) are in PD1, whereas TC4, present in all device configurations, is always located in power domain PD0.

Each TC 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 timestamp input events or IO pin edges, allowing for capturing of frequency and/or pulse width.

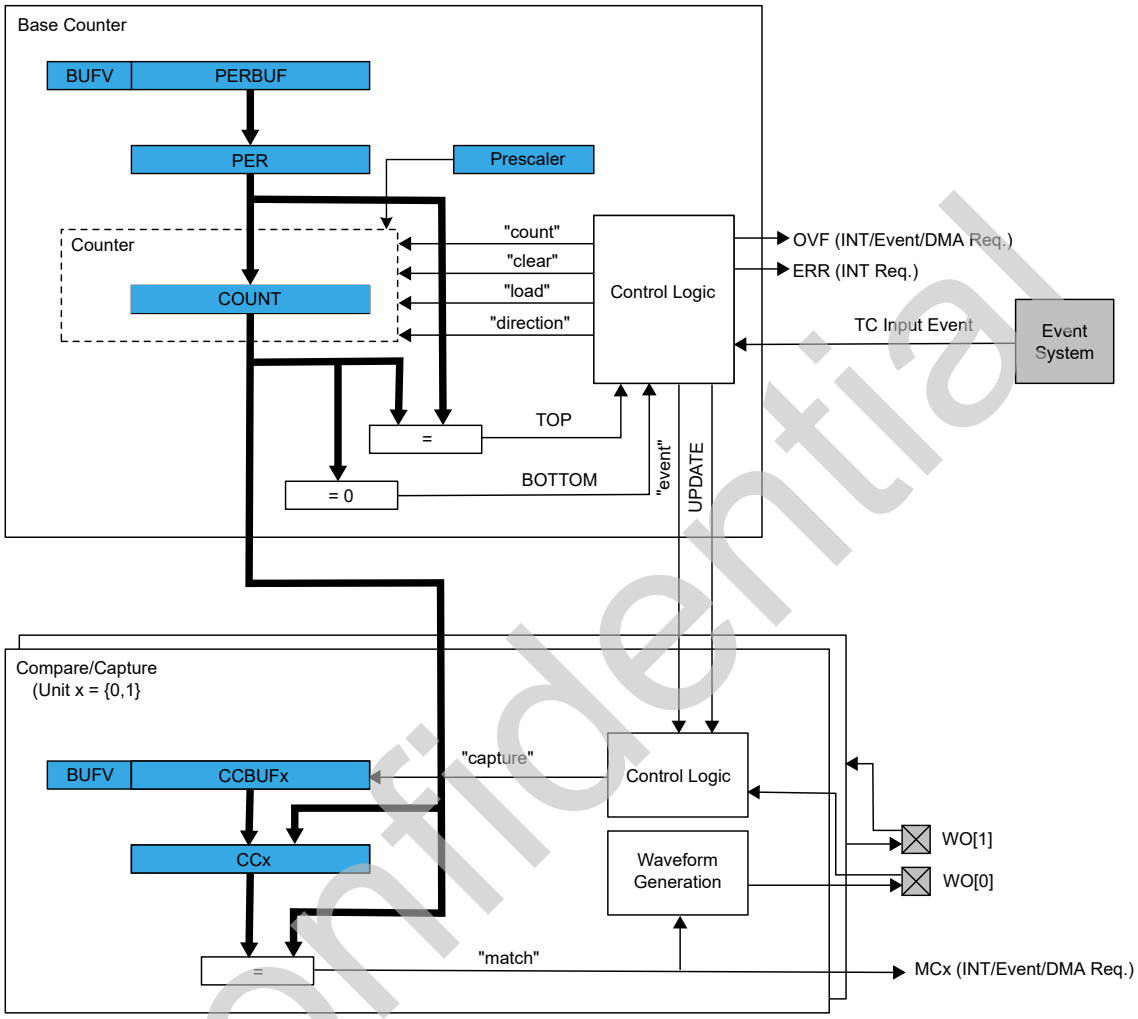
A TC can also perform waveform generation, such as frequency generation and pulse-width modulation.

35.2 Features

- Selectable configuration
  - 8-, 16- or 32-bit TC operation, with compare/capture channels
- 2 compare/capture channels (CC) with:
  - Double buffered timer period setting (in 8-bit mode only)
  - Double buffered compare channel
- Waveform generation
  - Frequency generation
  - Single-slope pulse-width modulation
- Input capture
  - Event / IO pin edge capture
  - Frequency capture
  - Pulse-width capture
  - Time-stamp capture
  - Minimum and maximum capture (only available on SAM C20/C21 N variants)
- One input event
- Interrupts/output events on:
  - Counter overflow/underflow
  - Compare match or capture
- Internal prescaler
- DMA support

35.3 Block Diagram

Figure 35-1. Timer/Counter Block Diagram



35.4 Signal Description

Table 35-1. Signal Description for TC

Signal Name	Type	Description
WO[1:0]	Digital output	Waveform output
	Digital input	Capture input

For details on the pin mapping for this peripheral, see *I/O Ports and Peripheral Pin Select (PPS)* from Related Links. One signal can be mapped on several pins

Related Links

5. I/O Ports and Peripheral Pin Select (PPS)

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 I/O lines of this peripheral, the I/O pins must be configured using the I/O Pin Controller (PORT), see *Port Register Summary* from Related Links.

.

Related Links

5.10. Port Register Summary

35.5.2 Power Management

This peripheral can continue to operate in any Sleep mode (Idle, Standby sleep) where its source clock is running. The interrupts can wake-up the device from Sleep modes. Events connected to the event system can trigger other operations in the system without exiting Sleep modes.

35.5.3 Clocks

The TC bus clocks (CLK_TCx_APB) can be enabled and disabled in the Clock and Reset Unit (CRU).

The generic clocks (GCLK_TCx) are asynchronous to the user interface clock (CLK_TCx_APB). Due to this asynchronicity, accessing certain registers will require synchronization between the clock domains.

Note: Two instances of the TC may share a peripheral clock channel. In this case, they cannot be set to different clock frequencies. See *Clock and Reset Unit (CRU)* from Related Links.

Related Links

13. Clock and Reset Unit (CRU)

35.5.4 DMA

The DMA request lines are connected to the DMA Controller (DMAC). In order to use DMA requests with this peripheral, the DMAC must be configured first (see *Direct Memory Access Controller (DMAC)* from Related Links).

Related Links

22. Direct Memory Access Controller (DMAC)

35.5.5 Interrupts

The interrupt request line is connected to the Interrupt Controller. In order to use interrupt requests of this peripheral, the Interrupt Controller (NVIC) must be configured first. See *Nested Vector Interrupt Controller (NVIC)* from Related Links.

Related Links

8.2. Nested Vector Interrupt Controller (NVIC)

35.5.6 Events

The events of this peripheral are connected to the Event System.

Related Links

26. Event System (EVSYS)

35.5.7 Debug Operation

When the CPU is halted in Debug mode, this peripheral will halt normal operation. This peripheral can be forced to continue operation during debugging. For more details, see *DBGCTRL* from Related Links.

Related Links
35.8.11. DBGCTRL

35.5.8 Register Access Protection

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

- Interrupt Flag Status and Clear register (INTFLAG)
- Status register (STATUS)
- Count register (COUNT)
- Count register (COUNT)
- Period and Period Buffer registers (PER, PERBUF)
- Compare/Capture Value registers and Compare/Capture Value Buffer registers (CCx, CCBUFx)

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.

35.5.9 Analog Connections

Not applicable.

35.6 Functional Description

35.6.1 Principle of Operation

The following definitions are used throughout the documentation:

Table 35-2. Timer/Counter Definitions

Name	Description
TOP	The counter reaches TOP when it becomes equal to the highest value in the count sequence. The TOP value can be the same as Period (PER) or the Compare Channel 0 (CC0) register value depending on the waveform generator mode in Waveform Output Operations. See *Waveform Output Operations* from Related Links.
ZERO	The counter is ZERO when it contains all zeros.
MAX	The counter reaches MAX when it contains all ones.
UPDATE	The timer/counter signals an update when it reaches ZERO or TOP, depending on the direction settings.
Timer	The timer/counter clock control is handled by an internal source.
Counter	The clock control is handled externally (e.g., counting external events).
CC	For compare operations, the CC are referred to as “compare channels.” For capture operations, the CC are referred to as “capture channels.”

Each TC instance has up to two compare/capture channels (CC0 and CC1).

The counter in the TC can either count events from the Event System or clock ticks of the GCLK_TCx clock, which may be divided by the prescaler.

The counter value is passed to the CCx where it can be either compared to user-defined values or captured.

For optimized timing, the CCx and CCBUFx registers share a common resource. When writing into CCBUFx, lock the access to the corresponding CCx register (SYNCBUSY.CCX = 1) until the CCBUFx register value is not loaded into the CCx register (BUFVx == 1). Each buffer register has a buffer valid (BUFV) flag that indicates when the buffer contains a new value.

The Counter register (COUNT) and the Compare and Capture registers with buffers (CCx and CCBUFx) can be configured as 8-, 16- or 32-bit registers, with corresponding MAX values. Mode settings (CTRLA.MODE) determine the maximum range of the Counter register.

In 8-bit mode, a Period Value (PER) register and its Period Buffer Value (PERBUF) register are also available. The counter range and the operating frequency determine the maximum time resolution achievable with the TC peripheral.

The TC can be set to count up or down. Under normal operation, the counter value is continuously compared to the TOP or ZERO value to determine whether the counter has reached that value. On a comparison match, the TC can request DMA transactions, or generate interrupts or events for the Event System.

In a compare operation, the counter value is continuously compared to the values in the CCx registers. In the case of a match, the TC can request DMA transactions, or generate interrupts or events for the Event System. In waveform generator mode, these comparisons are used to set the waveform period or pulse width.

Capture operation can be enabled to perform input signal period and pulse width measurements, or to capture selectable edges from an IO pin or internal event from Event System.

Related Links

35.6.2.6.1. Waveform Output Operations

35.6.2 Basic Operation

35.6.2.1 Initialization

The following registers are enable-protected, meaning that they can only be written when the TC is disabled (CTRLA.ENABLE =0):

- Control A register (CTRLA), except the Enable (ENABLE) and Software Reset (SWRST) bits
- Drive Control register (DRVCTRL)
- Wave register (WAVE)
- Event Control register (EVCTRL)

Writing to Enable-Protected bits and setting the CTRLA.ENABLE bit can be performed in a single 32-bit access of the CTRLA register. Writing to Enable-Protected bits and clearing the CTRLA.ENABLE bit cannot be performed in a single 32-bit access.

Before enabling the TC, the peripheral must be configured by the following steps:

1. Enable the TC bus clock if not already enabled by default (CLK_TCx_APB).
2. Select 8-, 16- or 32-bit counter mode via the TC Mode bit group in the Control A register (CTRLA.MODE). The default mode is 16-bit.
3. Select one wave generation operation in the Waveform Generation Operation bit group in the WAVE register (WAVE.WAVEGEN).
4. If desired, the GCLK_TCx clock can be prescaled via the Prescaler bit group in the Control A register (CTRLA.PRESCALER).
   - If the prescaler is used, select a prescaler synchronization operation via the Prescaler and Counter Synchronization bit group in the Control A register (CTRLA.PRESYNC).
5. If desired, select one-shot operation by writing a '1' to the One-Shot bit in the Control B Set register (CTRLBSET.ONESHOT).
6. If desired, configure the counting direction 'down' (starting from the TOP value) by writing a '1' to the Counter Direction bit in the Control B register (CTRLBSET.DIR).
7. For capture operation, enable the individual channels to capture in the Capture Channel x Enable bit group in the Control A register (CTRLA.CAPTEN).
8. If desired, enable inversion of the waveform output or IO pin input signal for individual channels via the Invert Enable bit group in the Drive Control register (DRVCTRL.INVEN).

35.6.2.2 Enabling, Disabling, and Resetting

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

The TC is reset by writing a '1' to the Software Reset bit in the Control A register (CTRLA.SWRST). All registers in the TC, except DBGCTRL, will be reset to their initial state. See *CTRLA* from Related Links.

The TC should be disabled before the TC is reset in order to avoid undefined behavior.

Related Links

35.8.1. CTRLA

35.6.2.3 Prescaler Selection

The GCLK_TCx is fed into the internal prescaler.

The prescaler consists of a counter that counts up to the selected prescaler value, whereupon the output of the prescaler toggles.

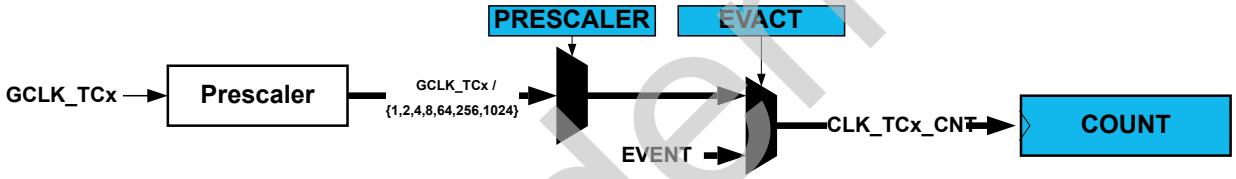
If the prescaler value is higher than one, the Counter Update condition can be optionally executed on the next GCLK_TCx clock pulse or the next prescaled clock pulse. For further details, refer to Prescaler (CTRLA.PRESCALER) and Counter Synchronization (CTRLA.PRESYNC) description.

Prescaler outputs from 1 to 1/1024 are available. For a complete list of available prescaler outputs, see the register description for the Prescaler bit group in the Control A register (CTRLA.PRESCALER).

Note: When counting events, the prescaler is bypassed.

The joint stream of prescaler ticks and event action ticks is called CLK_TCx_CNT.

Figure 35-2. Prescaler



35.6.2.4 Counter Mode

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

- COUNT8: The 8-bit TC has its own Period Value and Period Buffer Value registers (PER and PERBUF).
- COUNT16: 16-bit is the default counter mode. There is no dedicated period register in this mode.
- COUNT32: 32-bit mode is achieved by pairing two 16-bit TC peripherals. TC(2n) is paired with TC(n+1). When paired, the TC peripherals are configured using the registers of the even-numbered TC (TC0 or TC2 respectively).

  The TC bus clocks (CLK_TCx_APB) for both master and slave TCs need to be enabled.

  The odd-numbered partner (TC1 or TC3 respectively) will act as a slave, and the Slave bit in the Status register (STATUS.SLAVE) will be set. The register values of a slave will not reflect the registers of the 32-bit counter. Writing to any of the slave registers will not affect the 32-bit counter. Normal access to the slave COUNT and CCx registers is not allowed.

35.6.2.5 Counter Operations

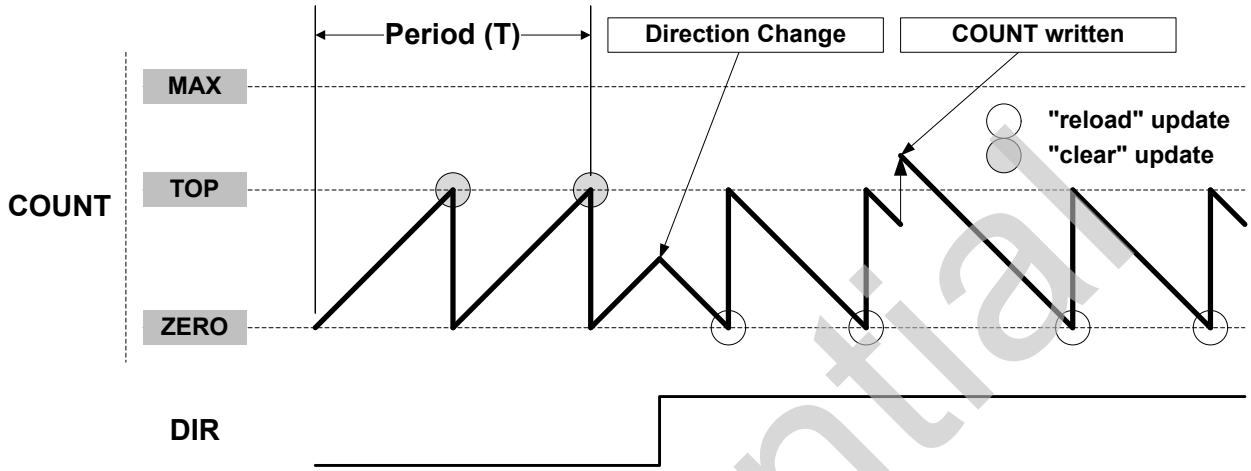
Depending on the mode of operation, the counter is cleared, reloaded, incremented, or decremented at each TC clock input (CLK_TCx_CNT). A counter clear or reload marks the end of the current counter cycle and the start of a new one.

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

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

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

Figure 35-3. Counter Operation



Due to asynchronous clock domains, the internal counter settings are written when the synchronization is complete. Normal operation must be used when using the counter as timer base for the capture channels.

35.6.2.5.1 Stop Command and Event Action

A Stop command can be issued from software by using Command bits in the Control B Set register (CTRLBSET.CMD = 0x2, STOP). When a Stop is detected while the counter is running, the counter will not retain its current value. All waveforms are cleared and the Stop bit in the Status register is set (STATUS.STOP).

35.6.2.5.2 Re-Trigger Command and Event Action

A re-trigger command can be issued from software by writing the Command bits in the Control B Set register (CTRLBSET.CMD = 0x1, RETRIGGER), or from event when a re-trigger event action is configured in the Event Control register (EVCTRL.EVACT = 0x1, RETRIGGER).

When the command is detected during counting operation, the counter will be reloaded or cleared, depending on the counting direction (CTRLBSET.DIR or CTRLBCLR.DIR). When the re-trigger command is detected while the counter is stopped, the counter will resume counting from the current value in the COUNT register.

Note: When a re-trigger event action is configured in the Event Action bits in the Event Control register (EVCTRL.EVACT=0x1, RETRIGGER), enabling the counter will not start the counter. The counter will start on the next incoming event and restart on corresponding following event.

35.6.2.5.3 Count Event Action

The TC can count events. When an event is received, the counter increases or decreases the value, depending on direction settings (CTRLBSET.DIR or CTRLBCLR.DIR). The count event action can be selected by the Event Action bit group in the Event Control register (EVCTRL.EVACT=0x2, COUNT).

Note: If this operation mode is selected, PWM generation is not supported.

35.6.2.5.4 Start Event Action

The TC can start counting operation on an event when previously stopped. In this configuration, the event has no effect if the counter is already counting. When the peripheral is enabled, the counter operation starts when the event is received or when a re-trigger software command is applied.

The Start TC on Event action can be selected by the Event Action bit group in the Event Control register (EVCTRL.EVACT=0x3, START).

35.6.2.6 Compare Operations

By default, the Compare/Capture channel is configured for compare operations.

When using the TC and the Compare/Capture Value registers (CCx) for compare operations, the counter value is continuously compared to the values in the CCx registers. This can be used for timer or for waveform operation.

The Channel x Compare Buffer (CCBUFx) registers provide double buffer capability. The double buffering synchronizes the update of the CCx register with the buffer value at the UPDATE condition or a forced update command (CTRLBSET.CMD=UPDATE). See *Double Buffering* from Related Links. The synchronization prevents the occurrence of odd-length, non-symmetrical pulses and ensures glitch-free output.

Related Links

35.6.2.7. Double Buffering

35.6.2.6.1 Waveform Output Operations

The compare channels can be used for waveform generation on output port pins. To make the waveform available on the connected pin, the following requirements must be fulfilled:

1. Choose a Waveform Generation mode in the Waveform Generation Operation bit in Waveform register (WAVE.WAVEGEN).
2. Optionally invert the waveform output WO[x] by writing the corresponding Output Waveform x Invert Enable bit in the Driver Control register (DRVCTRL.INVENx).
3. Configure the pins with the I/O Peripheral Pin Select (PPS). See *I/O Ports and Peripheral Pin Select (PPS)* from Related Links.
   Note: Event must not be used when the compare channel is set in waveform output operating mode.

The counter value is continuously compared with each CCx value. On a comparison match, the Match or Capture Channel x bit in the Interrupt Flag Status and Clear register (INTFLAG.MCx) will be set on the next zero-to-one transition of CLK_TC_CNT (see Normal Frequency Operation). An interrupt/and or event can be generated on comparison match if enabled. The same condition generates a DMA request.

There are four waveform configurations for the Waveform Generation Operation bit group in the Waveform register (WAVE.WAVEGEN). This will influence how the waveform is generated and impose restrictions on the top value. The configurations are:

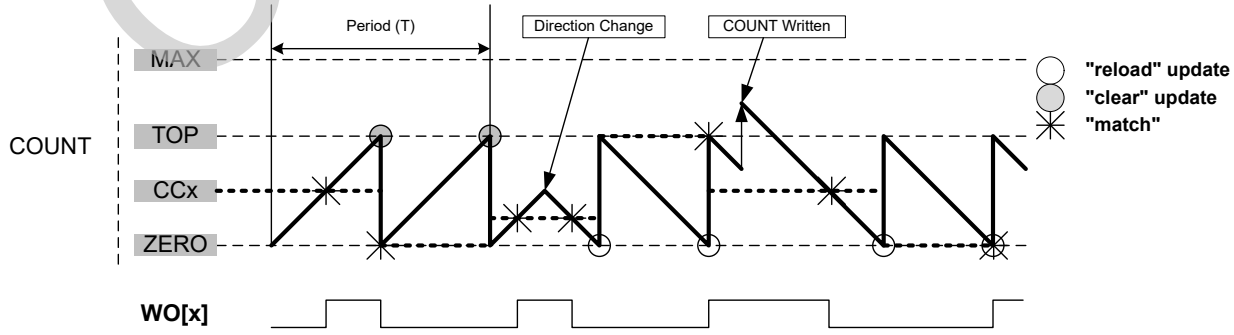
- Normal frequency (NFRQ)
- Match frequency (MFRQ)
- Normal pulse-width modulation (NPWM)
- Match pulse-width modulation (MPWM)

When using NPWM or NFRQ configuration, the TOP will be determined by the counter resolution. In 8-bit Counter mode, the Period register (PER) is used as TOP, and the TOP can be changed by writing to the PER register. In 16- and 32-bit Counter mode, TOP is fixed to the maximum (MAX) value of the counter.

Normal Frequency Generation (NFRQ)

For Normal Frequency Generation, the period time (T) is controlled by the period register (PER) for 8-bit Counter mode and MAX for 16- and 32-bit mode. The waveform generation output (WO[x]) is toggled on each compare match between COUNT and CCx, and the corresponding Match or Capture Channel x Interrupt Flag (INTFLAG.MCx) will be set.

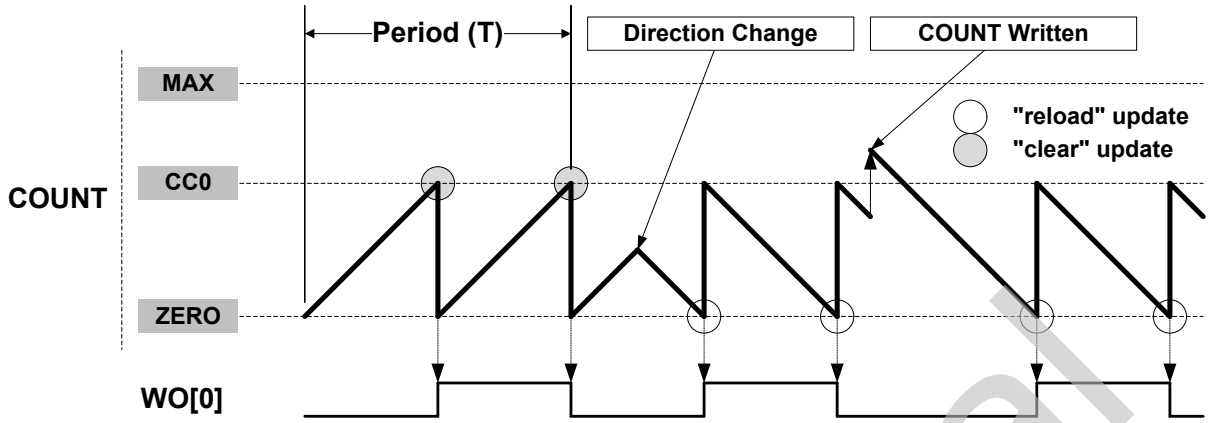
Figure 35-4. Normal Frequency Operation



Match Frequency Generation (MFRQ)

For Match Frequency Generation, the period time (T) is controlled by the CC0 register instead of PER or MAX. WO[0] toggles on each Update condition.

Figure 35-5. Match Frequency Operation



Normal Pulse-Width Modulation Operation (NPWM)
NPWM uses single-slope PWM generation.

For single-slope PWM generation, the period time (T) is controlled by the TOP value, and CCx controls the duty cycle of the generated waveform output. When up-counting, the WO[x] is set at start or compare match between the COUNT and TOP values, and cleared on compare match between COUNT and CCx register values. When down-counting, the WO[x] is cleared at start or compare match between the COUNT and ZERO values, and set on compare match between COUNT and CCx register values.

The following equation calculates the exact resolution for a single-slope PWM (R_{PWM_SS}) waveform:

$$R_{PWM\_SS} = \frac{\log(TOP+1)}{\log(2)}$$

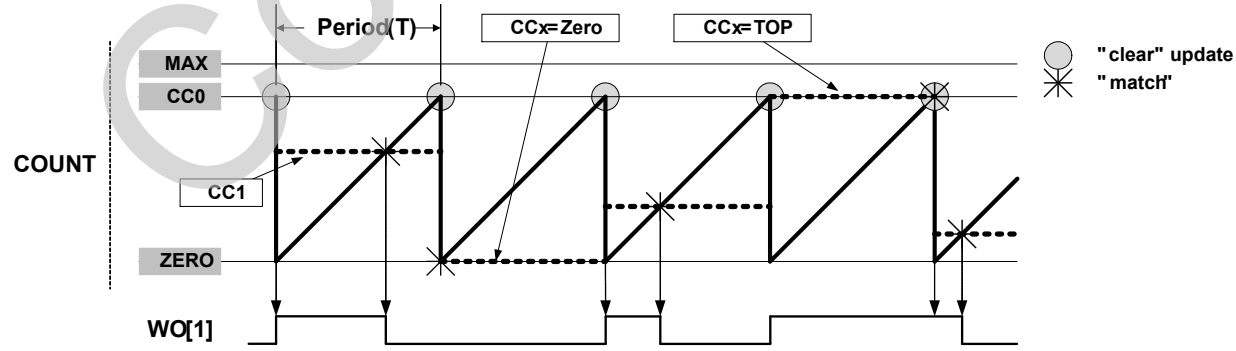
The PWM frequency (f_{PWM_SS}) depends on TOP value and the peripheral clock frequency (f_{GCLK_TC}), and can be calculated by the following equation:

$$f_{PWM\_SS} = \frac{f_{GCLK\_TC}}{N(TOP+1)}$$

Where N represents the prescaler divider used (1, 2, 4, 8, 16, 64, 256, 1024).

Match Pulse-Width Modulation Operation (MPWM)
In MPWM, the output of WO[1] is depending on CC1 as shown in the figure below. On every overflow/underflow, a one-TC-clock-cycle negative pulse is put out on WO[0] (not shown in the figure).

Figure 35-6. Match PWM Operation



The following table shows the Update Counter and Overflow Event/Interrupt Generation conditions in different operation modes.

Table 35-3. Counter Update and Overflow Event/interrupt Conditions in TC

Name	Operation	TOP	Update	Output Waveform		OVFIF/Event	
				On Match	On Update	Up	Down
NFRQ	Normal Frequency	PER	TOP/ ZERO	Toggle	Stable	TOP	ZERO
MFRQ	Match Frequency	CC0	TOP/ ZERO	Toggle	Stable	TOP	ZERO
NPWM	Single-slope PWM	PER	TOP/ ZERO	See description above.		TOP	ZERO
MPWM	Single-slope PWM	CC0	TOP/ ZERO	Toggle	Toggle	TOP	ZERO

Related Links

5. I/O Ports and Peripheral Pin Select (PPS)

35.6.2.7 Double Buffering

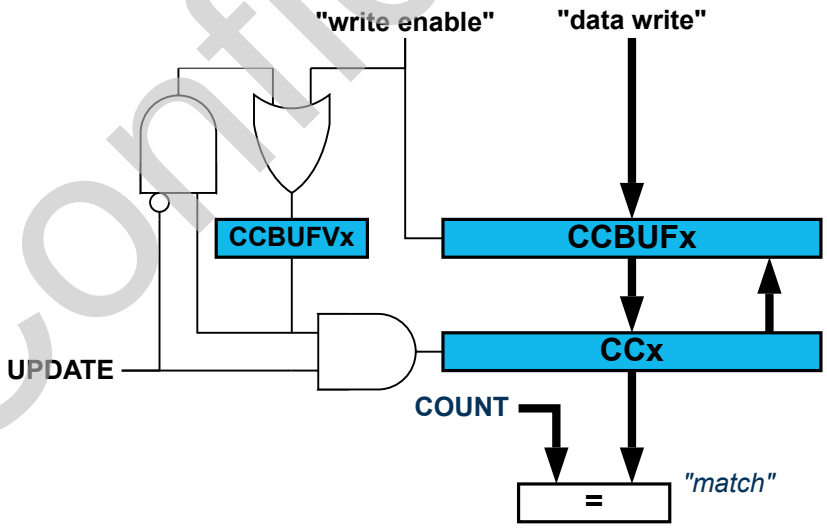
The Compare Channels (CCx) registers, and the Period (PER) register in 8-bit mode are double buffered. Each buffer register has a buffer valid bit (CCBUFVx or PERBUFV) in the STATUS register, which indicates that the buffer register contains a new valid value that can be copied into the corresponding register. As long as the respective buffer valid status flag (PERBUFV or CCBUFVx) are set to '1', related syncbusy bits are set (SYNCBUSY.PER or SYNCBUSY.CCx), a write to the respective PER/PERBUF or CCx/CCBUFx registers will generate a PAC error, and access to the respective PER or CCx register is invalid.

When the buffer valid flag bit in the STATUS register is '1' and the Lock Update bit in the CTRLB register is set to '0', (writing CTRLBCLR.LUPD to '1'), double buffering is enabled: the data from buffer registers will be copied into the corresponding register under hardware UPDATE conditions, then the buffer valid flags bit in the STATUS register are automatically cleared by hardware.

Note: The software update command (CTRLBSET.CMD=0x3) is acting independently of the LUPD value.

A compare register is double buffered as in the following figure.

Figure 35-7. Compare Channel Double Buffering



Both the registers (PER/CCx) and corresponding buffer registers (PERBUF/CCBUFx) are available in the I/O register map, and the double buffering feature is not mandatory. The double buffering is disabled by writing a '1' to CTRLBSET.LUPD.

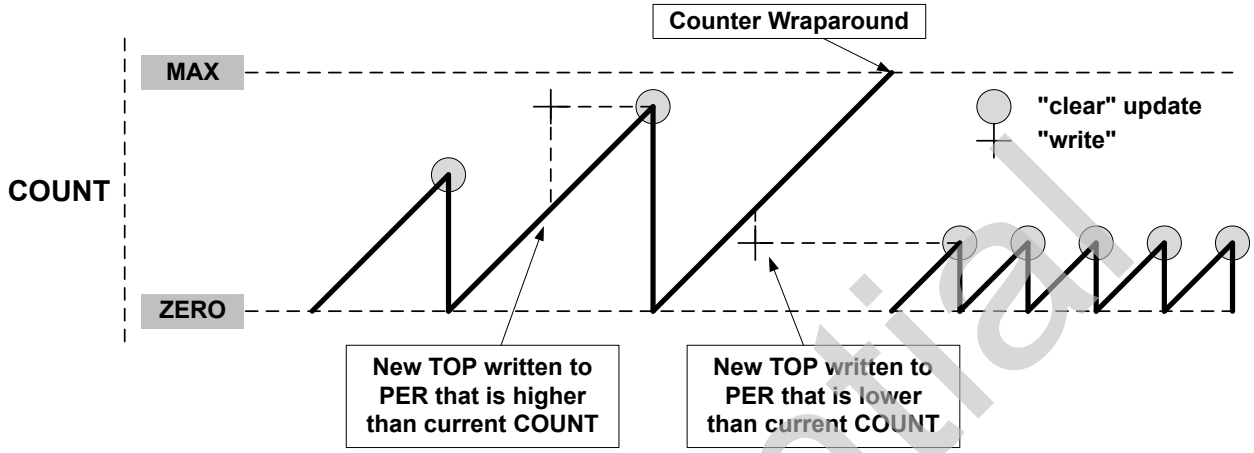
Note: In NFRQ, MFRQ or PWM, down-counting counter mode (CTRLBSET.DIR=1), when double buffering is enabled (CTRLBCLR.LUPD=1), PERBUF register is continuously copied into the PER independently of update conditions.

Changing the Period

The counter period can be changed by writing a new TOP value to the Period register (PER or CC0, depending on the waveform generation mode), which is available in 8-bit mode. Any period update on registers (PER or CCx) is effective after the synchronization delay.

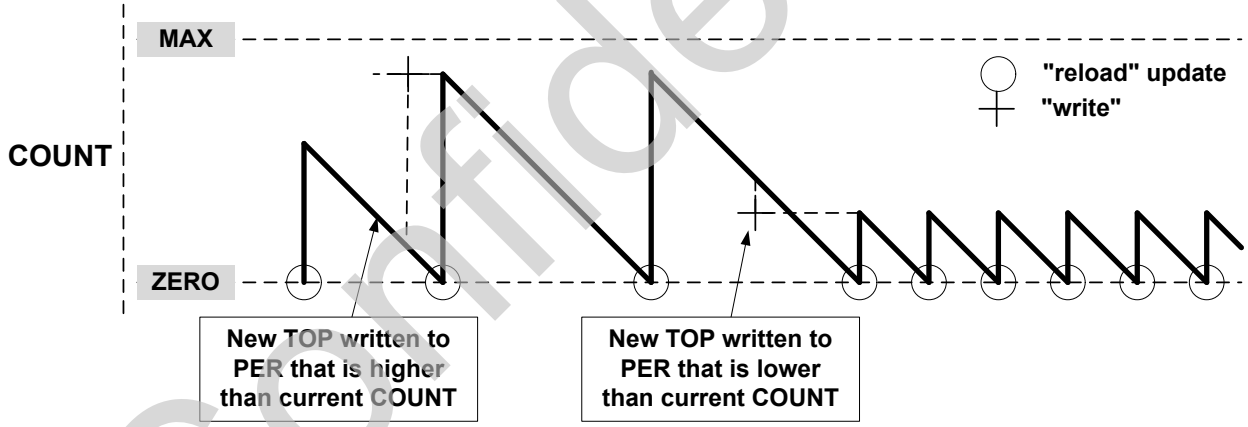
A counter wraparound can occur in any operation mode when up-counting without buffering (see the following figure).

Figure 35-8. Unbuffered Single-Slope Up-Counting Operation



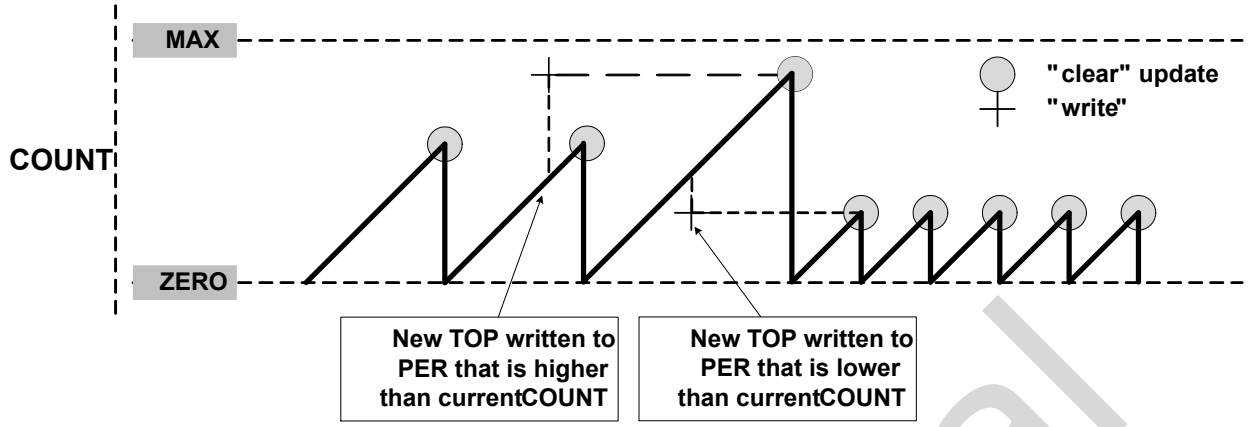
COUNT and TOP are continuously compared, so when a new TOP value that is lower than current COUNT is written to TOP, COUNT will wrap before a compare match.

Figure 35-9. Unbuffered Single-Slope Down-Counting Operation



When double buffering is used, the buffer can be written at any time and the counter will still maintain correct operation. The period register is always updated on the update condition, as shown in the following figure. This prevents wraparound and the generation of odd waveforms.

Figure 35-10. Changing the Period Using Buffering



35.6.2.8 Capture Operations

To enable and use capture operations, the corresponding Capture Channel x Enable bit in the Control A register (CTRLA.CAPTENx) must be written to '1'.

A capture trigger can be provided by input event line TC_EV or by asynchronous I/O pin WO[x] for each capture channel or by a TC event. To enable the capture from input event line, Event Input Enable bit in the Event Control register (EVCTRL.TCEI) must be written to '1'. To enable the capture from the I/O pin, the Capture On Pin x Enable bit in the CTRLA register (CTRLA.COPENx) must be written to '1'.

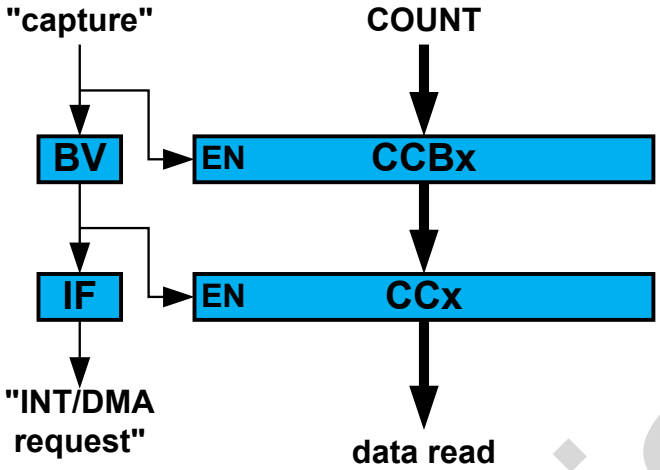
Notes:

1. Capture on I/Os is only possible in 'Event' and 'Time-Stamp' capture action modes. Other modes can only use internal events. (If I/Os toggling is needed in other modes, then the I/Os edge must be configured for generating internal events).
2. Capture on an event from the Event System is possible in 'Event', 'PPW/PWP/PW', and 'Time-Stamp' capture action modes. In this case, the event system channels must be configured to operate in asynchronous mode of operation.
3. Depending on CTRLA.COPENx, channel x can be configured for I/Os or internal event capture (both are mutually exclusive). One channel can be configured for I/Os capture while the other uses internal event capture.

By default, a capture operation is done when a rising edge is detected on the input signal. Capture on falling edge is available, its activation is depending on the input source:

- When the channel is used with a I/O pin, write a '1' to the corresponding Invert Enable bit in the Drive Control register (DRVCTRL.INVENx).
- When the channel is counting events from the Event System, write a '1' to the TC Event Input Invert Enable bit in Event Control register (EVCTRL.TCINV).

Figure 35-11. Capture Double Buffering



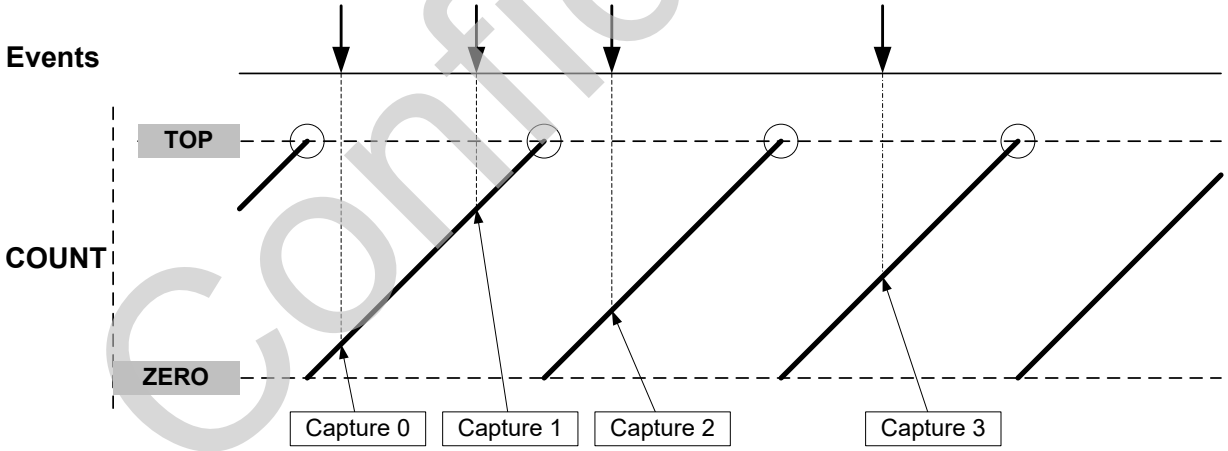
For input capture, the buffer register and the corresponding CCx act like a FIFO. When CCx is empty or read, any content in CCBUFx is transferred to CCx. The buffer valid flag is passed to set the CCx interrupt flag (IF) and generate the optional interrupt, event or DMA request. The CCBUFx register value can't be read, all captured data must be read from CCx register.

Note:
When up-counting (CTRLBSET.DIR = 0), counter values lower than '1' cannot be captured. To capture the full range including value '0', the TC must be in down-counting mode (CTRLBSET.DIR = 0).

35.6.2.8.1 Event Capture Action on Events or I/Os

The compare and capture channels can be used as input capture channels to capture events from the Event System or I/O pins and give them a timestamp. This mode is selected when EVTCTRL.EVACT is configured either as OFF, RETRIGGER, COUNT or START. The following figure shows four capture events for one capture channel.

Figure 35-12. Input Capture Timing



The TC can detect capture overflow of the input capture channels: When a new capture event is detected while the Capture Interrupt flag (INTFLAG.MCx) is still set, the new timestamp will not be stored and INTFLAG.ERR will be set.

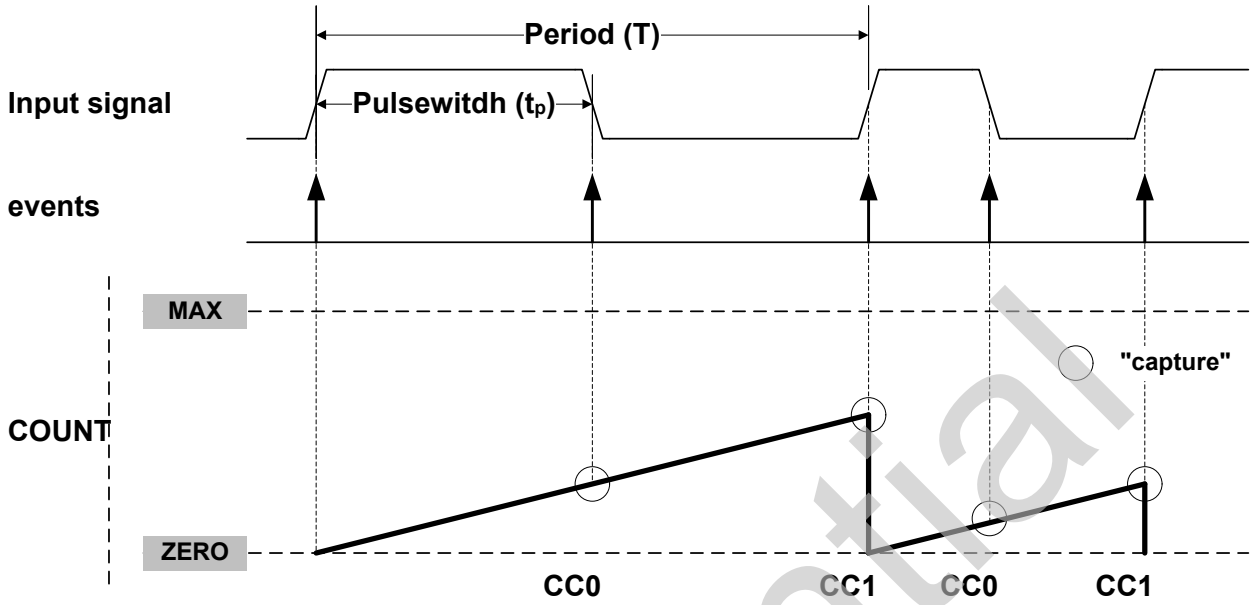
35.6.2.8.2 Period and Pulse-Width (PPW/PWP) Capture Action on Events

The TC can perform two input captures and restart the counter on one of the edges. This enables the TC to measure the pulse width and period and to characterize the frequency *f* and duty cycle of an input signal:

$$f = \frac{1}{T}$$

$$\text{dutyCycle} = \frac{t_p}{T}$$

Figure 35-13. PWP Capture



Selecting PWP in the Event Action bit group in the Event Control register (EVCTRL.EVACT) enables the TC to perform one capture action on the rising edge and another one on the falling edge. The period *T* will be captured into CC1 and the pulse width t_p in CC0. EVCTRL.EVACT = PPW (period and pulse-width) offers identical functionality, but will capture *T* into CC0 and t_p into CC1.

The TC Event Input Invert Enable bit in the Event Control register (EVCTRL.TCINV) is used to select whether the wraparound must occur on the rising edge or the falling edge. If EVCTRL.TCINV = 1, the wraparound will happen on the falling edge. In case pin capture is enabled, this can also be achieved by modifying the value of the DRVCTRL.INVENx bit.

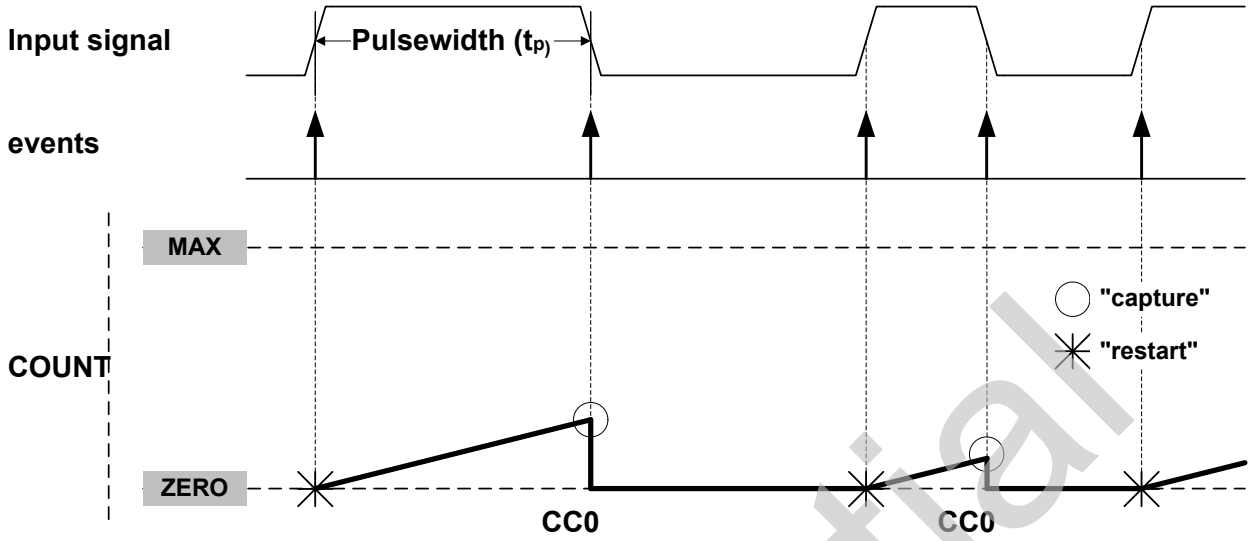
The TC can detect capture overflow of the input capture channels: When a new capture event is detected while the Capture Interrupt flag (INTFLAG.MCx) is still set, the new timestamp will not be stored and INTFLAG.ERR will be set.

Note: The corresponding capture is working only if the channel is enabled in capture mode (CTRLA.CAPTENx = 1). Consequently, both channels must be enabled to fully characterize the input.

35.6.2.8.3 Pulse-Width (PW) Capture Action on Events

The TC performs the input capture on the falling edge of the input signal. When the edge is detected, the counter value is cleared and the TC stops counting. When a rising edge is detected on the input signal, the counter restarts the counting operation. To enable the operation on opposite edges, the input signal to capture must be inverted (refer to EVCTRL.TCEINV).

Figure 35-14. Pulse-Width Capture on Channel 0



The TC can detect capture overflow of the input capture channels: When a new capture event is detected while the Capture Interrupt flag (INTFLAG.MCx) is still set, the new timestamp will not be stored and INTFLAG.ERR will be set.

35.6.3 Additional Features

35.6.3.1 One-Shot Operation

When one-shot is enabled, the counter automatically stops on the next Counter Overflow or Underflow condition. When the counter is stopped, the Stop bit in the Status register (STATUS.STOP) is automatically set and the waveform outputs are set to zero.

One-shot operation is enabled by writing a '1' to the One-Shot bit in the Control B Set register (CTRLBSET.ONESHOT), and disabled by writing a '1' to CTRLBCLR.ONESHOT. When enabled, the TC will count until an overflow or underflow occurs and stops counting operation. The one-shot operation can be restarted by a re-trigger software command, a re-trigger event, or a start event. When the counter restarts its operation, STATUS.STOP is automatically cleared.

35.6.3.2 Time-Stamp Capture on Events or I/Os

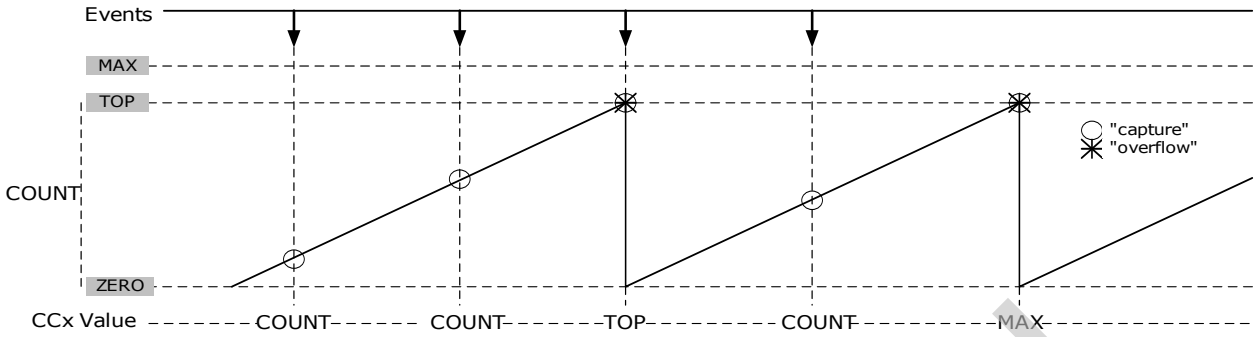
This feature is enabled when the Capture Time Stamp (STAMP) Event Action in Event Control register (EVCTRL.EVACT) is selected. The counter TOP value must be smaller than MAX.

When a capture event from the Event System or the I/O pin is detected, the COUNT value is copied into the corresponding Channel x Compare/Capture Value (CCx) register. In case of an overflow, the MAX value is copied into the corresponding CCx register.

When a valid captured value is present in the capture channel register, the corresponding Capture Channel x Interrupt Flag (INTFLAG.MCx) is set.

The timer/counter can detect capture overflow of the input capture channels: When a new capture event is detected while the Capture Channel interrupt flag (INTFLAG.MCx) is still set, the new time-stamp will not be stored and INTFLAG.ERR will be set.

Figure 35-15. Time Stamp



35.6.3.3 Minimum Capture (SAM C20/C21 N only)

The minimum capture is enabled by writing the CAPTMIN mode in the Channel n Capture Mode bits in the Control A register (CTRLA.CAPTMODEn = CAPTMIN).

CCx Content:

In CAPTMIN operations, CCx keeps the Minimum captured values. Before enabling this mode of capture, the user must initialize the corresponding CCx register value to a value different from zero. If the CCx register initial value is zero, no captures will be performed using the corresponding channel.

MCx Behaviour:

In CAPTMIN operation, capture is performed only when on capture event time, the counter value is lower than the last captured value. The MCx interrupt flag is set only when on capture event time, the counter value is upper or equal to the value captured on the previous event. So interrupt flag is set when a new absolute local Minimum value has been detected.

35.6.3.4 Maximum Capture (SAM C20/C21 N only)

The maximum capture is enabled by writing the CAPTMAX mode in the Channel n Capture Mode bits in the Control A register (CTRLA.CAPTMODEn = CAPTMAX).

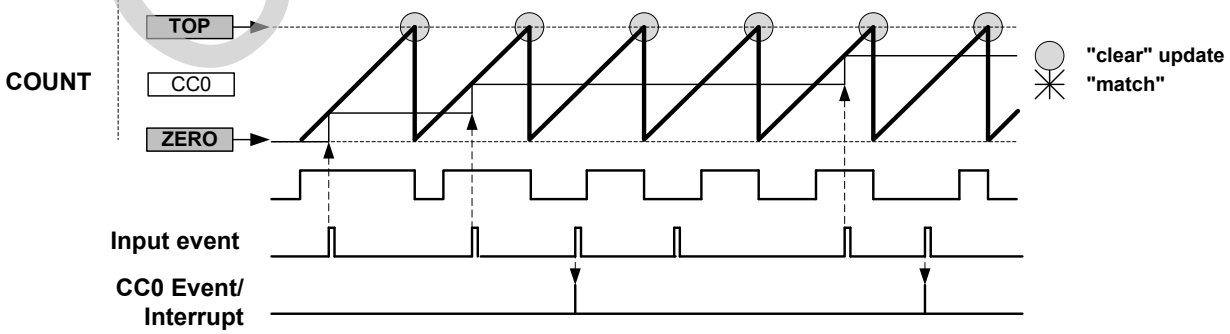
CCx Content:

In CAPTMAX operations, CCx keeps the Maximum captured values. Before enabling this mode of capture, the user must initialize the corresponding CCx register value to a value different from TOP. If the CCx register initial value is TOP, no captures will be performed using the corresponding channel.

MCx Behaviour:

In CAPTMAX operation, capture is performed only when on capture event time, the counter value is upper than the last captured value. The MCx interrupt flag is set only when on capture event time, the counter value is lower or equal to the value captured on the previous event. So interrupt flag is set when a new absolute local Maximum value has been detected.

Figure 35-16. Maximum Capture Operation with CC0 Initialized with ZERO Value



35.6.4 DMA Operation

The TC can generate the following DMA requests:

- Overflow (OVF): the request is set when an update condition (overflow, underflow or re-trigger) is detected, the request is cleared by hardware on DMA acknowledge.
- Match or Capture Channel x (MCx): for a compare channel, the request is set on each compare match detection, the request is cleared by hardware on DMA acknowledge. For a capture channel, the request is set when valid data is present in the CCx register, and cleared when CCx register is read.

35.6.5 Interrupts

The TC has the following interrupt sources:

- Overflow/Underflow (OVF)
- Match or Capture Channel x (MCx)
- Capture Overflow Error (ERR)

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG) is set when the interrupt condition occurs.

Each interrupt can be individually enabled by writing a '1' to the corresponding bit in the Interrupt Enable Set register (INTENSET), and disabled by writing a '1' to the corresponding bit in the Interrupt Enable Clear register (INTENCLR). The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until either the interrupt flag is cleared, the interrupt is disabled, or the TC is reset. See *INTFLAG* from Related Links for more details on how to clear the interrupt flags.

The TC has one common interrupt request line for all the interrupt sources. The user must read the INTFLAG register to determine which interrupt condition is present.

Note that interrupts must be globally enabled for interrupt requests to be generated. See *Nested Vector Interrupt Controller (NVIC)* from Related Links.

Related Links

8.2. Nested Vector Interrupt Controller (NVIC)

35.8.7. INTFLAG

35.6.6 Events

The TC can generate the following output events:

- Overflow/Underflow (OVF)
- Match or Capture Channel x (MCX0-1)

Writing a '1' to an Event Output bit in the Event Control register (EVCTRL.MCEOx) enables the corresponding output event. The output event is disabled by writing EVCTRL.MCEOx='0'.

One of the following event actions can be selected by the Event Action bit group in the Event Control register (EVCTRL.EVACT):

- Disable event action (OFF)
- Start TC (START)
- Re-trigger TC (RETRIGGER)
- Count on event (COUNT)
- Capture time stamp (STAMP)
- Capture Period (PPW and PWP)
- Capture Pulse Width (PW)

Writing a '1' to the TC Event Input bit in the Event Control register (EVCTRL.TCEI) enables input events (EVU0-2) to the TC. Writing a '0' to this bit disables input events to the TC. The TC requires only asynchronous event inputs. See *Event System (EVSYS)* from Related Links for additional information on configuring the asynchronous events.

Related Links

26. Event System (EVSYS)

35.6.7 Sleep Mode Operation

The TC can be configured to operate in any sleep mode (Idle, Standby Sleep). To be able to run in standby sleep mode, the RUNSTDBY bit in the Control A register (CTRLA.RUNSTDBY) must be '1'. This peripheral can wake up the device from any sleep mode using interrupts or perform actions through the Event System.

If the On Demand bit in the Control A register (CTRLA.ONDEMAND) is written to '1', the module stops requesting its peripheral clock when the STOP bit in STATUS register (STATUS.STOP) is set to '1'. When a re-trigger or start condition is detected, the TC requests the clock before the operation starts.

35.6.8 Synchronization

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

The following bits are synchronized when written:

- Software Reset and Enable bits in Control A register (CTRLA.SWRST and CTRLA.ENABLE)
- Capture Channel Buffer Valid bit in STATUS register (STATUS.CCBUFVx)

The following registers are synchronized when written:

- Control B Clear and Control B Set registers (CTRLBCLR and CTRLBSET)
- Count Value register (COUNT)
- Period Value and Period Buffer Value registers (PER and PERBUF)
- Channel x Compare/Capture Value and Channel x Compare/Capture Buffer Value registers (CCx and CCBUFx)

The following registers are synchronized when read:

- Count Value register (COUNT): synchronization is done on demand through READSYNC command (CTRLBSET.CMD)
- Control B Clear and Control B Set registers (CTRLBCLR and CTRLBSET)
- Channel x Compare/Capture Value (CCx)

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

Required read synchronization is denoted by the "Read-Synchronized" property in the register description.

35.7 Register Summary

See *TCx (x = 0 to 7)* module in the *Product Memory Mapping Overview* from Related Links for base address based on the TC instant used.

Note: All registers in this table have corresponding CLR, SET and INV registers at its virtual address, plus an offset of 0x4, 0x8 and 0xC, respectively. See *CLR, SET, and INV Registers* from Related Links.

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x00	CTRLA	7:0	ONDEMAND	RUNSTDBY	PRESCSYNC[1:0]		MODE[1:0]		ENABLE	SWRST
		15:8	DMAOS				ALOCK	PRESCALER[2:0]		
		23:16			COPEN1	COPEN0			CAPTEN1	CAPTEN0
		31:24				CAPTMODE1[1:0]			CAPTMODE0[1:0]	
0x04	CTRLBCLR	7:0	CMD[2:0]					ONESHOT	LUPD	DIR
0x05	CTRLBSET	7:0	CMD[2:0]					ONESHOT	LUPD	DIR
0x06	EVCTRL	7:0			TCEI	TCINV		EVACT[2:0]		
		15:8			MCEO1	MCEO0				OVFEO
0x08	INTENCLR	7:0			MC1	MC0			ERR	OVF
0x09	INTENSET	7:0			MC1	MC0			ERR	OVF
0x0A	INTFLAG	7:0			MC1	MC0			ERR	OVF
0x0B	STATUS	7:0			CCBUFV1	CCBUFV0	PERBUFV		SLAVE	STOP
0x0C	WAVE	7:0							WAVEGEN[1:0]	
0x0D	DRVCTRL	7:0							INVEN1	INVEN0
0x0E	Reserved									
0x0F	DBGCTRL	7:0								DBGRUN
0x10	SYNCBUSY	7:0	CC1	CC0	PER	COUNT	STATUS	CTRLB	ENABLE	SWRST
0x11 ... 0x13	Reserved									
0x14	COUNT	7:0	COUNT[7:0]							
0x15 ... 0x1A	Reserved									
0x1B	PER	7:0	PER[7:0]							
0x1C	CC0	7:0	CC[7:0]							
0x1D	CC1	7:0	CC[7:0]							
0x1E ... 0x2E	Reserved									
0x2F	PERBUF	7:0	PERBUF[7:0]							
0x30	CCBUF0	7:0	CCBUF[7:0]							
0x31	CCBUF1	7:0	CCBUF[7:0]							

Related Links

5.4.1.9. CLR, SET and INV Registers

7. Product Memory Mapping Overview

35.8 Register Description - 8-bit Mode

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.

Some registers are synchronized when read and/or written. Synchronization is denoted by the “Write-Synchronized” or the “Read-Synchronized” property in each individual register description.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable protection is denoted by the “Enable-Protected” property in each individual register description.

Following conventions are used in the register description:

- – R = Readable bit
- – W = Writable bit
- – U = Unimplemented bit, read as ‘0’
- – -n = Value at POR
- – ‘1’ = Bit is set
- – ‘0’ = Bit is cleared
- – x = Bit is unknown
- HS = Hardware Set
- HC = Hardware Cleared

35.8.1 Control A

Name: CTRLA
Offset: 0x00
Reset: 0x00000000
Property: PAC Write-Protection, Write-Synchronized, Enable-Protected

Bit	31	30	29	28	27	26	25	24
				CAPTMODE1[1:0]			CAPTMODE0[1:0]	
Access				R/W	R/W		R/W	R/W
Reset				0	0		0	0

Bit	23	22	21	20	19	18	17	16
			COPEN1	COPEN0			CAPTEN1	CAPTEN0
Access			R/W	R/W			R/W	R/W
Reset			0	0			0	0

Bit	15	14	13	12	11	10	9	8
	DMAOS				ALOCK	PRESCALER[2:0]		
Access	R/W				R/W	R/W	R/W	R/W
Reset	0				0	0	0	0

Bit	7	6	5	4	3	2	1	0
	ONDEMAND	RUNSTDBY	PRESCSYNC[1:0]		MODE[1:0]		ENABLE	SWRST
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	W
Reset	0	0	0	0	0	0	0	0

Bits 28:27 – CAPTMODE1[1:0] Capture mode Channel 1
These bits select the channel 1 capture mode.

Value	Name	Description
0x0	DEFAULT	Default capture
0x1	CAPTMIN	Minimum capture
0x2	CAPTMAX	Maximum capture
0x3		Reserved

Bits 25:24 – CAPTMODE0[1:0] Capture mode Channel 0
These bits select the channel 0 capture mode.

Value	Name	Description
0x0	DEFAULT	Default capture
0x1	CAPTMIN	Minimum capture
0x2	CAPTMAX	Maximum capture
0x3		Reserved

Bits 20, 21 – COPENx Capture On Pin x Enable [x=1..0]
Bit x of COPEN[1:0] selects the trigger source for capture operation, either events or I/O pin input.
This bit is not synchronized.

Value	Description
0	Event from Event System is selected as trigger source for capture operation on channel x.
1	I/O pin is selected as trigger source for capture operation on channel x.

Bits 16, 17 – CAPTENx Capture Channel x Enable [x=1..0]
Bit x of CAPTEN[1:0] selects whether channel x is a capture or a compare channel.
These bits are not synchronized.

Value	Description
0	CAPTEN disables capture on channel x.
1	CAPTEN enables capture on channel x.

Bit 15 – DMAOS DMA One-Shot Trigger Mode

This bit enables the DMA One-shot Trigger Mode.
Writing a '1' to this bit will generate a DMA trigger on TC cycle following a TC_CTRLBSET_CMD_DMAOS command.
Writing a '0' to this bit will generate DMA triggers on each TC cycle.
This bit is not synchronized.

Bit 11 – ALOCK Auto Lock

When this bit is set, Lock bit update (LUPD) is set to '1' on each overflow/underflow or re-trigger event.
This bit is not synchronized.

Value	Description
0	The LUPD bit is not affected on overflow/underflow, and re-trigger event.
1	The LUPD bit is set on each overflow/underflow or re-trigger event.

Bits 10:8 – PRESCALER[2:0] Prescaler

These bits select the counter prescaler factor.
These bits are not synchronized.

Value	Name	Description
0x0	DIV1	Prescaler: GCLK_TC
0x1	DIV2	Prescaler: GCLK_TC/2
0x2	DIV4	Prescaler: GCLK_TC/4
0x3	DIV8	Prescaler: GCLK_TC/8
0x4	DIV16	Prescaler: GCLK_TC/16
0x5	DIV64	Prescaler: GCLK_TC/64
0x6	DIV256	Prescaler: GCLK_TC/256
0x7	DIV1024	Prescaler: GCLK_TC/1024

Bit 7 – ONDEMAND Clock On Demand

This bit selects the clock requirements when the TC is stopped.
In standby mode, if the Run in Standby bit (CTRLA.RUNSTDBY) is '0', ONDEMAND is forced to '0'.
This bit is not synchronized.

Value	Description
0	The On Demand is disabled. If On Demand is disabled, the TC will continue to request the clock when its operation is stopped (STATUS.STOP=1).
1	The On Demand is enabled. When On Demand is enabled, the stopped TC will not request the clock. The clock is requested when a software re-trigger command is applied or when an event with start/re-trigger action is detected.

Bit 6 – RUNSTDBY Run in Standby

This bit is used to keep the TC running in standby mode.
This bit is not synchronized.

Value	Description
0	The TC is halted in standby.
1	The TC continues to run in standby.

Bits 5:4 – PRESCSYNC[1:0] Prescaler and Counter Synchronization

These bits select whether the counter must wrap around on the next GCLK_TCx clock or the next prescaled GCLK_TCx clock. It also makes it possible to reset the prescaler.
These bits are not synchronized.

Value	Name	Description
0x0	GCLK	Reload or reset the counter on next generic clock
0x1	PRESC	Reload or reset the counter on next prescaler clock
0x2	RESYNC	Reload or reset the counter on next generic clock. Reset the prescaler counter
0x3	-	Reserved

Bits 3:2 – MODE[1:0] Timer Counter Mode

These bits select the counter mode.
These bits are not synchronized.

Value	Name	Description
0x0	COUNT16	Counter in 16-bit mode
0x1	COUNT8	Counter in 8-bit mode
0x2	COUNT32	Counter in 32-bit mode
0x3	-	Reserved

Bit 1 – ENABLE Enable

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

This bit is not enable-protected.

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

Bit 0 – SWRST Software Reset

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

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

Writing a '1' to CTRLA.SWRST will always take precedence; all other writes in the same write-operation will be discarded.

This bit is not enable-protected.

35.8.2 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 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.

Bit 0 – DIR Counter Direction
This bit is used to change the direction of the counter.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the bit and make the counter count down.

Value	Description
0	The timer/counter is counting up (incrementing).
1	The timer/counter is counting down (decrementing).

35.8.3 Control B Set

Name: CTRLBSET
Offset: 0x05
Reset: 0x00
Property: PAC Write-Protection, Read-synchronized, Write-Synchronized

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

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 value different from 0x0 to these bits will issue a command for execution.

Value	Name	Description
0x0	NONE	No action
0x1	RETRIGGER	Force a start, restart or retrigger
0x2	STOP	Force a stop
0x3	UPDATE	Force update of double buffered registers
0x4	READSYNC	Force a read synchronization of COUNT

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 enable 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 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.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the LUPD bit.
This bit has no effect when input capture operation is enabled.

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.

Bit 0 – DIR Counter Direction
This bit is used to change the direction of the counter.
Writing a '0' to this bit has no effect
Writing a '1' to this bit will set the bit and make the counter count down.

Value	Description
0	The timer/counter is counting up (incrementing).
1	The timer/counter is counting down (decrementing).

35.8.4 Event Control

Name: EVCTRL
Offset: 0x06
Reset: 0x0000
Property: PAC Write-Protection, Enable-Protected

Bit	15	14	13	12	11	10	9	8
			MCEO1	MCEO0				OVFEO
Access			R/W	R/W				R/W
Reset			0	0				0

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

Bits 12, 13 – MCEOx Match or Capture Channel x Event Output Enable [x = 1..0]
These bits enable the generation of an event for every match or capture on channel x.

Value	Description
0	Match/Capture event on channel x is disabled and will not be generated.
1	Match/Capture event on channel x is enabled and will be generated for every compare/capture.

Bit 8 – OVFEO Overflow/Underflow Event Output Enable
This bit enables the Overflow/Underflow event. When enabled, an event will be generated when the counter overflows/underflows.

Value	Description
0	Overflow/Underflow event is disabled and will not be generated.
1	Overflow/Underflow event is enabled and will be generated for every counter overflow/underflow.

Bit 5 – TCEI TC Event Enable
This bit is used to enable asynchronous input events to the TC.

Value	Description
0	Incoming events are disabled.
1	Incoming events are enabled.

Bit 4 – TCINV TC Inverted Event Input Polarity
This bit inverts the asynchronous input event source.

Value	Description
0	Input event source is not inverted.
1	Input event source is inverted.

Bits 2:0 – EVACT[2:0] Event Action
These bits define the event action the TC will perform on an event.

Value	Name	Description
0x0	OFF	Event action disabled
0x1	RETRIGGER	Start, restart or retrigger TC on event
0x2	COUNT	Count on event
0x3	START	Start TC on event
0x4	STAMP	Time stamp capture
0x5	PPW	Period captured in CC0, pulse width in CC1
0x6	PWP	Period captured in CC1, pulse width in CC0
0x7	PW	Pulse width capture

35.8.5 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

Bits 4, 5 – MCx Match or Capture Channel x Interrupt Enable [x = 1..0]
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 these bits 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 these bits 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.

35.8.6 Interrupt Enable Set

Name: INTENSET
Offset: 0x09
Reset: 0x00
Property: PAC Write-Protection

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

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

Bits 4, 5 – MCx Match or Capture Channel x Interrupt Enable [x = 1..0]
Writing a '0' to these bits has no effect.
Writing a '1' to MCx will set the corresponding Match or Capture Channel x Interrupt Enable bit, which enables 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 Enable
Writing a '0' to these bits has no effect.
Writing a '1' to this bit will set the Error Interrupt Enable bit, which enables the Error interrupt.

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

Bit 0 – OVF Overflow Interrupt Enable
Writing a '0' to these bits has no effect.
Writing a '1' to this bit will set the Overflow Interrupt Enable bit, which enables the Overflow interrupt request.

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

35.8.7 Interrupt Flag Status and Clear

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

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

Bits 4, 5 – MCx Match or Capture Channel x [x = 1..0]
This flag is set on a comparison match, or when the corresponding CCx register contains a valid capture value. This flag is set on the next CLK_TC_CNT cycle, and will generate an interrupt request if the corresponding Match or Capture Channel x Interrupt Enable bit in the Interrupt Enable Set register (INTENSET.MCx) is '1'.
Writing a '0' to these bits has no effect.
Writing a '1' to one of these bits will clear the corresponding Match or Capture Channel x interrupt flag
In capture operation, this flag is automatically cleared when CCx register is read.

Bit 1 – ERR Error Interrupt Flag
This flag is set when a new capture occurs on a channel while the corresponding Match or Capture Channel x interrupt flag is set, in which case there is no place to store the new capture.
Writing a '0' to these bits has no effect.
Writing a '1' to this bit clears the Error interrupt flag.

Bit 0 – OVF Overflow Interrupt Flag
This flag is set on the next CLK_TC_CNT cycle after an overflow condition occurs, and will generate an interrupt request if INTENCLR.OVF or INTENSET.OVF is '1'.
Writing a '0' to these bits has no effect.
Writing a '1' to this bit clears the Overflow interrupt flag.

35.8.8 Status

Name: STATUS
Offset: 0x0B
Reset: 0x01
Property: Read-Synchronized

Bit	7	6	5	4	3	2	1	0
			CCBUFV1	CCBUFV0	PERBUFV		SLAVE	STOP
Access			R/W	R/W	R/W		R	R
Reset			0	0	0		0	1

Bits 4, 5 – CCBUFVx Channel x Compare or Capture Buffer Valid [x = 1..0]
For a compare channel x, the bit x is set when a new value is written to the corresponding CCBUFx register. The bit x is cleared by writing a '1' to it when CTRLB.LUPD is set, or it is cleared automatically by hardware on UPDATE condition.
For a capture channel x, the bit x is set when a valid capture value is stored in the CCBUFx register. The bit x is cleared automatically when the CCx register is read.

Bit 3 – PERBUFV Period Buffer Valid
This bit is set when a new value is written to the PERBUF register. The bit is cleared by writing '1' to the corresponding location when CTRLB.LUPD is set, or automatically cleared by hardware on UPDATE condition. This bit is available only in 8-bit mode and will always read zero in 16- and 32-bit modes.

Bit 1 – SLAVE Client Status Flag
This bit is only available in 32-bit mode on the Client TC (i.e., TC1, TC3, TC5 and/or TC7). The bit is set when the associated Host TC (TC0, TC2, TC4 and/or TC6, respectively) is set to run in 32-bit mode.

Bit 0 – STOP Stop Status Flag
This bit is set when the TC is disabled, on a Stop command, or on an overflow/underflow condition when the One-Shot bit in the Control B Set register (CTRLBSET.ONESHOT) is '1'.

Value	Description
0	Counter is running.
1	Counter is stopped.

35.8.9 Waveform Generation Control

Name: WAVE
Offset: 0x0C
Reset: 0x00
Property: PAC Write-Protection, Enable-Protected

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

Bits 1:0 – WAVEGEN[1:0] Waveform Generation Mode
These bits select the waveform generation operation. They affect the top value, as shown in Waveform Output Operations. They also control whether frequency or PWM waveform generation must be used. The waveform generation operations are explained in Waveform Output Operations. See *Waveform Output Operations* from Related Links.
These bits are not synchronized.

Value	Name	Operation	Top Value	Output Waveform on Match	Output Waveform on Wraparound
0x0	NFRQ	Normal frequency	PER[1] / Max	Toggle	No action
0x1	MFRQ	Match frequency	CC0	Toggle	No action
0x2	NPWM	Normal PWM	PER[1] / Max	Set	Clear
0x3	MPWM	Match PWM	CC0	Set	Clear

1. This depends on the TC mode: In 8-bit mode, the top value is the Period Value register (PER). In 16- and 32-bit mode, it is the respective MAX value.

Related Links
35.6.2.6.1. Waveform Output Operations

35.8.10 Driver Control

Name: DRVCTRL
Offset: 0x0D
Reset: 0x00
Property: PAC Write-Protection, Enable-Protected

Bit	7	6	5	4	3	2	1	0
							INVEN1	INVEN0
Access							R/W	R/W
Reset							0	0

Bits 0, 1 – INVENx Output Waveform x Invert Enable [x=1..0]
Bit x of INVEN[1:0] selects inversion of the output or capture trigger input of channel x.

Value	Description
0	Disable inversion of the WO[x] output and IO input pin.
1	Enable inversion of the WO[x] output and IO input pin.

35.8.11 Debug Control

Name: DBGCTRL
Offset: 0x0F
Reset: 0x00
Property: PAC Write-Protection

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

Bit 0 – DBGRUN Run in Debug Mode

This bit is not affected by a software Reset, and must not be changed by software while the TC is enabled.

Value	Description
0	The TC is halted when the device is halted in debug mode.
1	The TC continues normal operation when the device is halted in debug mode.

35.8.12 Synchronization Busy

Name: SYNCBUSY
Offset: 0x10
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	CC1	CC0	PER	COUNT	STATUS	CTRLB	ENABLE	SWRST
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 6, 7 – CCx Compare/Capture Channel x Synchronization Busy [x=0..1]
For details on CC channels number, refer to each TC feature list.
This bit is set when the synchronization of CCx between clock domains is started.
This bit is also set when the CCBUFx is written, and cleared on update condition. The bit is automatically cleared when the STATUS.CCBUFx bit is cleared.

Bit 5 – PER PER Synchronization Busy
This bit is cleared when the synchronization of PER between the clock domains is complete.
This bit is set when the synchronization of PER between clock domains is started.
This bit is also set when the PER is written, and cleared on update condition. The bit is automatically cleared when the STATUS.PERBUF bit is cleared.

Bit 4 – COUNT COUNT Synchronization Busy
This bit is cleared when the synchronization of COUNT between the clock domains is complete.
This bit is set when the synchronization of COUNT between clock domains is started.

Bit 3 – STATUS STATUS Synchronization Busy
This bit is cleared when the synchronization of STATUS between the clock domains is complete.
This bit is set when a '1' is written to the Capture Channel Buffer Valid status flags (STATUS.CCBUFVx) and the synchronization of STATUS between clock domains is started.

Bit 2 – CTRLB CTRLB Synchronization Busy
This bit is cleared when the synchronization of CTRLB between the clock domains is complete.
This bit is set when the synchronization of CTRLB between clock domains is started.

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

Bit 0 – SWRST SWRST Synchronization Busy
This bit is cleared when the synchronization of SWRST bit between the clock domains is complete.
This bit is set when the synchronization of SWRST bit between clock domains is started.
Note: During a SWRST, access to registers/bits without SWRST are disallowed until SYNCBUSY.SWRST cleared by hardware.

35.8.13 Counter Value, 8-bit Mode

Name: COUNT
Offset: 0x14
Reset: 0x00
Property: PAC Write-Protection, Write-Synchronized, Read-Synchronized

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

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 7:0 – COUNT[7:0] Counter Value
These bits contain the current counter value.

35.8.14 Period Value, 8-bit Mode

Name: PER
Offset: 0x1B
Reset: 0xFF
Property: Write-Synchronized

Bit	7	6	5	4	3	2	1	0
	PER[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	1

Bits 7:0 – PER[7:0] Period Value
These bits hold the value of the Period Buffer register PERBUF. The value is copied to PER register on UPDATE condition.

35.8.15 Channel x Compare/Capture Value, 8-bit Mode

Name: CCx
Offset: 0x1C + x*0x01 [x=0..1]
Reset: 0x00
Property: Write-Synchronized

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 7:0 – CC[7:0] Channel x Compare/Capture Value
These bits contain the compare/capture value in 8-bit TC mode. In Match frequency (MFRQ) or Match PWM (MPWM) waveform operation (WAVE.WAVEGEN), the CC0 register is used as a period register.

35.8.16 Period Buffer Value, 8-bit Mode

Name: PERBUF
Offset: 0x2F
Reset: 0xFF
Property: Write-Synchronized

Bit	7	6	5	4	3	2	1	0
	PERBUF[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	1

Bits 7:0 – PERBUF[7:0] Period Buffer Value
These bits hold the value of the period buffer register. The value is copied to PER register on UPDATE condition.

35.8.17 Channel x Compare Buffer Value, 8-bit Mode

Name: CCBUFx
Offset: 0x30 + x*0x01 [x=0..1]
Reset: 0x00
Property: Write-Synchronized

Bit	7	6	5	4	3	2	1	0
	CCBUF[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 7:0 – CCBUF[7:0] Channel x Compare Buffer Value
These bits hold the value of the Channel x Compare Buffer Value. When the buffer valid flag is '1' and double buffering is enabled (CTRLBCLR.LUPD=1), the data from buffer registers will be copied into the corresponding CCx register under UPDATE condition (CTRLBSET.CMD=0x3), including the software update command.

35.9 Register Summary

See *TCx (x = 0 to 7)* module in the *Product Memory Mapping Overview* from Related Links for base address based on the TC instant used.

Note: All registers in this table have corresponding CLR, SET and INV registers at its virtual address, plus an offset of 0x4, 0x8 and 0xC, respectively. See *CLR, SET, and INV Registers* from Related Links.

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x00	CTRLA	7:0	ONDEMAND	RUNSTDBY	PRESCSYNC[1:0]		MODE[1:0]		ENABLE	SWRST
		15:8	DMAOS				ALOCK	PRESCALER[2:0]		
		23:16			COPEN1	COPEN0			CAPTEN1	CAPTEN0
		31:24				CAPTMODE1[1:0]			CAPTMODE0[1:0]	
0x04	CTRLBCLR	7:0	CMD[2:0]					ONESHOT	LUPD	DIR
0x05	CTRLBSET	7:0	CMD[2:0]					ONESHOT	LUPD	DIR
0x06	EVCTRL	7:0			TCEI	TCINV		EVACT[2:0]		
		15:8			MCEO1	MCEO0				OVFEO
0x08	INTENCLR	7:0			MC1	MC0			ERR	OVF
0x09	INTENSET	7:0			MC1	MC0			ERR	OVF
0x0A	INTFLAG	7:0			MC1	MC0			ERR	OVF
0x0B	STATUS	7:0			CCBUFV1	CCBUFV0	PERBUFV		SLAVE	STOP
0x0C	WAVE	7:0							WAVEGEN[1:0]	
0x0D	DRVCTRL	7:0							INVEN1	INVEN0
0x0E	Reserved									
0x0F	DBGCTRL	7:0								DBGRUN
0x10	SYNCBUSY	7:0	CC1	CC0	PER	COUNT	STATUS	CTRLB	ENABLE	SWRST
0x11 ... 0x13	Reserved									
0x14	COUNT	7:0	COUNT[7:0]							
		15:8	COUNT[15:8]							
0x16 ... 0x19	Reserved									
0x1A	PER	7:0	PER[7:0]							
		15:8	PER[15:8]							
0x1C	CC0	7:0	CC[7:0]							
		15:8	CC[15:8]							
0x1E	CC1	7:0	CC[7:0]							
		15:8	CC[15:8]							
0x20 ... 0x2D	Reserved									
0x2E	PERBUF	7:0	PERBUF[7:0]							
		15:8	PERBUF[15:8]							
0x30	CCBUF0	7:0	CCBUF[7:0]							
		15:8	CCBUF[15:8]							
0x32	CCBUF1	7:0	CCBUF[7:0]							
		15:8	CCBUF[15:8]							

Related Links

5.4.1.9. CLR, SET and INV Registers

7. Product Memory Mapping Overview

35.10 Register Description - 16-bit Mode

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.

Some registers are synchronized when read and/or written. Synchronization is denoted by the “Write-Synchronized” or the “Read-Synchronized” property in each individual register description.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable protection is denoted by the “Enable-Protected” property in each individual register description.

Following conventions are used in the register description:

- – R = Readable bit
- – W = Writable bit
- – U = Unimplemented bit, read as ‘0’
- – -n = Value at POR
- – ‘1’ = Bit is set
- – ‘0’ = Bit is cleared
- – x = Bit is unknown
- HS = Hardware Set
- HC = Hardware Cleared

35.10.1 Control A

Name: CTRLA
Offset: 0x00
Reset: 0x00000000
Property: PAC Write-Protection, Write-Synchronized, Enable-Protected

Bit	31	30	29	28	27	26	25	24
				CAPTMODE1[1:0]			CAPTMODE0[1:0]	
Access				R/W	R/W		R/W	R/W
Reset				0	0		0	0

Bit	23	22	21	20	19	18	17	16
			COPEN1	COPEN0			CAPTEN1	CAPTEN0
Access			R/W	R/W			R/W	R/W
Reset			0	0			0	0

Bit	15	14	13	12	11	10	9	8
	DMAOS				ALOCK	PRESCALER[2:0]		
Access	R/W				R/W	R/W	R/W	R/W
Reset	0				0	0	0	0

Bit	7	6	5	4	3	2	1	0
	ONDEMAND	RUNSTDBY	PRESCSYNC[1:0]		MODE[1:0]		ENABLE	SWRST
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	W
Reset	0	0	0	0	0	0	0	0

Bits 28:27 – CAPTMODE1[1:0] Capture mode Channel 1
These bits select the channel 1 capture mode.

Value	Name	Description
0x0	DEFAULT	Default capture
0x1	CAPTMIN	Minimum capture
0x2	CAPTMAX	Maximum capture
0x3		Reserved

Bits 25:24 – CAPTMODE0[1:0] Capture mode Channel 0
These bits select the channel 0 capture mode.

Value	Name	Description
0x0	DEFAULT	Default capture
0x1	CAPTMIN	Minimum capture
0x2	CAPTMAX	Maximum capture
0x3		Reserved

Bits 20, 21 – COPENx Capture On Pin x Enable [x=1..0]
Bit x of COPEN[1:0] selects the trigger source for capture operation, either events or I/O pin input.
This bit is not synchronized.

Value	Description
0	Event from Event System is selected as trigger source for capture operation on channel x.
1	I/O pin is selected as trigger source for capture operation on channel x.

Bits 16, 17 – CAPTENx Capture Channel x Enable [x=1..0]
Bit x of CAPTEN[1:0] selects whether channel x is a capture or a compare channel.
These bits are not synchronized.

Value	Description
0	CAPTEN disables capture on channel x.
1	CAPTEN enables capture on channel x.

Bit 15 – DMAOS DMA One-Shot Trigger Mode

This bit enables the DMA One-shot Trigger Mode.
Writing a '1' to this bit will generate a DMA trigger on TC cycle following a TC_CTRLBSET_CMD_DMAOS command.
Writing a '0' to this bit will generate DMA triggers on each TC cycle.
This bit is not synchronized.

Bit 11 – ALOCK Auto Lock

When this bit is set, Lock bit update (LUPD) is set to '1' on each overflow/underflow or re-trigger event.
This bit is not synchronized.

Value	Description
0	The LUPD bit is not affected on overflow/underflow, and re-trigger event.
1	The LUPD bit is set on each overflow/underflow or re-trigger event.

Bits 10:8 – PRESCALER[2:0] Prescaler

These bits select the counter prescaler factor.
These bits are not synchronized.

Value	Name	Description
0x0	DIV1	Prescaler: GCLK_TC
0x1	DIV2	Prescaler: GCLK_TC/2
0x2	DIV4	Prescaler: GCLK_TC/4
0x3	DIV8	Prescaler: GCLK_TC/8
0x4	DIV16	Prescaler: GCLK_TC/16
0x5	DIV64	Prescaler: GCLK_TC/64
0x6	DIV256	Prescaler: GCLK_TC/256
0x7	DIV1024	Prescaler: GCLK_TC/1024

Bit 7 – ONDEMAND Clock On Demand

This bit selects the clock requirements when the TC is stopped.
In standby mode, if the Run in Standby bit (CTRLA.RUNSTDBY) is '0', ONDEMAND is forced to '0'.
This bit is not synchronized.

Value	Description
0	The On Demand is disabled. If On Demand is disabled, the TC will continue to request the clock when its operation is stopped (STATUS.STOP=1).
1	The On Demand is enabled. When On Demand is enabled, the stopped TC will not request the clock. The clock is requested when a software re-trigger command is applied or when an event with start/re-trigger action is detected.

Bit 6 – RUNSTDBY Run in Standby

This bit is used to keep the TC running in standby mode.
This bit is not synchronized.

Value	Description
0	The TC is halted in standby.
1	The TC continues to run in standby.

Bits 5:4 – PRESCSYNC[1:0] Prescaler and Counter Synchronization

These bits select whether the counter must wrap around on the next GCLK_TCx clock or the next prescaled GCLK_TCx clock. It also makes it possible to reset the prescaler.
These bits are not synchronized.

Value	Name	Description
0x0	GCLK	Reload or reset the counter on next generic clock
0x1	PRESC	Reload or reset the counter on next prescaler clock
0x2	RESYNC	Reload or reset the counter on next generic clock. Reset the prescaler counter
0x3	-	Reserved

Bits 3:2 – MODE[1:0] Timer Counter Mode

These bits select the counter mode.
These bits are not synchronized.

Value	Name	Description
0x0	COUNT16	Counter in 16-bit mode
0x1	COUNT8	Counter in 8-bit mode
0x2	COUNT32	Counter in 32-bit mode
0x3	-	Reserved

Bit 1 – ENABLE Enable

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

This bit is not enable-protected.

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

Bit 0 – SWRST Software Reset

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

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

Writing a '1' to CTRLA.SWRST will always take precedence; all other writes in the same write-operation will be discarded.

This bit is not enable-protected.

35.10.2 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 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.

Bit 0 – DIR Counter Direction
This bit is used to change the direction of the counter.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the bit and make the counter count down.

Value	Description
0	The timer/counter is counting up (incrementing).
1	The timer/counter is counting down (decrementing).

35.10.3 Control B Set

Name: CTRLBSET
Offset: 0x05
Reset: 0x00
Property: PAC Write-Protection, Read-synchronized, Write-Synchronized

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

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 value different from 0x0 to these bits will issue a command for execution.

Value	Name	Description
0x0	NONE	No action
0x1	RETRIGGER	Force a start, restart or retrigger
0x2	STOP	Force a stop
0x3	UPDATE	Force update of double buffered registers
0x4	READSYNC	Force a read synchronization of COUNT

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 enable 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 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.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the LUPD bit.
This bit has no effect when input capture operation is enabled.

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.

Bit 0 – DIR Counter Direction
This bit is used to change the direction of the counter.
Writing a '0' to this bit has no effect
Writing a '1' to this bit will set the bit and make the counter count down.

Value	Description
0	The timer/counter is counting up (incrementing).
1	The timer/counter is counting down (decrementing).

35.10.4 Event Control

Name: EVCTRL
Offset: 0x06
Reset: 0x0000
Property: PAC Write-Protection, Enable-Protected

Bit	15	14	13	12	11	10	9	8
			MCEO1	MCEO0				OVFEO
Access			R/W	R/W				R/W
Reset			0	0				0

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

Bits 12, 13 – MCEOx Match or Capture Channel x Event Output Enable [x = 1..0]
These bits enable the generation of an event for every match or capture on channel x.

Value	Description
0	Match/Capture event on channel x is disabled and will not be generated.
1	Match/Capture event on channel x is enabled and will be generated for every compare/capture.

Bit 8 – OVFEO Overflow/Underflow Event Output Enable
This bit enables the Overflow/Underflow event. When enabled, an event will be generated when the counter overflows/underflows.

Value	Description
0	Overflow/Underflow event is disabled and will not be generated.
1	Overflow/Underflow event is enabled and will be generated for every counter overflow/underflow.

Bit 5 – TCEI TC Event Enable
This bit is used to enable asynchronous input events to the TC.

Value	Description
0	Incoming events are disabled.
1	Incoming events are enabled.

Bit 4 – TCINV TC Inverted Event Input Polarity
This bit inverts the asynchronous input event source.

Value	Description
0	Input event source is not inverted.
1	Input event source is inverted.

Bits 2:0 – EVACT[2:0] Event Action
These bits define the event action the TC will perform on an event.

Value	Name	Description
0x0	OFF	Event action disabled
0x1	RETRIGGER	Start, restart or retrigger TC on event
0x2	COUNT	Count on event
0x3	START	Start TC on event
0x4	STAMP	Time stamp capture
0x5	PPW	Period captured in CC0, pulse width in CC1
0x6	PWP	Period captured in CC1, pulse width in CC0
0x7	PW	Pulse width capture

35.10.5 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

Bits 4, 5 – MCx Match or Capture Channel x Interrupt Enable [x = 1..0]
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 these bits 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 these bits 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.

35.10.6 Interrupt Enable Set

Name: INTENSET
Offset: 0x09
Reset: 0x00
Property: PAC Write-Protection

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

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

Bits 4, 5 – MCx Match or Capture Channel x Interrupt Enable [x = 1..0]
Writing a '0' to these bits has no effect.
Writing a '1' to MCx will set the corresponding Match or Capture Channel x Interrupt Enable bit, which enables 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 Enable
Writing a '0' to these bits has no effect.
Writing a '1' to this bit will set the Error Interrupt Enable bit, which enables the Error interrupt.

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

Bit 0 – OVF Overflow Interrupt Enable
Writing a '0' to these bits has no effect.
Writing a '1' to this bit will set the Overflow Interrupt Enable bit, which enables the Overflow interrupt request.

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

35.10.7 Interrupt Flag Status and Clear

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

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

Bits 4, 5 – MCx Match or Capture Channel x [x = 1..0]
This flag is set on a comparison match, or when the corresponding CCx register contains a valid capture value. This flag is set on the next CLK_TC_CNT cycle, and will generate an interrupt request if the corresponding Match or Capture Channel x Interrupt Enable bit in the Interrupt Enable Set register (INTENSET.MCx) is '1'.
Writing a '0' to these bits has no effect.
Writing a '1' to one of these bits will clear the corresponding Match or Capture Channel x interrupt flag
In capture operation, this flag is automatically cleared when CCx register is read.

Bit 1 – ERR Error Interrupt Flag
This flag is set when a new capture occurs on a channel while the corresponding Match or Capture Channel x interrupt flag is set, in which case there is no place to store the new capture.
Writing a '0' to these bits has no effect.
Writing a '1' to this bit clears the Error interrupt flag.

Bit 0 – OVF Overflow Interrupt Flag
This flag is set on the next CLK_TC_CNT cycle after an overflow condition occurs, and will generate an interrupt request if INTENCLR.OVF or INTENSET.OVF is '1'.
Writing a '0' to these bits has no effect.
Writing a '1' to this bit clears the Overflow interrupt flag.

35.10.8 Status

Name: STATUS
Offset: 0x0B
Reset: 0x01
Property: Read-Synchronized

Bit	7	6	5	4	3	2	1	0
			CCBUFV1	CCBUFV0	PERBUFV		SLAVE	STOP
Access			R/W	R/W	R/W		R	R
Reset			0	0	0		0	1

Bits 4, 5 – CCBUFVx Channel x Compare or Capture Buffer Valid [x = 1..0]
For a compare channel x, the bit x is set when a new value is written to the corresponding CCBUFx register. The bit x is cleared by writing a '1' to it when CTRLB.LUPD is set, or it is cleared automatically by hardware on UPDATE condition.
For a capture channel x, the bit x is set when a valid capture value is stored in the CCBUFx register. The bit x is cleared automatically when the CCx register is read.

Bit 3 – PERBUFV Period Buffer Valid
This bit is set when a new value is written to the PERBUF register. The bit is cleared by writing '1' to the corresponding location when CTRLB.LUPD is set, or automatically cleared by hardware on UPDATE condition. This bit is available only in 8-bit mode and will always read zero in 16- and 32-bit modes.

Bit 1 – SLAVE Client Status Flag
This bit is only available in 32-bit mode on the Client TC (i.e., TC1, TC3, TC5 and/or TC7). The bit is set when the associated Host TC (TC0, TC2, TC4 and/or TC6, respectively) is set to run in 32-bit mode.

Bit 0 – STOP Stop Status Flag
This bit is set when the TC is disabled, on a Stop command, or on an overflow/underflow condition when the One-Shot bit in the Control B Set register (CTRLBSET.ONESHOT) is '1'.

Value	Description
0	Counter is running.
1	Counter is stopped.

35.10.9 Waveform Generation Control

Name: WAVE
Offset: 0x0C
Reset: 0x00
Property: PAC Write-Protection, Enable-Protected

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

Bits 1:0 – WAVEGEN[1:0] Waveform Generation Mode
These bits select the waveform generation operation. They affect the top value, as shown in Waveform Output Operations. They also control whether frequency or PWM waveform generation must be used. The waveform generation operations are explained in Waveform Output Operations. See *Waveform Output Operations* from Related Links.
These bits are not synchronized.

Value	Name	Operation	Top Value	Output Waveform on Match	Output Waveform on Wraparound
0x0	NFRQ	Normal frequency	PER[1] / Max	Toggle	No action
0x1	MFRQ	Match frequency	CC0	Toggle	No action
0x2	NPWM	Normal PWM	PER[1] / Max	Set	Clear
0x3	MPWM	Match PWM	CC0	Set	Clear

1. This depends on the TC mode: In 8-bit mode, the top value is the Period Value register (PER). In 16- and 32-bit mode, it is the respective MAX value.

Related Links
35.6.2.6.1. Waveform Output Operations

35.10.10 Driver Control

Name: DRVCTRL
Offset: 0x0D
Reset: 0x00
Property: PAC Write-Protection, Enable-Protected

Bit	7	6	5	4	3	2	1	0
							INVEN1	INVEN0
Access							R/W	R/W
Reset							0	0

Bits 0, 1 – INVENx Output Waveform x Invert Enable [x=1..0]
Bit x of INVEN[1:0] selects inversion of the output or capture trigger input of channel x.

Value	Description
0	Disable inversion of the WO[x] output and IO input pin.
1	Enable inversion of the WO[x] output and IO input pin.

35.10.11 Debug Control

Name: DBGCTRL
Offset: 0x0F
Reset: 0x00
Property: PAC Write-Protection

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

Bit 0 – DBGRUN Run in Debug Mode

This bit is not affected by a software Reset, and must not be changed by software while the TC is enabled.

Value	Description
0	The TC is halted when the device is halted in debug mode.
1	The TC continues normal operation when the device is halted in debug mode.

35.10.12 Synchronization Busy

Name: SYNCBUSY
Offset: 0x10
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	CC1	CC0	PER	COUNT	STATUS	CTRLB	ENABLE	SWRST
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 6, 7 – CCx Compare/Capture Channel x Synchronization Busy [x=0..1]
For details on CC channels number, refer to each TC feature list.
This bit is set when the synchronization of CCx between clock domains is started.
This bit is also set when the CCBUFx is written, and cleared on update condition. The bit is automatically cleared when the STATUS.CCBUFx bit is cleared.

Bit 5 – PER PER Synchronization Busy
This bit is cleared when the synchronization of PER between the clock domains is complete.
This bit is set when the synchronization of PER between clock domains is started.
This bit is also set when the PER is written, and cleared on update condition. The bit is automatically cleared when the STATUS.PERBUF bit is cleared.

Bit 4 – COUNT COUNT Synchronization Busy
This bit is cleared when the synchronization of COUNT between the clock domains is complete.
This bit is set when the synchronization of COUNT between clock domains is started.

Bit 3 – STATUS STATUS Synchronization Busy
This bit is cleared when the synchronization of STATUS between the clock domains is complete.
This bit is set when a '1' is written to the Capture Channel Buffer Valid status flags (STATUS.CCBUFVx) and the synchronization of STATUS between clock domains is started.

Bit 2 – CTRLB CTRLB Synchronization Busy
This bit is cleared when the synchronization of CTRLB between the clock domains is complete.
This bit is set when the synchronization of CTRLB between clock domains is started.

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

Bit 0 – SWRST SWRST Synchronization Busy
This bit is cleared when the synchronization of SWRST bit between the clock domains is complete.
This bit is set when the synchronization of SWRST bit between clock domains is started.
Note: During a SWRST, access to registers/bits without SWRST are disallowed until SYNCBUSY.SWRST cleared by hardware.

35.10.13 Counter Value, 16-bit Mode

Name: COUNT
Offset: 0x14
Reset: 0x00
Property: PAC Write-Protection, Write-Synchronized, Read-Synchronized

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

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 15:0 – COUNT[15:0] Counter Value
These bits contain the current counter value.

35.10.14 Period Value, 16-bit Mode

Name: PER
Offset: 0x1A
Reset: 0xFFFF
Property: Write-Synchronized

Bit	15	14	13	12	11	10	9	8
	PER[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
	PER[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	1

Bits 15:0 – PER[15:0] Period Value
These bits hold the value of the TC period count.

35.10.15 Channel x Compare/Capture Value, 16-bit Mode

Name: CCx
Offset: 0x1C + x*0x02 [x=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 (WAVE.WAVEGEN), the CC0 register is used as a period register.

35.10.16 Period Buffer Value, 16-bit Mode

Name: PERBUF
Offset: 0x2E
Reset: 0xFFFF
Property: Write-Synchronized

Bit	15	14	13	12	11	10	9	8
	PERBUF[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
	PERBUF[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	1

Bits 15:0 – PERBUF[15:0] Period Buffer Value
These bits hold the value of the period buffer register. The value is copied to PER register on UPDATE condition.

35.10.17 Channel x Compare Buffer Value, 16-bit Mode

Name: CCBUFx
Offset: 0x30 + x*0x02 [x=0..1]
Reset: 0x0000
Property: Write-Synchronized

Bit	15	14	13	12	11	10	9	8
	CCBUF[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
	CCBUF[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 – CCBUF[15:0] Channel x Compare Buffer Value
These bits hold the value of the Channel x Compare Buffer Value. When the buffer valid flag is '1' and double buffering is enabled (CTRLBCLR.LUPD=1), the data from buffer registers will be copied into the corresponding CCx register under UPDATE condition (CTRLBSET.CMD=0x3), including the software update command.

35.11 Register Summary

See *TCx (x = 0 to 7)* module in the *Product Memory Mapping Overview* from Related Links for base address based on the TC instant used.

Note: All registers in this table have corresponding CLR, SET and INV registers at its virtual address, plus an offset of 0x4, 0x8 and 0xC, respectively. See *CLR, SET, and INV Registers* from Related Links.

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x00	CTRLA	7:0	ONDEMAND	RUNSTDBY	PRESCSYNC[1:0]		MODE[1:0]		ENABLE	SWRST
		15:8	DMAOS				ALOCK	PRESCALER[2:0]		
		23:16			COPEN1	COPEN0			CAPTEN1	CAPTEN0
		31:24				CAPTMODE1[1:0]			CAPTMODE0[1:0]	
0x04	CTRLBCLR	7:0	CMD[2:0]					ONESHOT	LUPD	DIR
0x05	CTRLBSET	7:0	CMD[2:0]					ONESHOT	LUPD	DIR
0x06	EVCTRL	7:0			TCEI	TCINV			EVACT[2:0]	
		15:8			MCEO1	MCEO0				OVFEO
0x08	INTENCLR	7:0			MC1	MC0			ERR	OVF
0x09	INTENSET	7:0			MC1	MC0			ERR	OVF
0x0A	INTFLAG	7:0			MC1	MC0			ERR	OVF
0x0B	STATUS	7:0			CCBUFV1	CCBUFV0	PERBUFV		SLAVE	STOP
0x0C	WAVE	7:0							WAVEGEN[1:0]	
0x0D	DRVCTRL	7:0							INVEN1	INVEN0
0x0E	Reserved									
0x0F	DBGCTRL	7:0								DBGRUN
0x10	SYNCBUSY	7:0	CC1	CC0	PER	COUNT	STATUS	CTRLB	ENABLE	SWRST
0x11 ... 0x13	Reserved									
0x14	COUNT	7:0	COUNT[7:0]							
		15:8	COUNT[15:8]							
		23:16	COUNT[23:16]							
		31:24	COUNT[31:24]							
0x18 ... 0x1B	Reserved									
0x1C	CC0	7:0	CC[7:0]							
		15:8	CC[15:8]							
		23:16	CC[23:16]							
		31:24	CC[31:24]							
0x20	CC1	7:0	CC[7:0]							
		15:8	CC[15:8]							
		23:16	CC[23:16]							
		31:24	CC[31:24]							
0x24 ... 0x2F	Reserved									
0x30	CCBUF0	7:0	CCBUF[7:0]							
		15:8	CCBUF[15:8]							
		23:16	CCBUF[23:16]							
		31:24	CCBUF[31:24]							
0x34	CCBUF1	7:0	CCBUF[7:0]							
		15:8	CCBUF[15:8]							
		23:16	CCBUF[23:16]							
		31:24	CCBUF[31:24]							

Related Links

5.4.1.9. CLR, SET and INV Registers

7. Product Memory Mapping Overview

35.12 Register Description - 32-bit Mode

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.

Some registers are synchronized when read and/or written. Synchronization is denoted by the “Write-Synchronized” or the “Read-Synchronized” property in each individual register description.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable protection is denoted by the “Enable-Protected” property in each individual register description.

Following conventions are used in the register description:

- – R = Readable bit
- – W = Writable bit
- – U = Unimplemented bit, read as ‘0’
- – -n = Value at POR
- – ‘1’ = Bit is set
- – ‘0’ = Bit is cleared
- – x = Bit is unknown
- HS = Hardware Set
- HC = Hardware Cleared

35.12.1 Control A

Name: CTRLA
Offset: 0x00
Reset: 0x00000000
Property: PAC Write-Protection, Write-Synchronized, Enable-Protected

Bit	31	30	29	28	27	26	25	24
				CAPTMODE1[1:0]			CAPTMODE0[1:0]	
Access				R/W	R/W		R/W	R/W
Reset				0	0		0	0

Bit	23	22	21	20	19	18	17	16
			COPEN1	COPEN0			CAPTEN1	CAPTEN0
Access			R/W	R/W			R/W	R/W
Reset			0	0			0	0

Bit	15	14	13	12	11	10	9	8
	DMAOS				ALOCK	PRESCALER[2:0]		
Access	R/W				R/W	R/W	R/W	R/W
Reset	0				0	0	0	0

Bit	7	6	5	4	3	2	1	0
	ONDEMAND	RUNSTDBY	PRESCSYNC[1:0]		MODE[1:0]		ENABLE	SWRST
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	W
Reset	0	0	0	0	0	0	0	0

Bits 28:27 – CAPTMODE1[1:0] Capture mode Channel 1
These bits select the channel 1 capture mode.

Value	Name	Description
0x0	DEFAULT	Default capture
0x1	CAPTMIN	Minimum capture
0x2	CAPTMAX	Maximum capture
0x3		Reserved

Bits 25:24 – CAPTMODE0[1:0] Capture mode Channel 0
These bits select the channel 0 capture mode.

Value	Name	Description
0x0	DEFAULT	Default capture
0x1	CAPTMIN	Minimum capture
0x2	CAPTMAX	Maximum capture
0x3		Reserved

Bits 20, 21 – COPENx Capture On Pin x Enable [x=1..0]
Bit x of COPEN[1:0] selects the trigger source for capture operation, either events or I/O pin input.
This bit is not synchronized.

Value	Description
0	Event from Event System is selected as trigger source for capture operation on channel x.
1	I/O pin is selected as trigger source for capture operation on channel x.

Bits 16, 17 – CAPTENx Capture Channel x Enable [x=1..0]
Bit x of CAPTEN[1:0] selects whether channel x is a capture or a compare channel.
These bits are not synchronized.

Value	Description
0	CAPTEN disables capture on channel x.
1	CAPTEN enables capture on channel x.

Bit 15 – DMAOS DMA One-Shot Trigger Mode

This bit enables the DMA One-shot Trigger Mode.
Writing a '1' to this bit will generate a DMA trigger on TC cycle following a TC_CTRLBSET_CMD_DMAOS command.
Writing a '0' to this bit will generate DMA triggers on each TC cycle.
This bit is not synchronized.

Bit 11 – ALOCK Auto Lock

When this bit is set, Lock bit update (LUPD) is set to '1' on each overflow/underflow or re-trigger event.
This bit is not synchronized.

Value	Description
0	The LUPD bit is not affected on overflow/underflow, and re-trigger event.
1	The LUPD bit is set on each overflow/underflow or re-trigger event.

Bits 10:8 – PRESCALER[2:0] Prescaler

These bits select the counter prescaler factor.
These bits are not synchronized.

Value	Name	Description
0x0	DIV1	Prescaler: GCLK_TC
0x1	DIV2	Prescaler: GCLK_TC/2
0x2	DIV4	Prescaler: GCLK_TC/4
0x3	DIV8	Prescaler: GCLK_TC/8
0x4	DIV16	Prescaler: GCLK_TC/16
0x5	DIV64	Prescaler: GCLK_TC/64
0x6	DIV256	Prescaler: GCLK_TC/256
0x7	DIV1024	Prescaler: GCLK_TC/1024

Bit 7 – ONDEMAND Clock On Demand

This bit selects the clock requirements when the TC is stopped.
In standby mode, if the Run in Standby bit (CTRLA.RUNSTDBY) is '0', ONDEMAND is forced to '0'.
This bit is not synchronized.

Value	Description
0	The On Demand is disabled. If On Demand is disabled, the TC will continue to request the clock when its operation is stopped (STATUS.STOP=1).
1	The On Demand is enabled. When On Demand is enabled, the stopped TC will not request the clock. The clock is requested when a software re-trigger command is applied or when an event with start/re-trigger action is detected.

Bit 6 – RUNSTDBY Run in Standby

This bit is used to keep the TC running in standby mode.
This bit is not synchronized.

Value	Description
0	The TC is halted in standby.
1	The TC continues to run in standby.

Bits 5:4 – PRESCSYNC[1:0] Prescaler and Counter Synchronization

These bits select whether the counter must wrap around on the next GCLK_TCx clock or the next prescaled GCLK_TCx clock. It also makes it possible to reset the prescaler.
These bits are not synchronized.

Value	Name	Description
0x0	GCLK	Reload or reset the counter on next generic clock
0x1	PRESC	Reload or reset the counter on next prescaler clock
0x2	RESYNC	Reload or reset the counter on next generic clock. Reset the prescaler counter
0x3	-	Reserved

Bits 3:2 – MODE[1:0] Timer Counter Mode

These bits select the counter mode.
These bits are not synchronized.

Value	Name	Description
0x0	COUNT16	Counter in 16-bit mode
0x1	COUNT8	Counter in 8-bit mode
0x2	COUNT32	Counter in 32-bit mode
0x3	-	Reserved

Bit 1 – ENABLE Enable

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

This bit is not enable-protected.

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

Bit 0 – SWRST Software Reset

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

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

Writing a '1' to CTRLA.SWRST will always take precedence; all other writes in the same write-operation will be discarded.

This bit is not enable-protected.

35.12.2 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 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.

Bit 0 – DIR Counter Direction
This bit is used to change the direction of the counter.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the bit and make the counter count down.

Value	Description
0	The timer/counter is counting up (incrementing).
1	The timer/counter is counting down (decrementing).

35.12.3 Control B Set

Name: CTRLBSET
Offset: 0x05
Reset: 0x00
Property: PAC Write-Protection, Read-synchronized, Write-Synchronized

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

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 value different from 0x0 to these bits will issue a command for execution.

Value	Name	Description
0x0	NONE	No action
0x1	RETRIGGER	Force a start, restart or retrigger
0x2	STOP	Force a stop
0x3	UPDATE	Force update of double buffered registers
0x4	READSYNC	Force a read synchronization of COUNT

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 enable 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 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.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the LUPD bit.
This bit has no effect when input capture operation is enabled.

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.

Bit 0 – DIR Counter Direction
This bit is used to change the direction of the counter.
Writing a '0' to this bit has no effect
Writing a '1' to this bit will set the bit and make the counter count down.

Value	Description
0	The timer/counter is counting up (incrementing).
1	The timer/counter is counting down (decrementing).

35.12.4 Event Control

Name: EVCTRL
Offset: 0x06
Reset: 0x0000
Property: PAC Write-Protection, Enable-Protected

Bit	15	14	13	12	11	10	9	8
			MCEO1	MCEO0				OVFEO
Access			R/W	R/W				R/W
Reset			0	0				0

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

Bits 12, 13 – MCEOx Match or Capture Channel x Event Output Enable [x = 1..0]
These bits enable the generation of an event for every match or capture on channel x.

Value	Description
0	Match/Capture event on channel x is disabled and will not be generated.
1	Match/Capture event on channel x is enabled and will be generated for every compare/capture.

Bit 8 – OVFEO Overflow/Underflow Event Output Enable
This bit enables the Overflow/Underflow event. When enabled, an event will be generated when the counter overflows/underflows.

Value	Description
0	Overflow/Underflow event is disabled and will not be generated.
1	Overflow/Underflow event is enabled and will be generated for every counter overflow/underflow.

Bit 5 – TCEI TC Event Enable
This bit is used to enable asynchronous input events to the TC.

Value	Description
0	Incoming events are disabled.
1	Incoming events are enabled.

Bit 4 – TCINV TC Inverted Event Input Polarity
This bit inverts the asynchronous input event source.

Value	Description
0	Input event source is not inverted.
1	Input event source is inverted.

Bits 2:0 – EVACT[2:0] Event Action
These bits define the event action the TC will perform on an event.

Value	Name	Description
0x0	OFF	Event action disabled
0x1	RETRIGGER	Start, restart or retrigger TC on event
0x2	COUNT	Count on event
0x3	START	Start TC on event
0x4	STAMP	Time stamp capture
0x5	PPW	Period captured in CC0, pulse width in CC1
0x6	PWP	Period captured in CC1, pulse width in CC0
0x7	PW	Pulse width capture

35.12.5 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

Bits 4, 5 – MCx Match or Capture Channel x Interrupt Enable [x = 1..0]
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 these bits 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 these bits 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.

35.12.6 Interrupt Enable Set

Name: INTENSET
Offset: 0x09
Reset: 0x00
Property: PAC Write-Protection

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

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

Bits 4, 5 – MCx Match or Capture Channel x Interrupt Enable [x = 1..0]
Writing a '0' to these bits has no effect.
Writing a '1' to MCx will set the corresponding Match or Capture Channel x Interrupt Enable bit, which enables 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 Enable
Writing a '0' to these bits has no effect.
Writing a '1' to this bit will set the Error Interrupt Enable bit, which enables the Error interrupt.

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

Bit 0 – OVF Overflow Interrupt Enable
Writing a '0' to these bits has no effect.
Writing a '1' to this bit will set the Overflow Interrupt Enable bit, which enables the Overflow interrupt request.

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

35.12.7 Interrupt Flag Status and Clear

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

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

Bits 4, 5 – MCx Match or Capture Channel x [x = 1..0]
This flag is set on a comparison match, or when the corresponding CCx register contains a valid capture value. This flag is set on the next CLK_TC_CNT cycle, and will generate an interrupt request if the corresponding Match or Capture Channel x Interrupt Enable bit in the Interrupt Enable Set register (INTENSET.MCx) is '1'.
Writing a '0' to these bits has no effect.
Writing a '1' to one of these bits will clear the corresponding Match or Capture Channel x interrupt flag
In capture operation, this flag is automatically cleared when CCx register is read.

Bit 1 – ERR Error Interrupt Flag
This flag is set when a new capture occurs on a channel while the corresponding Match or Capture Channel x interrupt flag is set, in which case there is no place to store the new capture.
Writing a '0' to these bits has no effect.
Writing a '1' to this bit clears the Error interrupt flag.

Bit 0 – OVF Overflow Interrupt Flag
This flag is set on the next CLK_TC_CNT cycle after an overflow condition occurs, and will generate an interrupt request if INTENCLR.OVF or INTENSET.OVF is '1'.
Writing a '0' to these bits has no effect.
Writing a '1' to this bit clears the Overflow interrupt flag.

35.12.8 Status

Name: STATUS
Offset: 0x0B
Reset: 0x01
Property: Read-Synchronized

Bit	7	6	5	4	3	2	1	0
			CCBUFV1	CCBUFV0	PERBUFV		SLAVE	STOP
Access			R/W	R/W	R/W		R	R
Reset			0	0	0		0	1

Bits 4, 5 – CCBUFVx Channel x Compare or Capture Buffer Valid [x = 1..0]
For a compare channel x, the bit x is set when a new value is written to the corresponding CCBUFx register. The bit x is cleared by writing a '1' to it when CTRLB.LUPD is set, or it is cleared automatically by hardware on UPDATE condition.
For a capture channel x, the bit x is set when a valid capture value is stored in the CCBUFx register. The bit x is cleared automatically when the CCx register is read.

Bit 3 – PERBUFV Period Buffer Valid
This bit is set when a new value is written to the PERBUF register. The bit is cleared by writing '1' to the corresponding location when CTRLB.LUPD is set, or automatically cleared by hardware on UPDATE condition. This bit is available only in 8-bit mode and will always read zero in 16- and 32-bit modes.

Bit 1 – SLAVE Client Status Flag
This bit is only available in 32-bit mode on the Client TC (i.e., TC1, TC3, TC5 and/or TC7). The bit is set when the associated Host TC (TC0, TC2, TC4 and/or TC6, respectively) is set to run in 32-bit mode.

Bit 0 – STOP Stop Status Flag
This bit is set when the TC is disabled, on a Stop command, or on an overflow/underflow condition when the One-Shot bit in the Control B Set register (CTRLBSET.ONESHOT) is '1'.

Value	Description
0	Counter is running.
1	Counter is stopped.

35.12.9 Waveform Generation Control

Name: WAVE
Offset: 0x0C
Reset: 0x00
Property: PAC Write-Protection, Enable-Protected

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

Bits 1:0 – WAVEGEN[1:0] Waveform Generation Mode
These bits select the waveform generation operation. They affect the top value, as shown in Waveform Output Operations. They also control whether frequency or PWM waveform generation must be used. The waveform generation operations are explained in Waveform Output Operations. See *Waveform Output Operations* from Related Links.
These bits are not synchronized.

Value	Name	Operation	Top Value	Output Waveform on Match	Output Waveform on Wraparound
0x0	NFRQ	Normal frequency	PER[1] / Max	Toggle	No action
0x1	MFRQ	Match frequency	CC0	Toggle	No action
0x2	NPWM	Normal PWM	PER[1] / Max	Set	Clear
0x3	MPWM	Match PWM	CC0	Set	Clear

1. This depends on the TC mode: In 8-bit mode, the top value is the Period Value register (PER). In 16- and 32-bit mode, it is the respective MAX value.

Related Links
35.6.2.6.1. Waveform Output Operations

35.12.10 Driver Control

Name: DRVCTRL
Offset: 0x0D
Reset: 0x00
Property: PAC Write-Protection, Enable-Protected

Bit	7	6	5	4	3	2	1	0
							INVEN1	INVEN0
Access							R/W	R/W
Reset							0	0

Bits 0, 1 – INVENx Output Waveform x Invert Enable [x=1..0]
Bit x of INVEN[1:0] selects inversion of the output or capture trigger input of channel x.

Value	Description
0	Disable inversion of the WO[x] output and IO input pin.
1	Enable inversion of the WO[x] output and IO input pin.

35.12.11 Debug Control

Name: DBGCTRL
Offset: 0x0F
Reset: 0x00
Property: PAC Write-Protection

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

Bit 0 – DBGRUN Run in Debug Mode

This bit is not affected by a software Reset, and must not be changed by software while the TC is enabled.

Value	Description
0	The TC is halted when the device is halted in debug mode.
1	The TC continues normal operation when the device is halted in debug mode.

35.12.12 Synchronization Busy

Name: SYNCBUSY
Offset: 0x10
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	CC1	CC0	PER	COUNT	STATUS	CTRLB	ENABLE	SWRST
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 6, 7 – CCx Compare/Capture Channel x Synchronization Busy [x=0..1]
For details on CC channels number, refer to each TC feature list.
This bit is set when the synchronization of CCx between clock domains is started.
This bit is also set when the CCBUFx is written, and cleared on update condition. The bit is automatically cleared when the STATUS.CCBUFx bit is cleared.

Bit 5 – PER PER Synchronization Busy
This bit is cleared when the synchronization of PER between the clock domains is complete.
This bit is set when the synchronization of PER between clock domains is started.
This bit is also set when the PER is written, and cleared on update condition. The bit is automatically cleared when the STATUS.PERBUF bit is cleared.

Bit 4 – COUNT COUNT Synchronization Busy
This bit is cleared when the synchronization of COUNT between the clock domains is complete.
This bit is set when the synchronization of COUNT between clock domains is started.

Bit 3 – STATUS STATUS Synchronization Busy
This bit is cleared when the synchronization of STATUS between the clock domains is complete.
This bit is set when a '1' is written to the Capture Channel Buffer Valid status flags (STATUS.CCBUFVx) and the synchronization of STATUS between clock domains is started.

Bit 2 – CTRLB CTRLB Synchronization Busy
This bit is cleared when the synchronization of CTRLB between the clock domains is complete.
This bit is set when the synchronization of CTRLB between clock domains is started.

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

Bit 0 – SWRST SWRST Synchronization Busy
This bit is cleared when the synchronization of SWRST bit between the clock domains is complete.
This bit is set when the synchronization of SWRST bit between clock domains is started.
Note: During a SWRST, access to registers/bits without SWRST are disallowed until SYNCBUSY.SWRST cleared by hardware.

35.12.13 Counter Value, 32-bit Mode

Name: COUNT
Offset: 0x14
Reset: 0x00
Property: PAC Write-Protection, Write-Synchronized, Read-Synchronized

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

Bit	31	30	29	28	27	26	25	24
	COUNT[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
	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 31:0 – COUNT[31:0] Counter Value
These bits contain the current counter value.

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

Name: CCx
Offset: 0x1C + x*0x04 [x=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 (WAVE.WAVEGEN), the CC0 register is used as a period register.

35.12.15 Channel x Compare Buffer Value, 32-bit Mode

Name: CCBUFx
Offset: 0x30 + x*0x04 [x=0..1]
Reset: 0x00000000
Property: Write-Synchronized

Bit	31	30	29	28	27	26	25	24
	CCBUF[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
	CCBUF[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
	CCBUF[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
	CCBUF[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 – CCBUF[31:0] Channel x Compare Buffer Value
These bits hold the value of the Channel x Compare Buffer Value. When the buffer valid flag is '1' and double buffering is enabled (CTRLBCLR.LUPD=1), the data from buffer registers will be copied into the corresponding CCx register under UPDATE condition (CTRLBSET.CMD=0x3), including the software update command.

36. Timer/Counter for Control Applications (TCC)

36.1 Overview

The device provides three instances of the Timer/Counter for Control Applications (TCC).

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 featured for motor control, ballast, LED, H-bridge, power converters and other types of power control applications. They allow for low-side 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.

Note: The TCC configurations, such as channel numbers and features, may be reduced for some of the TCC instances.

Table 36-1. TCC Specific Configuration

TCC No.	Counter Size (SIZE)	Link Role (Host_Client_MODE) (0 = NA, 1 = Host, 2 = Client)	Channels (CC_NUM)	Pins WO_NUM (OW_NUM)
0	24	1	6	6
1	24	2	6	6
2	16	0	2	2

Note: Traditional Timer/Counter for Control Applications (TCC) documentation uses the terminology “Master” and “Slave”. The equivalent Microchip terminology used in this document is “Host” and “Client”, respectively.

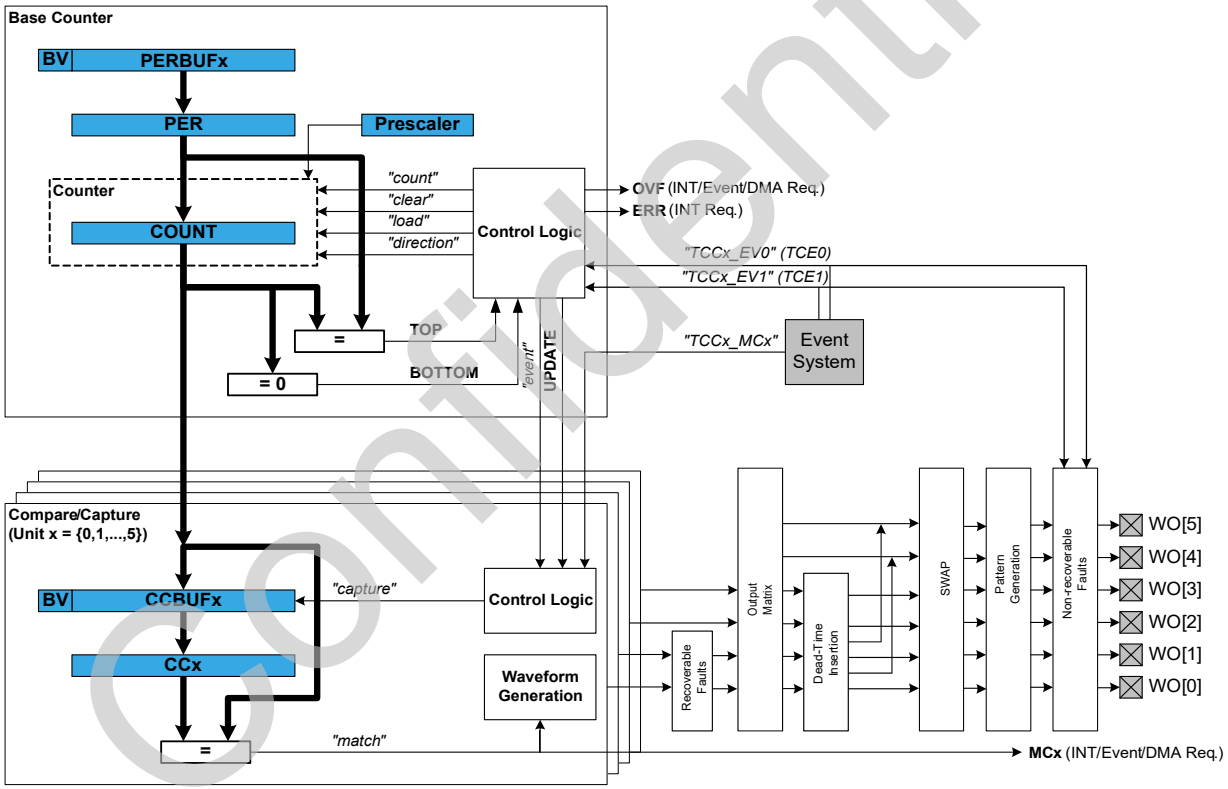
36.2 Features

- Up to six 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 PWM 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-side 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 a source of non-recoverable fault
- Input Events:
  - Two input events (EVx) for counter
  - One input event (MCx) for each channel
- Output Events:
  - Three output events (Count, re-trigger and overflow) are available for counter
  - One compare match/input capture event output for each channel
- Interrupts:
  - Overflow and re-trigger interrupt
  - Compare match/input capture interrupt
  - Interrupt on fault detection

36.3 Block Diagram

Figure 36-1. Timer/Counter for Control Applications - Block Diagram



36.4 Signal Description

Table 36-2. Signal Description

Pin Name	Type	Description
TCCx/WO[0]	Digital output	Compare channel 0 waveform output
TCCx/WO[1]	Digital output	Compare channel 1 waveform output
…	...	...

.....continued		
Pin Name	**Type**	**Description**
TCCx/WO[WO_NUM-1]	Digital output	Compare channel n waveform output

See *I/O Ports and Peripheral Pin Select (PPS)* from Related Links for details on the pin mapping for this peripheral. One signal can be mapped on several pins.

Related Links

5. I/O Ports and Peripheral Pin Select (PPS)

36.5 Product Dependencies

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

36.5.1 I/O Lines

In order to use the I/O lines of this peripheral, the I/O pins must be configured using the I/O Peripheral Pin Select (PPS).

36.5.2 Power Management

This peripheral can continue to operate in any Sleep mode (Idle, Standby sleep) where its source clock is running. The interrupts can wake-up the device from Sleep modes. Events connected to the event system can trigger other operations in the system without exiting Sleep modes.

36.5.3 Clocks

A generic clock (GCLK_TCCx) is required to clock the TCC. This clock must be configured and enabled in the generic clock controller before using the TCC. Note that TCC1 and TCC2 share a single peripheral clock generator.

The generic clocks (GCLK_TCCx) are asynchronous to the bus clock (PB1_CLK). Due to this asynchronicity, writing certain registers will require synchronization between the clock domains.

36.5.4 DMA

The DMA request lines are connected to the DMA Controller (DMAC). In order to use DMA requests with this peripheral, the DMAC must be configured first (see *Direct Memory Access Controller (DMAC)* from Related Links).

Related Links

22. Direct Memory Access Controller (DMAC)

36.5.5 Interrupts

The interrupt request line is connected to the Interrupt Controller. In order to use interrupt requests of this peripheral, the Interrupt Controller (NVIC) must be configured first. See *Nested Vector Interrupt Controller (NVIC)* from Related Links.

Related Links

8.2. Nested Vector Interrupt Controller (NVIC)

36.5.6 Events

The events of this peripheral are connected to the Event System.

Related Links

26. Event System (EVSYS)

36.5.7 Debug Operation

When the CPU is halted in Debug mode, this peripheral will halt normal operation. This peripheral can be forced to continue operation during debugging - refer to the Debug Control (DBGCTRL) register for details.

Refer to 36.8.8. DBGCTRL register for details.

36.5.8 Register Access Protection

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

- Interrupt Flag register (INTFLAG)
- Status register (STATUS)
- Period and Period Buffer registers (PER, PERBUF)
- Compare/Capture and Compare/Capture Buffer registers (CCx, CCBUFx)
- Control Waveform register (WAVE)
- Pattern Generation Value and Pattern Generation Value Buffer registers (PATT, PATTBUF)

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.

36.5.9 Analog Connections

Not applicable.

36.6 Functional Description

36.6.1 Principle of Operation

The following definitions are used throughout the documentation:

Table 36-3. Timer/Counter for Control Applications – Definitions

Name	Description
TOP	The counter reaches TOP when it becomes equal to the highest value in the count sequence. The TOP value can be the same as Period (PER) or the Compare Channel 0 (CC0) register value depending on the Waveform Generator mode in Waveform Output Operations. See *Waveform Output Generation Operations* from Related Links.
ZERO	The counter reaches ZERO when it contains all zeros.
MAX	The counter reaches maximum when it contains all ones.
UPDATE	The timer/counter signals an update when it reaches ZERO or TOP, depending on the direction settings.
Timer	The timer/counter clock control is handled by an internal source.
Counter	The clock control is handled externally (e.g., counting external events).
CC	For compare operations, the CC are referred to as "compare channels." For capture operations, the CC are referred to as "capture channels."

Each TCC instance has up to six compare/capture channels (CCx).

The Counter register (COUNT), Period registers with Buffer (PER and PERBUF), and Compare and Capture registers with buffers (CCx and CCBUFx) are 16- or 24-bit registers, depending on each TCC instance. Each Buffer register has a Buffer Valid (BUFV) flag that indicates when the buffer contains a new value.

Under normal operation, the counter value is continuously compared to the TOP or ZERO value to determine whether the counter has reached TOP or ZERO. In either case, the TCC can generate interrupt requests or generate events for the Event System. In Waveform Generator mode, these comparisons are used to set the waveform period or pulse width.

A prescaled generic clock (GCLK_TCCx) and events from the event system can be used to control the counter. The event system is also used as a source to the input capture.

The Recoverable Fault Unit enables event-controlled waveforms by acting directly on the generated waveforms of the TCC compare channels output. These events can restart, halt the timer/counter period, shorten the output pulse active time, or disable waveform output as long as the fault condition is present. This can typically be used for current sensing regulation, and zero-crossing and demagnetization re-triggering.

The MCE0 and MCE1 asynchronous event sources are shared with the recoverable fault unit. Only asynchronous events are used internally when fault unit extension is enabled. See *Event System (EVSYS)* from Related Links for further details on how to configure asynchronous events routing.

Recoverable fault sources can be filtered and/or windowed to avoid false triggering, for example from I/O pin glitches, by using digital filtering, input blanking and qualification options. See *Recoverable Faults* from Related Links.

In order to support applications with different types of motor control, ballast, LED, H-bridge, power converter and other types of power switching applications, the following independent units are implemented in some of the TCC instances as optional and successive units:

- Recoverable faults and non-recoverable faults
- Output matrix
- Dead-time insertion
- Swap
- Pattern generation

See *Timer/Counter for Control Applications - Block Diagram* in the *Block Diagram* from Related Links.

The output matrix (OTMX) can distribute and route out the TCC waveform outputs across the port pins in different configurations, each optimized for different application types. The Dead-Time Insertion (DTI) unit splits the four lower OTMX outputs into two non-overlapping signals: the non-inverted Low Side (LS) and inverted High Side (HS) of the waveform output with optional dead-time insertion between LS and HS switching. The SWAP unit can swap the LS and HS pin outputs and can be used for fast decay motor control.

The pattern generation unit can be used to generate synchronized waveforms with constant logic level on TCC UPDATE conditions. This is useful for easy stepper motor and full bridge control.

The non-recoverable fault module enables event-controlled fault protection by acting directly on the generated waveforms of the timer/counter compare channel outputs. When a non-recoverable fault condition is detected, the output waveforms are forced to a preconfigured value that is safe for the application. This is typically used for instant and predictable shut-down and disabling high current or voltage drives.

The count event sources (TCE0 and TCE1) are shared with the non-recoverable fault extension. The events can be optionally filtered. If the filter options are not used, the non-recoverable faults provide an immediate asynchronous action on waveform output, even for cases where the clock is not present. See *Event System (EVSYS)* from Related Links for further details on how to configure asynchronous events routing.

Related Links

26. Event System (EVSYS)
36.3. Block Diagram
36.6.3.5. Recoverable Faults
36.6.2.5.1. Waveform Output Generation Operations

36.6.2 Basic Operation

36.6.2.1 Initialization

The following registers are enable-protected, meaning that they can only be written when the TCC is disabled (CTRLA.ENABLE=0):

- Control A (CTRLA) register, except Run Standby (RUNSTDBY), Enable (ENABLE) and Software Reset (SWRST) bits
- Recoverable Fault n Control registers (FCTRLA and FCTRLB)
- Waveform Extension Control register (WEXCTRL)
- Drive Control register (DRVCTRL)
- Event Control register (EVCTRL)

Enable-protected bits in the CTRLA register can be written at the same time as CTRLA.ENABLE is written to '1', but not at the same time as CTRLA.ENABLE is written to '0'. Enable-protection is denoted by the "Enable-Protected" property in the register description.

Before the TCC is enabled, it must be configured as outlined by the following steps:

1. Enable the TCC bus clock if not already enabled by default (PB1_CLK).
2. If Capture mode is required, enable the channel in Capture mode by writing a '1' to the Capture Enable bit in the Control A register (CTRLA.CPTEN).

Optionally, the following configurations can be set before enabling TCC:

1. Select PRESCALER setting in the Control A register (CTRLA.PRESCALER).
2. Select Prescaler Synchronization setting in Control A register (CTRLA.PRESCSYNC).
3. If down-counting operation is desired, write the Counter Direction bit in the Control B Set register (CTRLBSET.DIR) to '1'.
4. Select the Waveform Generation operation in the WAVE register (WAVE.WAVEGEN).
5. Select the Waveform Output Polarity in the WAVE register (WAVE.POL).
6. The waveform output can be inverted for the individual channels using the Waveform Output Invert Enable bit group in the Driver register (DRVCTRL.INVEN).

36.6.2.2 Enabling, Disabling, and Resetting

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

The TCC is reset by writing '1' to the Software Reset bit in the Control A register (CTRLA.SWRST). All registers in the TCC, except DBGCTRL, will be reset to their initial state, and the TCC will be disabled. See *CTRLA* register from Related Links.

The TCC should be disabled before the TCC is reset to avoid undefined behavior.

36.6.2.3 Prescaler Selection

The GCLK_TCCx clock is fed into the internal prescaler.

The prescaler consists of a counter that counts up to the selected prescaler value, whereupon the output of the prescaler toggles.

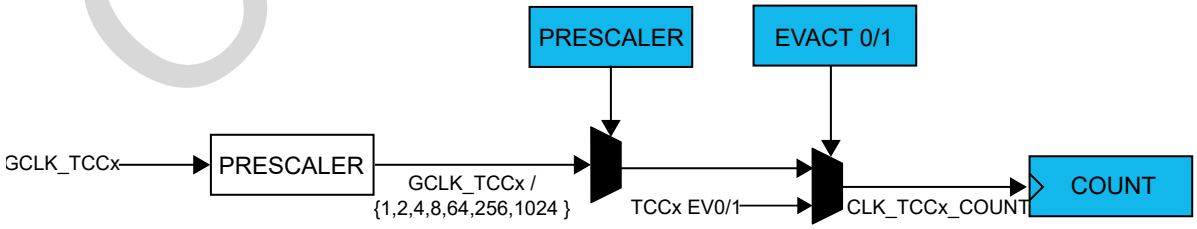
If the prescaler value is higher than one, the Counter Update condition can be optionally executed on the next GCLK_TCCx clock pulse or the next prescaled clock pulse. For further details, refer to the Prescaler (CTRLA.PRESCALER) and Counter Synchronization (CTRLA.PRESYNC) descriptions.

Prescaler outputs from 1 to 1/1024 are available. For a complete list of available prescaler outputs, see the register description for the Prescaler bit group in the Control A register (CTRLA.PRESCALER).

Note: When counting events, the prescaler is bypassed.

The joint stream of prescaler ticks and event action ticks is called CLK_TCCx_COUNT.

Figure 36-2. Prescaler



36.6.2.4 Counter Operation

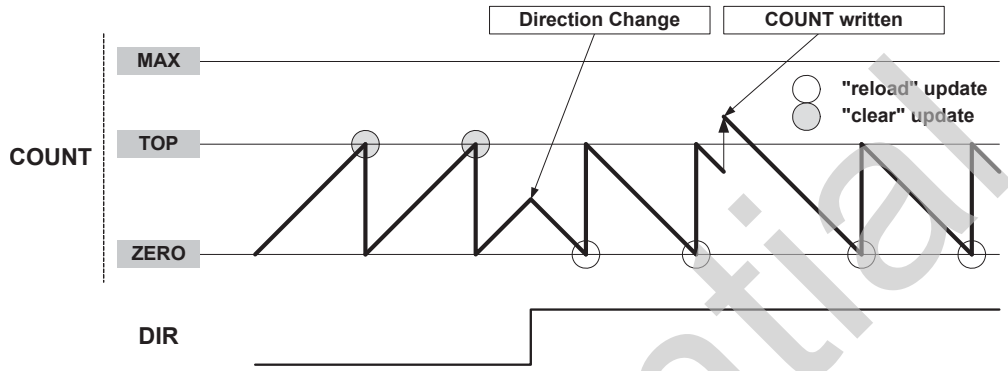
Depending on the mode of operation, the counter is cleared, reloaded, incremented, or decremented at each TCC clock input (CLK_TCCx_COUNT). A counter clear or reload mark the end of current counter cycle and the start of a new one.

The counting direction is set by the Direction bit in the Control B register (CTRLB.DIR). If the bit is zero, it's counting up and if one it's counting down.

The counter will count up or down for each tick (clock or event) until it reaches TOP or ZERO. When it's counting up and TOP is reached, the counter will be set to zero at the next tick (overflow) and the Overflow Interrupt Flag in the Interrupt Flag Status and Clear register (INTFLAG.OVF) will be set. When down-counting, the counter is reloaded with the TOP value when ZERO is reached (underflow), and INTFLAG.OVF is set.

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

Figure 36-3. Counter Operation



It is possible to change the counter value (by writing directly in the COUNT register) even when the counter is running. The COUNT value will always be ZERO or TOP, depending on direction set by CTRLBSET.DIR or CTRLBCLR.DIR, when starting the TCC, unless a different value has been written to it, or the TCC has been stopped at a value other than ZERO. The write access has higher priority than count, clear, or reload. The direction of the counter can also be changed during normal operation. See also Figure 36-3.

Stop Command

A stop command can be issued from software by using TCC Command bits in Control B Set register (CTRLBSET.CMD=0x2, STOP).

When a Stop is detected while the counter is running, the counter will not retain its current value. If the waveform generation (WG) is used, all waveforms are set to a state defined in Non-Recoverable State x Output Enable bit and Non- Recoverable State x Output Value bit in the Driver Control register (DRVCTRL.NREx and DRVCTRL.NRVx), and the Stop bit in the Status register is set (STATUS.STOP).

Pause Event Action

A pause command can be issued when the stop event action is configured in the Input Event Action 1 bits in Event Control register (EVCTRL.EVACT1=0x3, STOP).

When a pause is detected, the counter can stop immediatly maintaining its current value and all waveforms keep their current state, until a start event action is detected: Input Event Action 0 bits in Event Control register (EVCTRL.EVACT0=0x3, START).

Re-Trigger Command and Event Action

A re-trigger command can be issued from software by using TCC Command bits in Control B Set register (CTRLBSET.CMD=0x1, RETRIGGER), or from event when the re-trigger event action is configured in the Input Event 0/1 Action bits in Event Control register (EVCTRL.EVACTn=0x1, RETRIGGER).

When the command is detected during counting operation, the counter will be reloaded or cleared, depending on the counting direction (CTRLBSET.DIR or CTRLBCLR.DIR). The Re-Trigger bit in the Interrupt Flag Status and Clear register will be set (INTFLAG.TRG). It is also possible to generate an event by writing a '1' to the Re-Trigger Event Output Enable bit in the Event Control register (EVCTRL.TRGEO).If a re-trigger command is detected when the counter is stopped, the value in COUNT will be not be retained and will start either from ZERO or TOP depending on the direction set by CTRLBSET.DIR or CTRLBCLR.DIR.

Note:
When a re-trigger event action is configured in the Event Action bits in the Event Control register (EVCTRL.EVACTn=0x1, RETRIGGER), enabling the counter will not start the counter. The counter will start on the next incoming event and restart on corresponding following event.

Start Event Action

The start action can be selected in the Event Control register (EVCTRL.EVACT0=0x3, START) and can start the counting operation when previously stopped. The event has no effect if the counter is already counting. When the module is enabled, the counter operation starts when the event is received or when a re-trigger software command is applied.

Note:
When a start event action is configured in the Event Action bits in the Event Control register (EVCTRL.EVACT0=0x3, START), enabling the counter will not start the counter. The counter will start on the next incoming event, but it will not restart on subsequent events.

Count Event Action

The TCC can count events. When an event is received, the counter increases or decreases the value, depending on direction settings (CTRLBSET.DIR or CTRLBCLR.DIR).

The count event action is selected by the Event Action 0 bit group in the Event Control register (EVCTRL.EVACT0=0x5, COUNT).

Direction Event Action

The direction event action can be selected in the Event Control register (EVCTRL.EVACT1=0x2, DIR). When this event is used, the asynchronous event path specified in the event system must be configured or selected. The direction event action can be used to control the direction of the counter operation, depending on external events level. When received, the event level overrides the Direction settings (CTRLBSET.DIR or CTRLBCLR.DIR) and the direction bit value is updated accordingly.

Increment Event Action

The increment event action can be selected in the Event Control register (EVCTRL.EVACT0=0x4, INC) and can change the Counter state when an event is received. When the TCE0 event (TCCx_EV0) is received, the counter increments, whatever the direction setting (CTRLBSET.DIR or CTRLBCLR.DIR) is.

Decrement Event Action

The decrement event action can be selected in the Event Control register (EVCTRL.EVACT1=0x4, DEC) and can change the Counter state when an event is received. When the TCE1 (TCCx_EV1) event is received, the counter decrements, whatever the direction setting (CTRLBSET.DIR or CTRLBCLR.DIR) is.

Non-recoverable Fault Event Action

Non-recoverable fault actions can be selected in the Event Control register (EVCTRL.EVACTn=0x7, FAULT). When received, the counter will be stopped and the output of the compare channels is overridden according to the Driver Control register settings (DRVCTRL.NREx and DRVCTRL.NRVx). TCE0 and TCE1 must be configured as asynchronous events.

Event Action Off

If the event action is disabled (EVCTRL.EVACTn=0x0, OFF), enabling the counter will also start the counter.

36.6.2.5 Compare Operations

By default, the Compare/Capture channel is configured for compare operations. To perform capture operations, it must be re-configured.

When using the TCC with the Compare/Capture Value registers (CCx) for compare operations, the counter value is continuously compared to the values in the CCx registers. This can be used for timer or for waveform operation.

The Channel x Compare/Capture Buffer Value (CCBUFx) registers provide double buffer capability. The double buffering synchronizes the update of the CCx register with the buffer value at the UPDATE condition or a force update command (CTRLBSET.CMD=0x3, UPDATE). For further details, refer to 36.6.2.6. Double Buffering. The synchronization prevents the occurrence of odd-length, non-symmetrical pulses and ensures glitch-free output.

36.6.2.5.1 Waveform Output Generation Operations

The compare channels can be used for waveform generation on output port pins. To make the waveform available on the connected pin, the following requirements must be fulfilled:

1. Choose a Waveform Generation mode in the Waveform Generation Operation bit in Waveform register (WAVE.WAVEGEN).
2. Optionally, invert the waveform output WO[x] by writing the corresponding Waveform Output x Inversion bit in the Driver Control register (DRVCTRL.INVENx).
3. Configure the pins with the I/O Pin Controller. See *I/O Ports and Peripheral Pin Select (PPS)* from Related Links.
   Note: Event must not be used when the compare channel is set in waveform output operating mode.

The counter value is continuously compared with each CCx value. On a comparison match, the Match or Capture Channel x bit in the Interrupt Flag Status and Clear register (INTFLAG.MCx) will be set on the next zero-to-one transition of CLK_TCC_COUNT (see Normal Frequency Operation). An interrupt and/or event can be generated on the same condition if Match/Capture occurs, i.e., INTENSET.MCx and/or EVCTRL.MCEOx is '1'. Both interrupt and event can be generated simultaneously.

There are seven waveform configurations for the Waveform Generation Operation bit group in the Waveform register (WAVE.WAVEGEN). This will influence how the waveform is generated and impose restrictions on the top value. The configurations are:

- Normal Frequency (NFRQ)
- Match Frequency (MFRQ)
- Normal Pulse-Width Modulation (NPWM)
- Dual-slope, interrupt/event at TOP (DSTOP)
- Dual-slope, interrupt/event at ZERO (DSBOTTOM)
- Dual-slope, interrupt/event at Top and ZERO (DSBOTH)
- Dual-slope, critical interrupt/event at ZERO (DSCRITICAL)

When using MFRQ configuration, the TOP value is defined by the CC0 register value. For the other waveform operations, the TOP value is defined by the Period (PER) register value.

For dual-slope waveform operations, the update time occurs when the counter reaches ZERO. For the other Waveforms Generation modes, the update time occurs on counter wraparound, on overflow, underflow or re-trigger.

The table below shows the update counter and overflow event/interrupt generation conditions in different operation modes.

Table 36-4. Counter Update and Overflow Event/interrupt Conditions

Name	Operation	TOP	Update	Output Waveform		OVFIF/Event	
				On Match	On Update	Up	Down
NFRQ	Normal Frequency	PER	TOP/ ZERO	Toggle	Stable	TOP	ZERO
MFRQ	Match Frequency	CC0	TOP/ ZERO	Toggle	Stable	TOP	ZERO
NPWM	Single-slope PWM	PER	TOP/ ZERO	See section 'Output Polarity' below		TOP	ZERO
DSCRITICAL	Dual-slope PWM	PER	ZERO			—	ZERO
DSBOTTOM	Dual-slope PWM	PER	ZERO			—	ZERO
DSBOTH	Dual-slope PWM	PER	TOP(1) & ZERO			TOP	ZERO
DSTOP	Dual-slope PWM	PER	ZERO			TOP	—

1. The UPDATE condition on TOP only will occur when circular buffer is enabled for the channel.

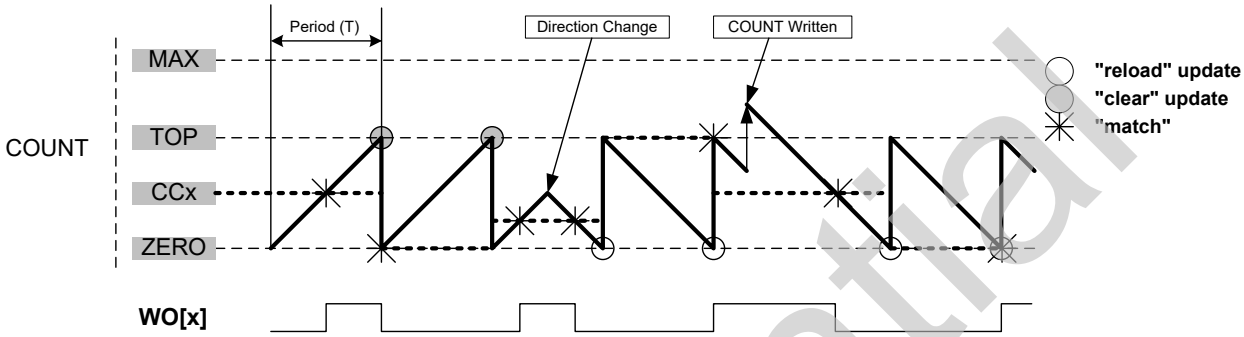
Related Links

5. I/O Ports and Peripheral Pin Select (PPS)

36.6.2.5.2 Normal Frequency (NFRQ)

For Normal Frequency generation, the period time (T) is controlled by the period register (PER). The waveform generation output (WO[x]) is toggled on each compare match between COUNT and CCx, and the corresponding Match or Capture Channel x Interrupt Flag (EVCTRL.MCEOx) will be set.

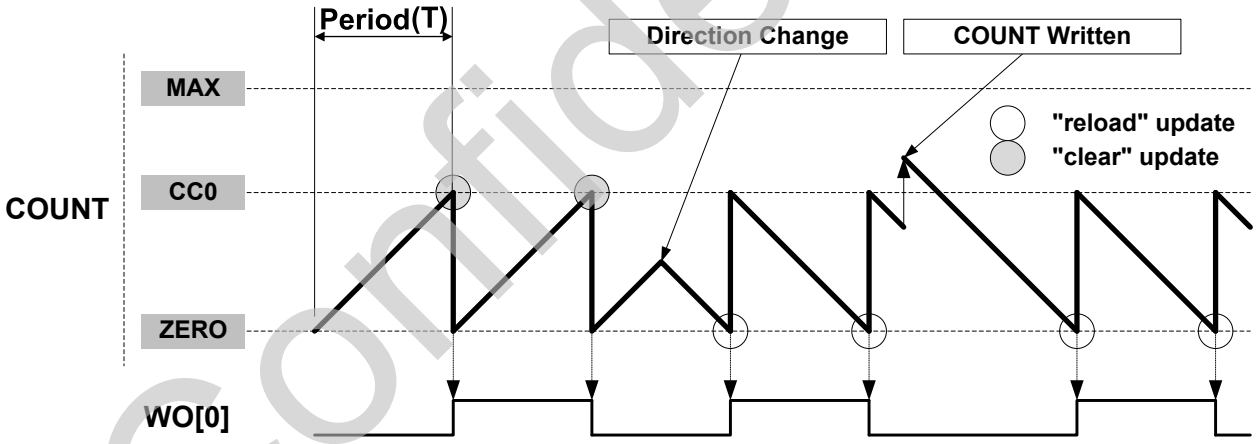
Figure 36-4. Normal Frequency Operation



36.6.2.5.3 Match Frequency (MFRQ)

For Match Frequency generation, the period time (T) is controlled by CC0 register instead of PER. WO[0] toggles on each update condition.

Figure 36-5. Match Frequency Operation



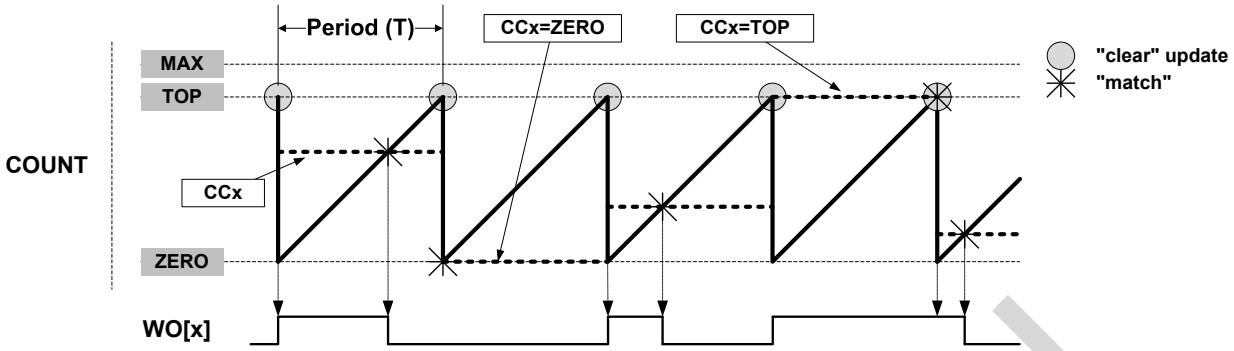
36.6.2.5.4 Normal Pulse-Width Modulation (NPWM)

NPWM uses single-slope PWM generation.

36.6.2.5.5 Single-Slope PWM Operation

For single-slope PWM generation, the period time (T) is controlled by Top value, and CCx controls the duty cycle of the generated waveform output. When up-counting, the WO[x] is set at start or compare match between the COUNT and TOP values, and cleared on compare match between COUNT and CCx register values. When down-counting, the WO[x] is cleared at start or compare match between the COUNT and ZERO values, and set on compare match between COUNT and CCx register values.

Figure 36-6. Single-Slope PWM Operation



The following equation calculates the exact resolution for a single-slope PWM (R_{PWM_SS}) waveform:

$$R_{PWM\_SS} = \frac{\log(TOP+1)}{\log(2)}$$

The PWM frequency depends on the Period register value (PER) and the peripheral clock frequency (f_{GCLK_TCCx}), and can be calculated by the following equation:

$$f_{PWM\_SS} = \frac{f_{GCLK\_TCCx}}{N(TOP+1)}$$

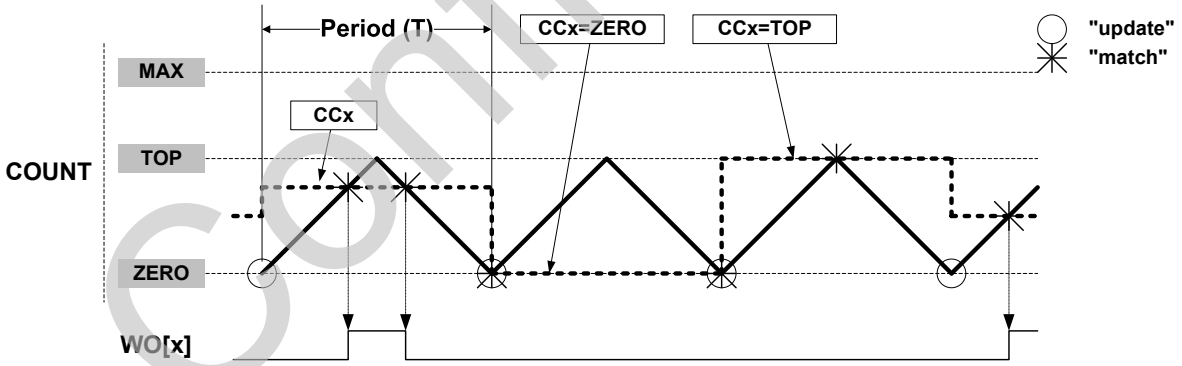
Where N represents the prescaler divider used (1, 2, 4, 8, 16, 64, 256, 1024).

36.6.2.5.6 Dual-Slope PWM Generation

For dual-slope PWM generation, the period setting (TOP) is controlled by PER, while CCx control the duty cycle of the generated waveform output. The figure below shows how the counter repeatedly counts from ZERO to PER and then from PER to ZERO. The waveform generator output is set on compare match when up-counting, and cleared on compare match when down-counting. An interrupt and/or event is generated on TOP (when counting upwards) and/or ZERO (when counting up or down).

In DSBOTH operation, the circular buffer must be enabled to enable the update condition on TOP.

Figure 36-7. Dual-Slope Pulse Width Modulation



Using dual-slope PWM results in a lower maximum operation frequency compared to single-slope PWM generation. The period (TOP) defines the PWM resolution. The minimum resolution is 1 bit (TOP=0x00000001).

The following equation calculates the exact resolution for dual-slope PWM (R_{PWM_DS}):

$$R_{PWM\_DS} = \frac{\log(PER+1)}{\log(2)}.$$

The PWM frequency f_{PWM_DS} depends on the period setting (TOP) and the peripheral clock frequency f_{GCLK_TCCx}, and can be calculated by the following equation:

$$f_{PWM\_DS} = \frac{f_{GCLK\_TCCx}}{2N \cdot PER}$$

N represents the prescaler divider used. The waveform generated will have a maximum frequency of half of the TCC clock frequency (f_{GCLK_TCCx}) when TOP is set to 0x00000001 and no prescaling is used.

The pulse width (P_{PWM_DS}) depends on the compare channel (CCx) register value and the peripheral clock frequency (f_{GCLK_TCCx}), and can be calculated by the following equation:

$$P_{PWM\_DS} = \frac{2N \cdot (TOP - CCx)}{f_{GCLK\_TCCx}}$$

N represents the prescaler divider used.

Note: In DSTOP, DSBOTTOM and DSBOTH operation, when TOP is lower than MAX/2, the CCx MSB bit defines the ramp on which the CCx Match interrupt or event is generated. (Rising if CCx[MSB] = 0, falling if CCx[MSB] = 1.)

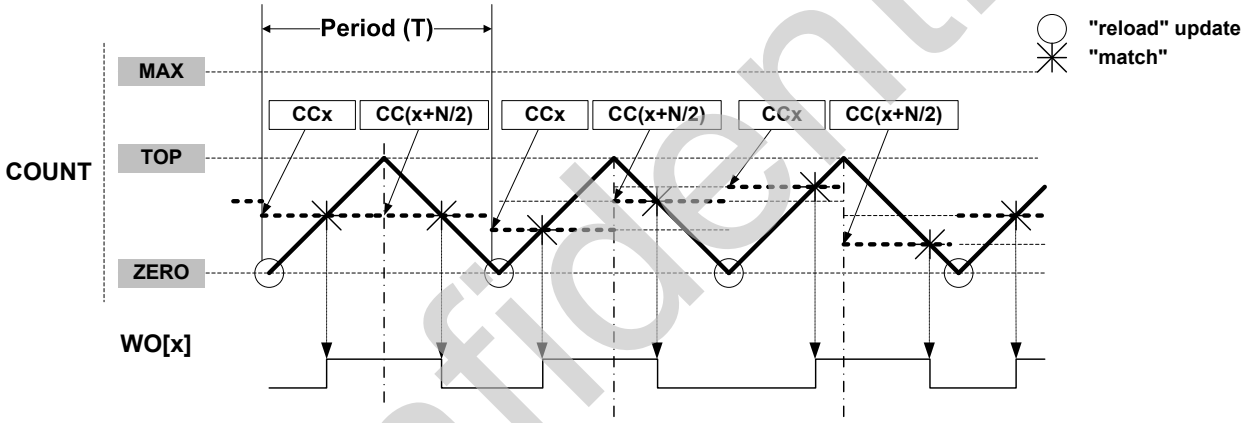
Related Links

36.6.3.2. Circular Buffer

36.6.2.5.7 Dual-Slope Critical PWM Generation

Critical mode generation allows generation of non-aligned centered pulses. In this mode, the period time is controlled by PER while CCx control the generated waveform output edge during up-counting and CC(x+CC_NUM/2) control the generated waveform output edge during down-counting.

Figure 36-8. Dual-Slope Critical Pulse Width Modulation (N=CC_NUM)



36.6.2.5.8 Output Polarity

The polarity (WAVE.POLx) is available in all waveform output generation. In single-slope and dual-slope PWM operation, it is possible to invert the pulse edge alignment individually on start or end of a PWM cycle for each compare channels. The table below shows the waveform output set/clear conditions, depending on the settings of timer/counter, direction, and polarity.

Table 36-5. Waveform Generation Set/Clear Conditions

Waveform Generation Operation	DIR	POLx	Waveform Generation Output Update	
			Set	Clear
Single-Slope PWM	0	0	Timer/counter matches TOP	Timer/counter matches CCx
		1	Timer/counter matches CC	Timer/counter matches TOP
	1	0	Timer/counter matches CC	Timer/counter matches ZERO
		1	Timer/counter matches ZERO	Timer/counter matches CC
Dual-Slope PWM	x	0	Timer/counter matches CC when counting up	Timer/counter matches CC when counting down
		1	Timer/counter matches CC when counting down	Timer/counter matches CC when counting up

In Normal and Match Frequency, the WAVE.POLx value represents the initial state of the waveform output.

36.6.2.6 Double Buffering

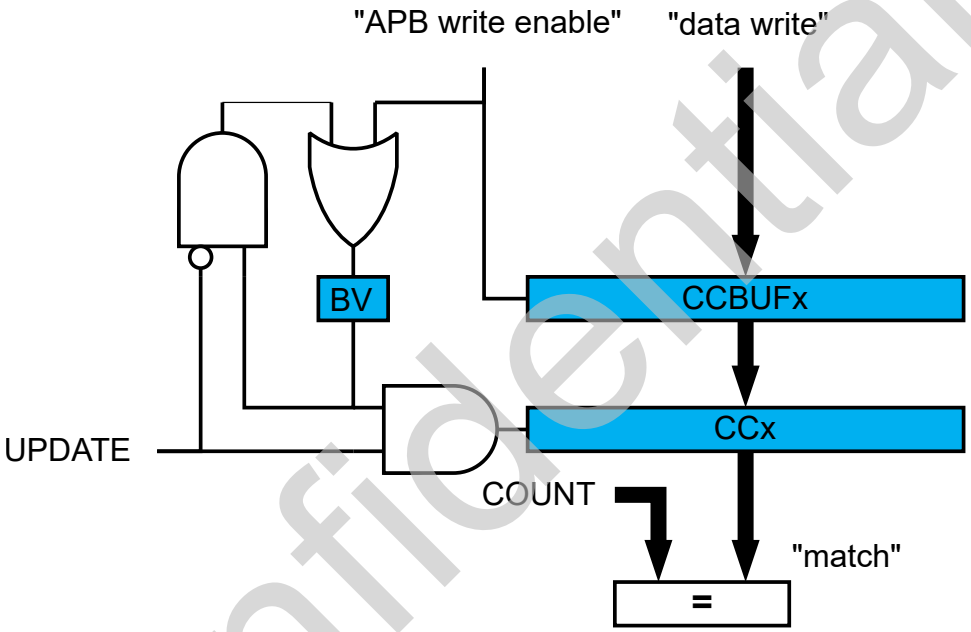
The Pattern (PATT), Period (PER) and Compare Channels (CCx) registers are all double buffered. Each buffer register has a buffer valid (PATTBUFV, PERBUFV and CCBUFVx) bit in the STATUS register, which indicates that the Buffer register contains a valid value that can be copied into the corresponding register. .

When the Buffer Valid Flag bit in the STATUS register is '1' and the Lock Update bit in the CTRLB register is set to '0', (writing CTRLBCLR.LUPD to '1'), double buffering is enabled: the data from buffer registers will be copied into the corresponding register under hardware UPDATE conditions, then the Buffer Valid flags bit in the STATUS register are automatically cleared by hardware.

Note: Software update command (CTRLBSET.CMD=0x3) act independently of LUPD value.

A compare register is double buffered as in the following figure.

Figure 36-9. Compare Channel Double Buffering



Both the registers (PATT/PER/CCx) and corresponding Buffer registers (PATTBUFPERBUF/CCBUFx) are available in the I/O register map, and the double buffering feature is not mandatory. The double buffering is disabled by writing a '1' to CTRLSET.LUPD.

Note: In NFRQ, MFRQ or PWM Down-Counting Counter mode (CTRLBSET.DIR=1), when double buffering is enabled (CTRLBCLR.LUPD=1), PERBUF register is continuously copied into the PER independently of update conditions.

Changing the Period

The counter period can be changed by writing a new Top value to the Period register (PER or CC0, depending on the Waveform Generation mode), any period update on registers (PER or CCx) is effective after the synchronization delay, whatever double buffering enabling is.

Figure 36-10. Unbuffered Single-Slope Up-Counting Operation

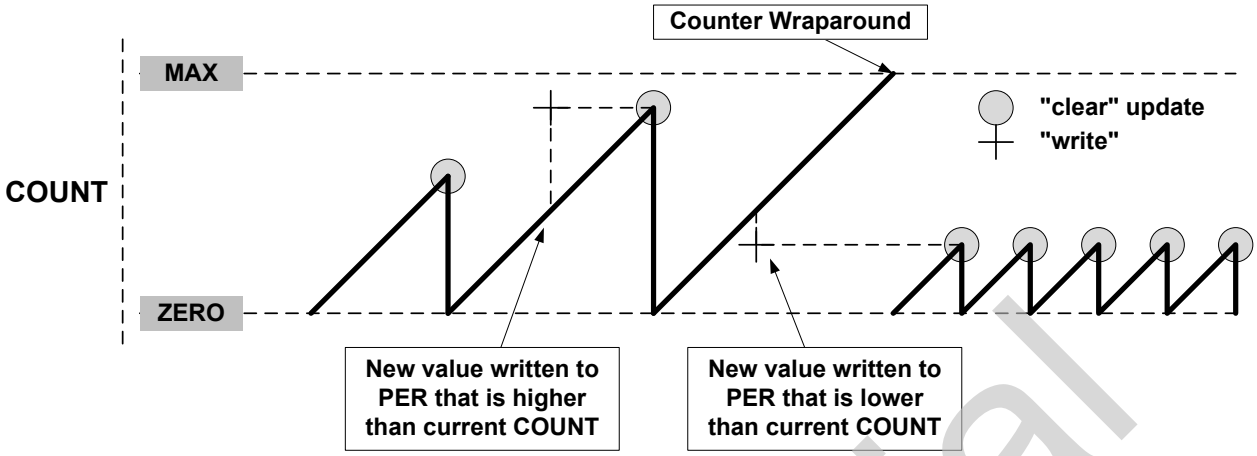
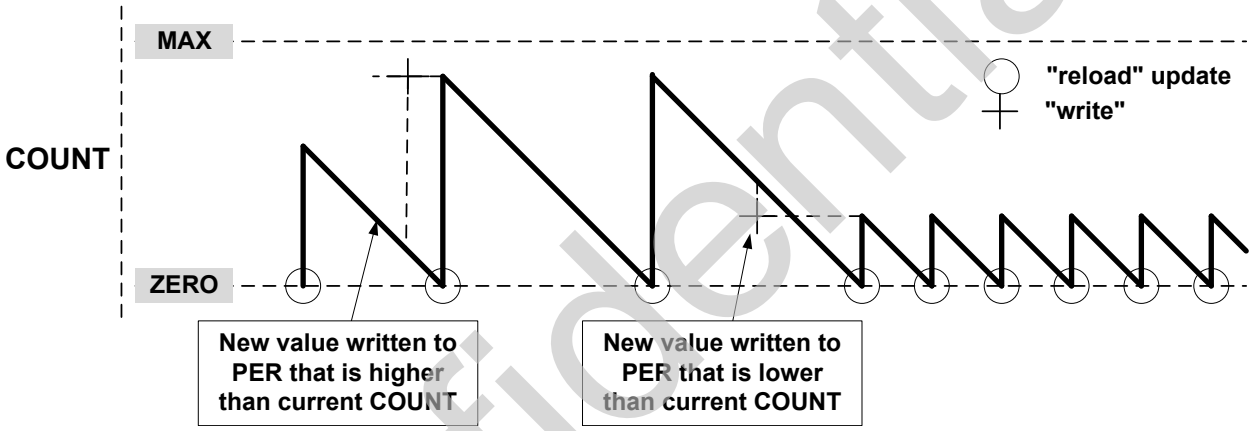
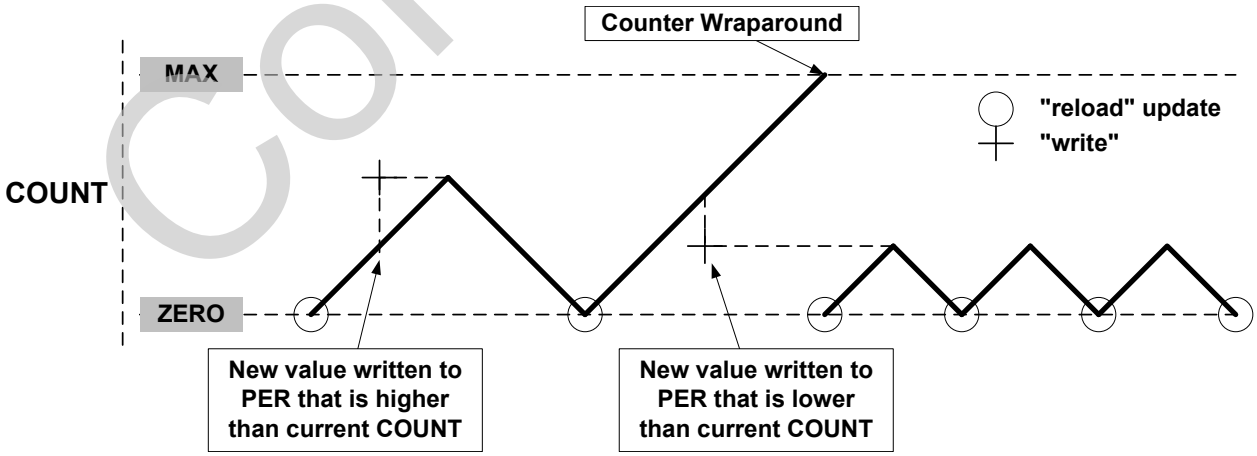


Figure 36-11. Unbuffered Single-Slope Down-Counting Operation



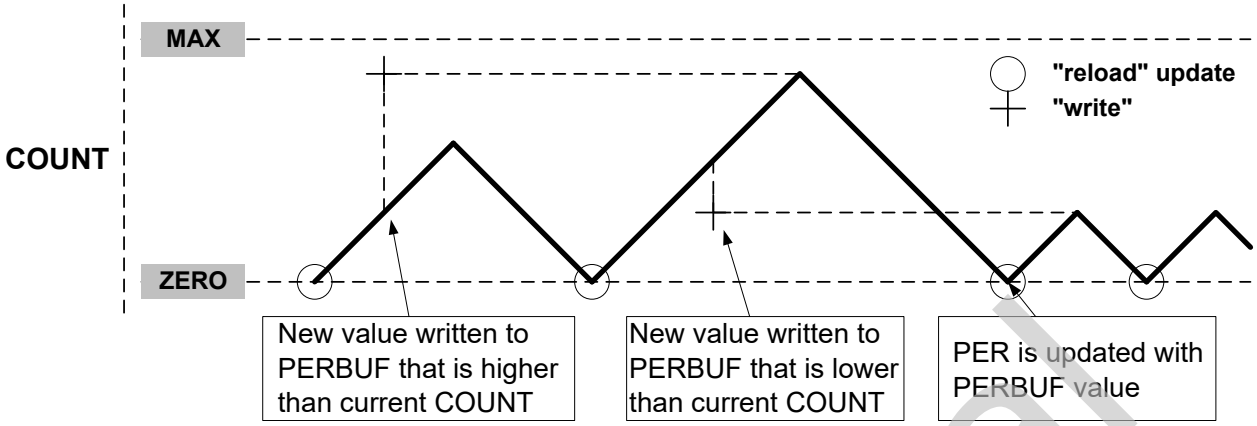
A counter wraparound can occur in any operation mode when up-counting without buffering, see Figure 36-10. COUNT and TOP are continuously compared, so when a new value that is lower than the current COUNT is written to TOP, COUNT will wrap before a compare match.

Figure 36-12. Unbuffered Dual-Slope Operation



When double buffering is used, the buffer can be written at any time and the counter will still maintain correct operation. The period register is always updated on the update condition, as shown in Figure 36-13. This prevents wraparound and the generation of odd waveforms.

Figure 36-13. Changing the Period Using Buffering



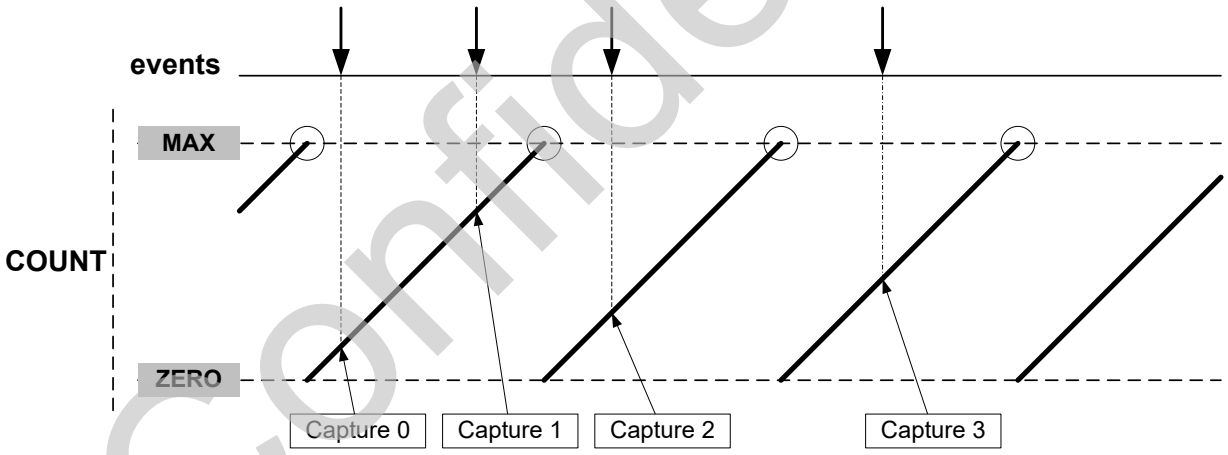
36.6.2.7 Capture Operations

To enable and use capture operations, the Match or Capture Channel x Event Input Enable bit in the Event Control register (EVCTRL.MCEIx) must be written to '1'. The capture channels to be used must also be enabled in the Capture Channel x Enable bit in the Control A register (CTRLA.CPTENx) before capturing can be performed.

Event Capture Action

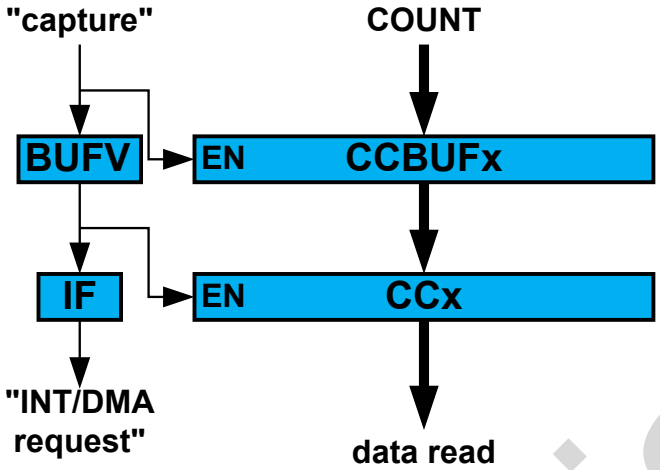
The compare/capture channels can be used as input capture channels to capture events from the Event System, and give them a timestamp. The following figure shows four capture events for one capture channel. Event system channels must be configured to operate in asynchronous mode when used for capture operations.

Figure 36-14. Input Capture Timing



For input capture, the Buffer register and the corresponding CCx act like a FIFO. When CCx is empty or read, any content in CCBUFx is transferred to CCx. The Buffer Valid flag is passed to set the CCx Interrupt flag (IF) and generate the optional interrupt, event, or DMA request. The CCBUFx register value cannot be read, all captured data must be read from the CCx register.

Figure 36-15. Capture Double Buffering



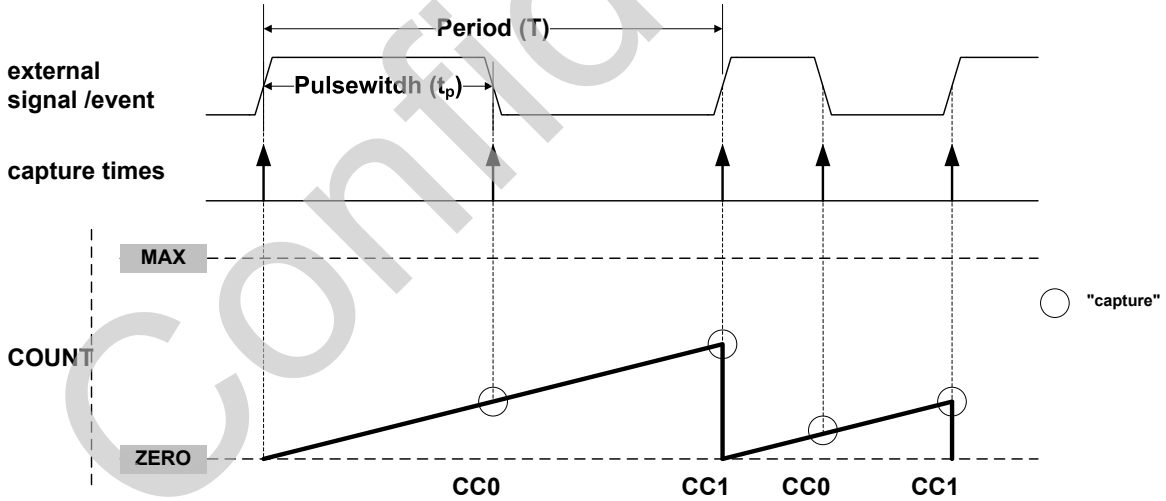
The TCC can detect capture overflow of the input capture channels. When a new capture event is detected while the Capture Buffer Valid flag (STATUS.CCBUFV) is still set, the new timestamp will not be stored and INTFLAG.ERR will be set.

Period and Pulse-Width (PPW) Capture Action

The TCC can perform two input captures and restart the counter on one of the edges. This enables the TCC to measure the pulse-width and period and to characterize the frequency *f* and *dutyCycle* of an input signal, as shown below:

$$f = \frac{1}{T} \quad , \quad dutyCycle = \frac{t_p}{T}$$

Figure 36-16. PWP Capture



Selecting PWP or PPW in the Timer/Counter Event Input 1 Action bit group in the Event Control register (EVCTRL.EVACT1) enables the TCC to perform one capture action on the rising edge and the other one on the falling edge. When using PPW event action, period *T* will be captured into CC0 and the pulse-width t_p into CC1. The PWP (Pulse-width and Period) event action offers the same functionality, but *T* will be captured into CC1 and t_p into CC0.

The Timer/Counter Event x Invert Enable bit in Event Control register (EVCTRL.TCEINVx) is used for event source x to select whether the wraparound should occur on the rising edge or the falling edge. If EVCTRL.TCEINVx = 1, the wraparound will happen on the falling edge.

The corresponding capture is done only if the channel is enabled in Capture mode (CTRLA.CPTENx = 1). If not, the capture action will be ignored and the channel will be enabled in compare mode of operation. When only one of these channels is required, the other channel can be used for other purposes.

The TCC can detect capture overflow of the input capture channels. When a new capture event is detected while the INTFLAG.MCx is still set, the new timestamp will not be stored and INTFLAG.ERR will be set.

Note: When up-counting (CTRLBSET.DIR = 0), counter values lower than 1 cannot be captured in Capture Minimum mode (FCTRLn.CAPTURE = CAPTMIN). To capture the full range including value 0, the TCC must be configured in Down-counting mode (CTRLBSET.DIR = 0).

Note: In dual-slope PWM operation, and when TOP is lower than MAX/2, the CCx MSB captures the CTRLB.DIR state to identify the ramp on which the capture has been done. For rising ramps CCx[MSB] is zero, for falling ramps CCx[MSB] = 1.

36.6.3 Additional Features

36.6.3.1 One-Shot Operation

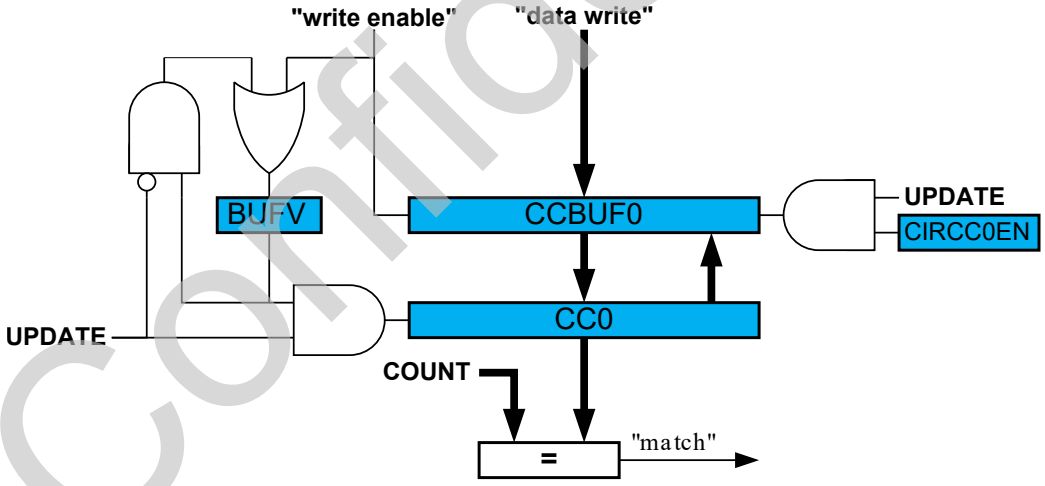
When one-shot is enabled, the counter automatically stops on the next Counter Overflow or Underflow condition. When the counter is stopped, the Stop bit in the Status register (STATUS.STOP) is set and the waveform outputs are set to the value defined by DRVCTRL.NREx and DRVCTRL.NRVx.

One-shot operation can be enabled by writing a '1' to the One-Shot bit in the Control B Set register (CTRLBSET.ONESHOT) and disabled by writing a '1' to CTRLBCLR.ONESHOT. When enabled, the TCC will count until an overflow or underflow occurs and stop counting. The one-shot operation can be restarted by a re-trigger software command, a re-trigger event or a start event. When the counter restarts its operation, STATUS.STOP is automatically cleared.

36.6.3.2 Circular Buffer

The Period register (PER) and the Compare Channels register (CC0 toCC5) support circular buffer operation. When circular buffer operation is enabled, the PER or CCx values are copied into the corresponding buffer registers at each update condition. Circular buffering is dedicated to RAMP2, RAMP2A, and DSBOTH operations.

Figure 36-17. Circular Buffer on Channel 0



36.6.3.3 Dithering Operation

The TCC supports dithering on Pulse-width or Period on a 16, 32 or 64 PWM cycles frame.

Dithering consists in adding some extra clocks cycles in a frame of several PWM cycles, and can improve the accuracy of the *average* output pulse width and period. The extra clock cycles are added on some of the compare match signals, one at a time, through a "blue noise" process that minimizes the flickering on the resulting dither patterns.

Dithering is enabled by writing the corresponding configuration in the Enhanced Resolution bits in CTRLA register (CTRLA.RESOLUTION):

- DITH4 enable dithering every 16 PWM frames
- DITH5 enable dithering every 32 PWM frames
- DITH6 enable dithering every 64 PWM frames

The DITHERCY bits of COUNT, PER and CCx define the number of extra cycles to add into the frame (DITHERCY bits from the respective COUNT, PER or CCx registers). The remaining bits of COUNT, PER, CCx define the compare value itself.

The pseudo code, giving the extra cycles insertion regarding the cycle is:

```
int extra_cycle(resolution, dithercy, cycle){
  int MASK;
  int value
  switch (resolution){
    DITH4: MASK = 0x0f;
    DITH5: MASK = 0x1f;
    DITH6: MASK = 0x3f;
  }
  value = cycle * dithercy;
  if (((MASK & value) + dithercy) > MASK)
    return 1;
 return 0;
}
```

Dithering on Period

Writing DITHERCY in PER will lead to an average PWM period configured by the following formulas.

DITH4 mode:

$$PwmPeriod = \left(\frac{\text{DITHERCY}}{16} + \text{PER}\right)\left(\frac{1}{f_{\text{GCLK\_TCC}x}}\right)$$

Note: If DITH4 mode is enabled, the last 4 significant bits from PER/CCx or COUNT register correspond to the DITHERCY value, rest of the bits corresponds to PER/CCx or COUNT value.

DITH5 mode:

$$PwmPeriod = \left(\frac{\text{DITHERCY}}{32} + \text{PER}\right)\left(\frac{1}{f_{\text{GCLK\_TCC}x}}\right)$$

DITH6 mode:

$$PwmPeriod = \left(\frac{\text{DITHERCY}}{64} + \text{PER}\right)\left(\frac{1}{f_{\text{GCLK\_TCC}x}}\right)$$

Dithering on Pulse-Width

Writing DITHERCY in CCx will lead to an average PWM pulse width configured by the following formula.

DITH4 mode:

$$PwmPulseWidth = \left(\frac{\text{DITHERCY}}{16} + \text{CCx}\right)\left(\frac{1}{f_{\text{GCLK\_TCC}x}}\right)$$

DITH5 mode:

$$PwmPulseWidth = \left(\frac{\text{DITHERCY}}{32} + \text{CCx}\right)\left(\frac{1}{f_{\text{GCLK\_TCC}x}}\right)$$

DITH6 mode:

$$PwmPulseWidth = \left(\frac{\text{DITHERCY}}{64} + \text{CCx}\right)\left(\frac{1}{f_{\text{GCLK\_TCC}x}}\right)$$

Note: The PWM period will remain static in this case.

36.6.3.4 Ramp Operations

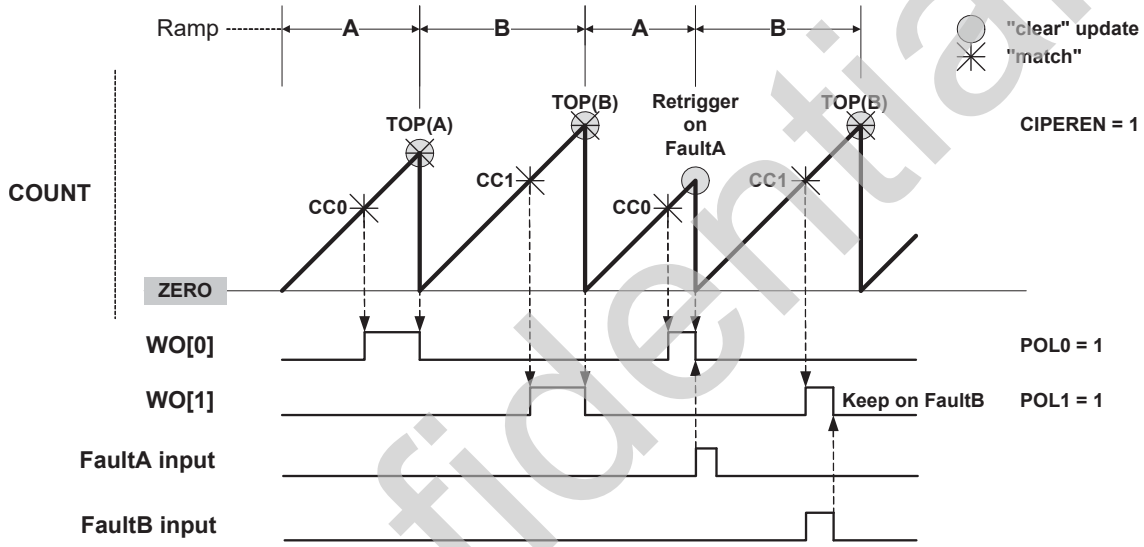
Three ramp operation modes are supported. All of them require the timer/counter running in single-slope PWM generation. The Ramp mode is selected by writing to the Ramp Mode bits in the Waveform Control register (WAVE.RAMP).

RAMP1 Operation
This is the default PWM operation, described in Single-Slope PWM Generation.

RAMP2 Operation
These operation modes are dedicated for power factor correction (PFC), Half-Bridge and Push-Pull SMPS topologies, where two consecutive timer/counter cycles are interleaved, see Figure 36-18. In cycle A, odd channel output is disabled, and in cycle B, even channel output is disabled. The ramp index changes after each update, but can be software modified using the Ramp index command bits in Control B Set register (CTRLBSET.IDXCMD).

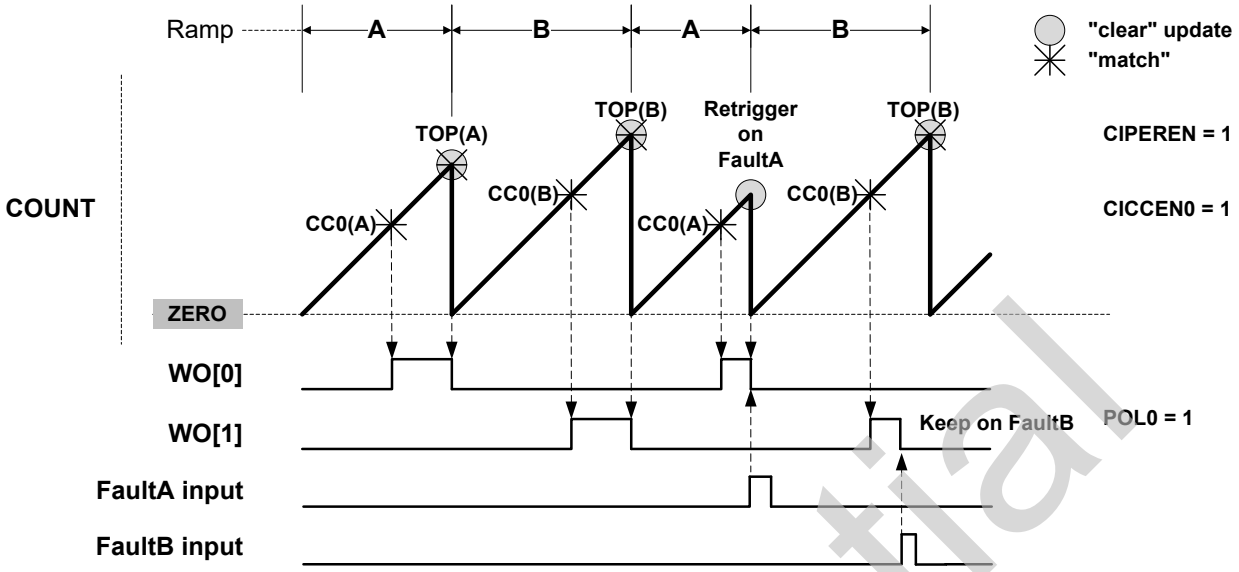
Standard RAMP2 (RAMP2) Operation
Ramp A and B periods are controlled by the PER register value. The PER value can be different on each ramp by the Circular Period buffer option in the Wave register (WAVE.CIPEREN=1). This mode uses a two-channel TCC to generate two output signals, or one output signal with another CC channel enabled in Capture mode.

Figure 36-18. RAMP2 Standard Operation



Alternate RAMP2 (RAMP2A) Operation
Alternate RAMP2 operation is similar to RAMP2, but CC0 controls both WO[0] and WO[1] waveforms when the corresponding circular buffer option is enabled (CIPEREN=1). The waveform polarity is the same on both outputs. Channel 1 can be used in capture mode.

Figure 36-19. RAMP2 Alternate Operation



Critical RAMP2 (RAMP2C) Operation

Critical RAMP2 operation provides a way to cover RAMP2 operation requirements without the update constraint associated with the use of circular buffers. In this mode, CC0 is controlling the period of ramp A and PER is controlling the period of ramp B. When using more than two channels, WO[0] output is controlled by CC2 (HIGH) and CC0 (LOW). On TCC with 2 channels, a pulse on WO[0] will last the entire period of ramp A, if WAVE.POL0=0.

Figure 36-20. RAMP2 Critical Operation With More Than 2 Channels

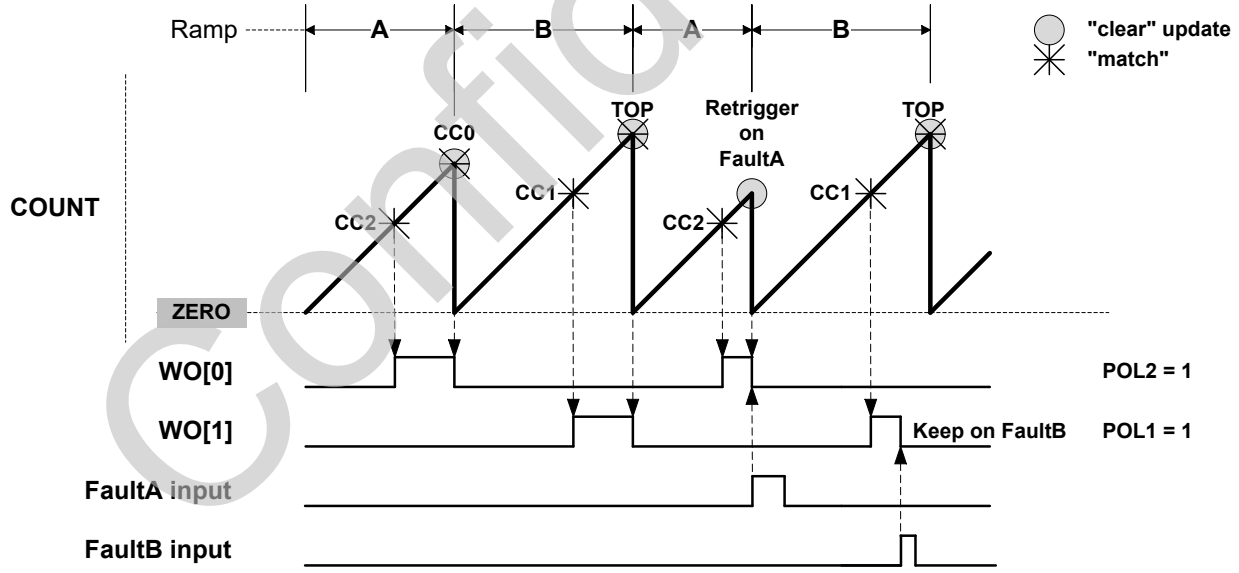
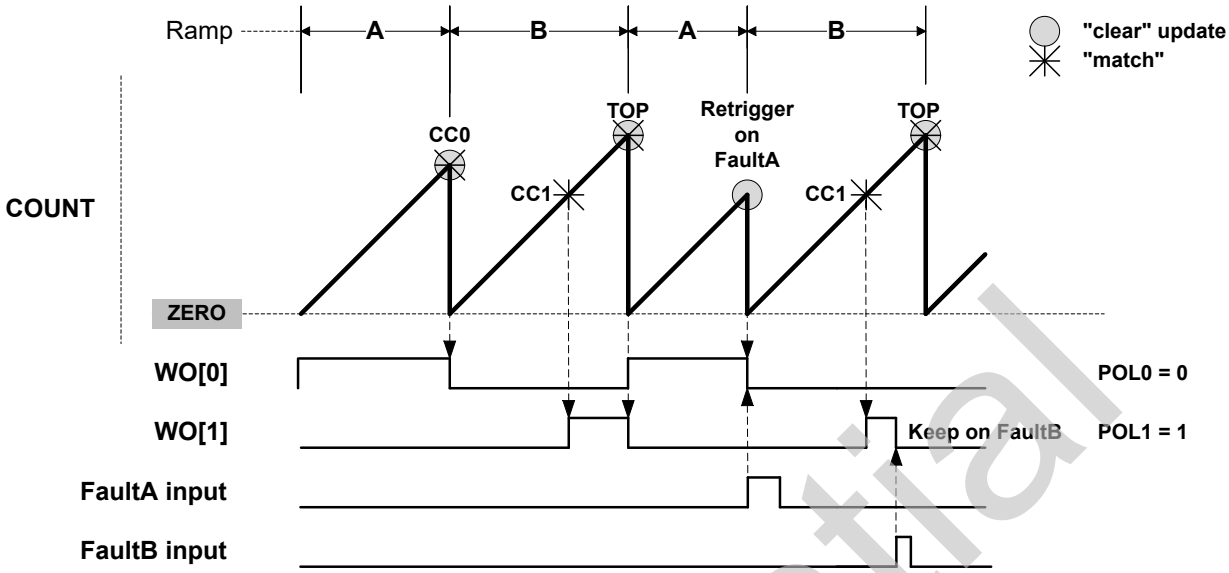


Figure 36-21. RAMP2 Critical Operation With 2 Channels



36.6.3.5 Recoverable Faults

Recoverable faults can restart or halt the timer/counter. Two faults, called Fault A and Fault B, can trigger recoverable fault actions on the compare channels CC0 and CC1 of the TCC. The compare channels' outputs can be clamped to inactive state either as long as the fault condition is present, or from the first valid fault condition detection on until the end of the timer/counter cycle.

Fault Inputs

The first two channel input events (TCCxMC0 and TCCxMC1) can be used as Fault A and Fault B inputs, respectively. Event system channels connected to these fault inputs must be configured as asynchronous. The TCC must work in a PWM mode.

Fault Filtering

There are three filters available for each input Fault A and Fault B. They are configured by the corresponding Recoverable Fault n Configuration registers (FCTRLA and FCTRLB). The three filters can either be used independently or in any combination.

Input Filtering

By default, the event detection is asynchronous. When the event occurs, the fault system will immediately and asynchronously perform the selected fault action on the compare channel output, also in device power modes where the clock is not available. To avoid false fault detection on external events (e.g. due to a glitch on an I/O port) a digital filter can be enabled and configured by the Fault B Filter Value bits in the Fault n Configuration registers (FCTRLn.FILTERVAL). If the event width is less than FILTERVAL (in clock cycles), the event will be discarded. A valid event will be delayed by FILTERVAL clock cycles.

Fault Blanking

This ignores any fault input for a certain time just after a selected waveform output edge. This can be used to prevent false fault triggering due to signal bouncing, as shown in the figure below. Blanking can be enabled by writing an edge triggering configuration to the Fault n Blanking Mode bits in the Recoverable Fault n Configuration register (FCTRLn.BLANK). The desired duration of the blanking must be written to the Fault n Blanking Time bits (FCTRLn.BLANKVAL).
The blanking time $t_b$is calculated by

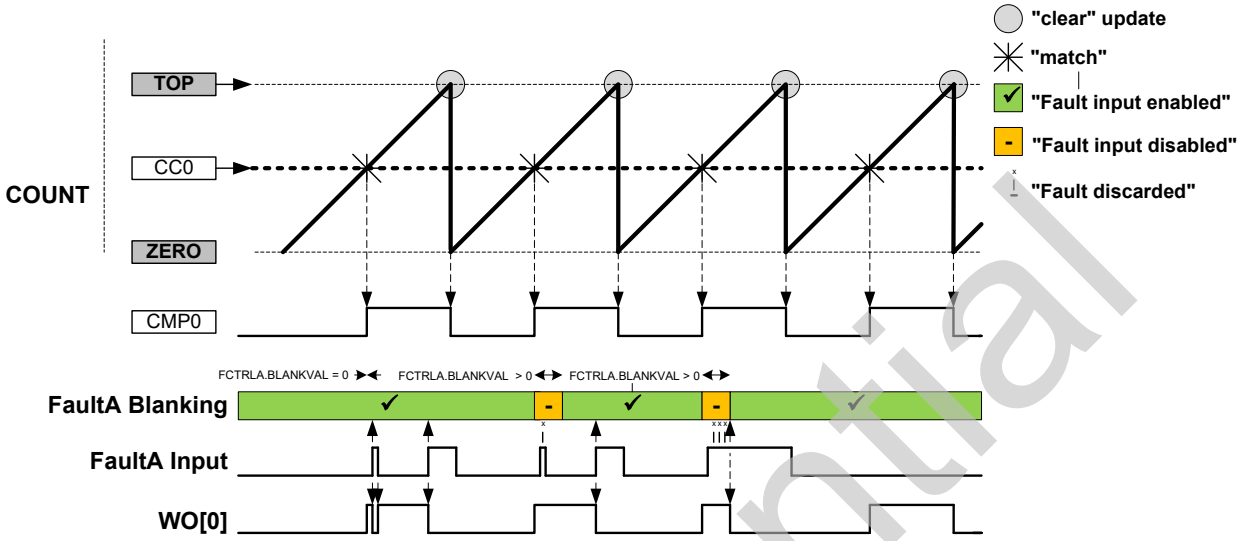
$$t_b = \frac{1 + \mathrm{BLANKVAL}}{f_{\mathrm{GCLK\_TCCx\_PRESC}}}$$

Here, $f_{\mathrm{GCLK\_TCCx\_PRESC}}$ is the frequency of the prescaled peripheral clock frequency $f_{\mathrm{GCLK\_TCCx}}$.

The prescaler is enabled by writing '1' to the Fault n Blanking Prescaler bit (FCTRLn.BLANKPRESC). When disabled, $f_{\mathrm{GCLK\_TCCx\_PRESC}}=f_{\mathrm{GCLK\_TCCx}}$. When enabled, $f_{\mathrm{GCLK\_TCCx\_PRESC}}=f_{\mathrm{GCLK\_TCCx}}/64$.

The maximum blanking time (FCTRLn.BLANKVAL = 255) at f_{GCLK_TCCx}= 64 MHz is 4 µs (no prescaler) or 256 µs (prescaling). For f_{GCLK_TCCx}=1 MHz, the maximum blanking time is either 256 µs (no prescaling) or 16.4 ms (prescaling enabled).

Figure 36-22. Fault Blanking in RAMP1 Operation with Inverted Polarity



Fault Qualification

This is enabled by writing a '1' to the Fault n Qualification bit in the Recoverable Fault n Configuration register (FCTRLn.QUAL). When the recoverable fault qualification is enabled (FCTRLn.QUAL=1), the fault input is disabled all the time the corresponding channel output has an inactive level, as shown in the figures below.

Figure 36-23. Fault Qualification in RAMP1 Operation

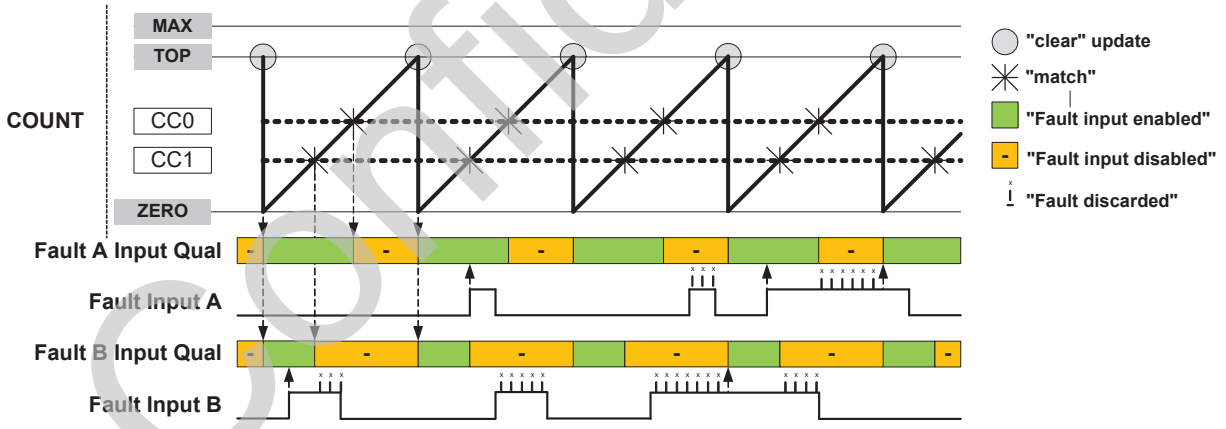
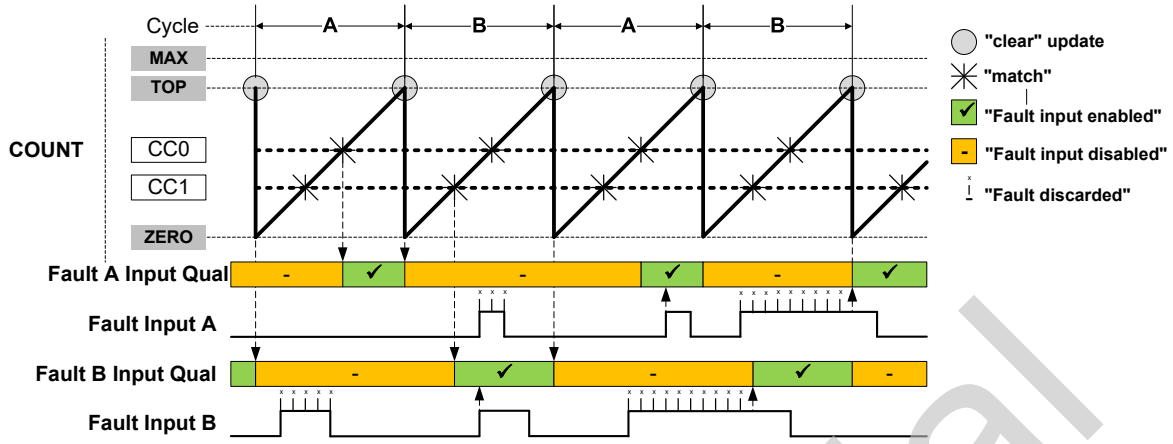


Figure 36-24. Fault Qualification in RAMP2 Operation with Inverted Polarity



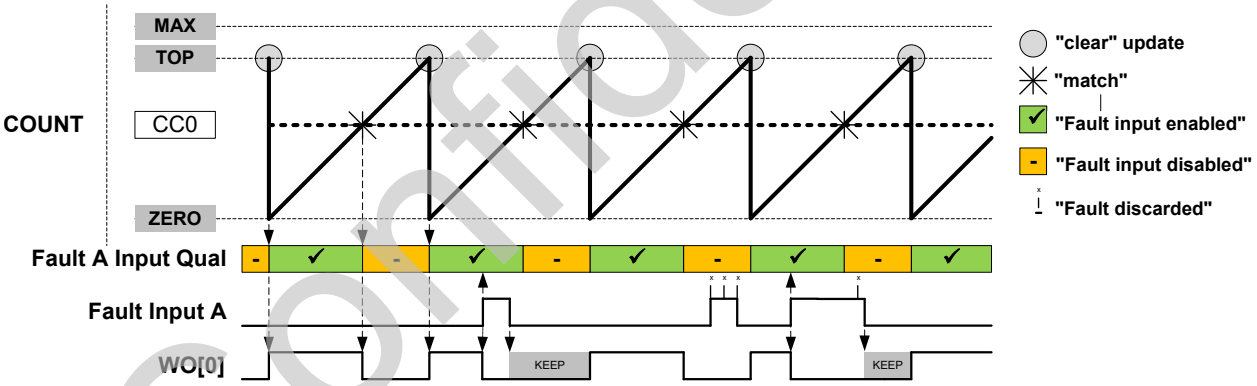
Fault Actions

Different fault actions can be configured individually for Fault A and Fault B. Most fault actions are not mutually exclusive; hence two or more actions can be enabled at the same time to achieve a result that is a combination of fault actions.

Keep Action

This is enabled by writing the Fault n Keeper bit in the Recoverable Fault n Configuration register (FCTRLn.KEEP) to '1'. When enabled, the corresponding channel output will be clamped to zero as long as the fault condition is present. The clamp will be released on the start of the first cycle after the fault condition is no longer present, see next Figure.

Figure 36-25. Waveform Generation with Fault Qualification and Keep Action



Restart Action

This is enabled by writing the Fault n Restart bit in Recoverable Fault n Configuration register (FCTRLn.RESTART) to '1'. When enabled, the timer/counter will be restarted as soon as the corresponding fault condition is present. The ongoing cycle is stopped and the timer/counter starts a new cycle, see *Waveform Generation in RAMP1 mode with Restart Action* figure. In Ramp 1 mode, when the new cycle starts, the compare outputs will be clamped to inactive level as long as the fault condition is present.

Note: For RAMP2 operation, when a new timer/counter cycle starts the cycle index will change automatically, see *Waveform Generation in RAMP2 mode with Restart Action* figure. Fault A and Fault B are qualified only during the cycle A and cycle B respectively: Fault A is disabled during cycle B, and Fault B is disabled during cycle A.

Figure 36-26. Waveform Generation in RAMP1 mode with Restart Action

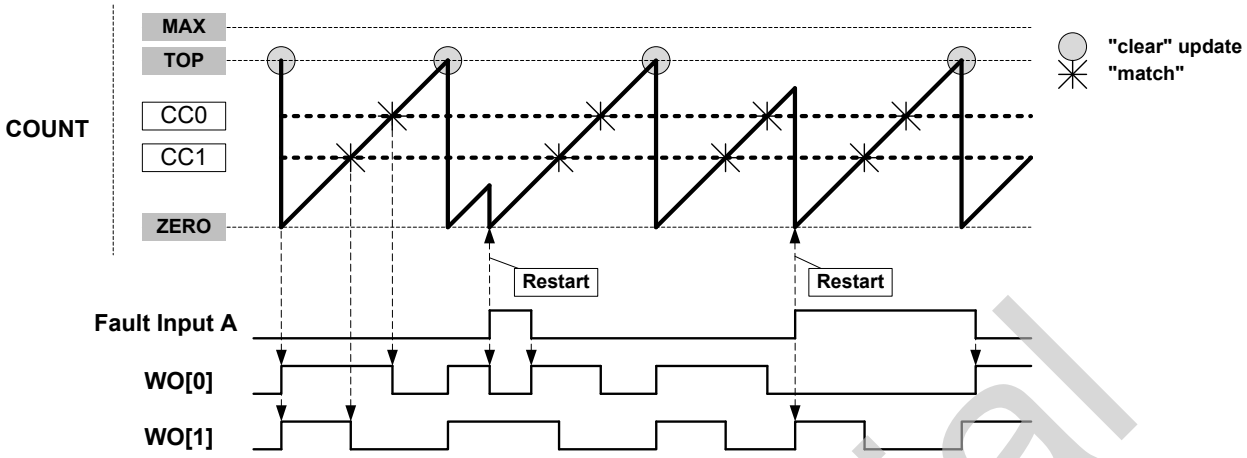
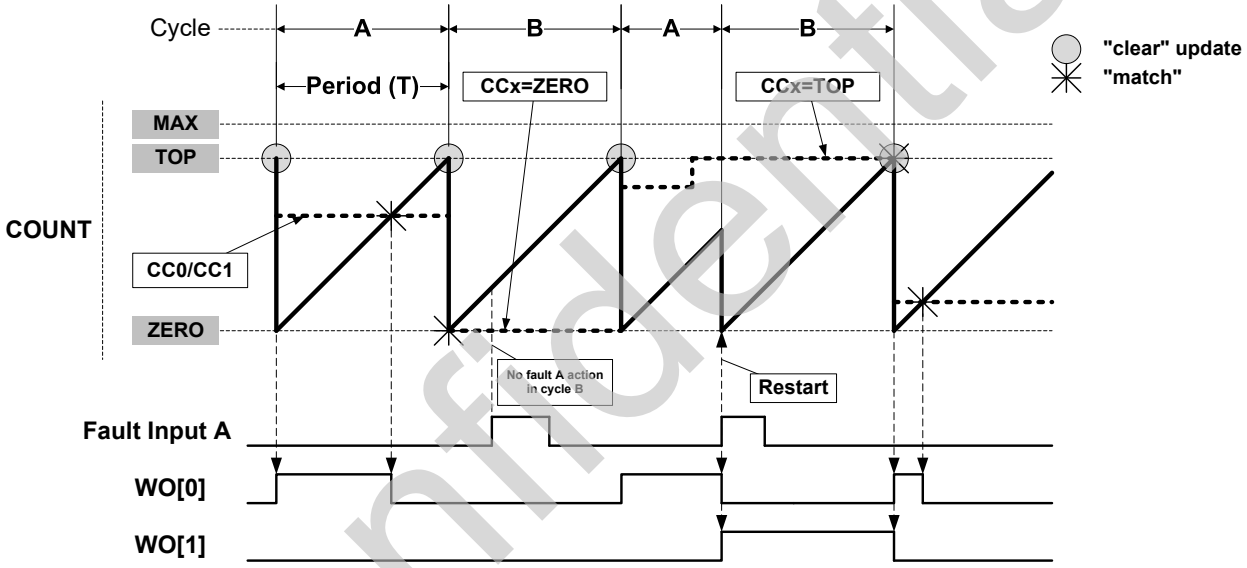


Figure 36-27. Waveform Generation in RAMP2 mode with Restart Action



Capture Action

Several capture actions can be selected by writing the Fault n Capture Action bits in the Fault n Control register (FCTRLn.CAPTURE). When one of the capture operations is selected, the counter value is captured when the fault occurs. These capture operations are available:

- CAPT - the equivalent to a standard capture operation, see *Capture Operations* from Related Links.
- CAPTMIN - gets the minimum time stamped value: on each new local minimum captured value, an event or interrupt is issued.
- CAPTMAX - gets the maximum time stamped value: on each new local maximum captured value, an event or interrupt (IT) is issued, see *Capture Action "CAPTMAX"* figure.
- LOCMIN - notifies by event or interrupt when a local minimum captured value is detected.
- LOCMAX - notifies by event or interrupt when a local maximum captured value is detected.
- DERIV0 - notifies by event or interrupt when a local extreme captured value is detected, see *Capture Action "DERIV0"* figure.

CCx Content:

In CAPTMIN and CAPTMAX operations, CCx keeps the respective extremum captured values, see *Capture Action "CAPTMAX"* figure. In LOCMIN, LOCMAX or DERIV0 operation, CCx follows the counter value at fault time, see *Capture Action "DERIV0"* figure.

Before enabling CAPTMIN or CAPTMAX mode of capture, the user must initialize the corresponding CCx register value to a value different from zero (for CAPTMIN) or top (for CAPTMAX). If the CCx register initial value is zero (for CAPTMIN) or top (for CAPTMAX), no captures will be performed using the corresponding channel.

MCx Behaviour:

In LOCMIN and LOCMAX operation, capture is performed on each capture event. The MCx interrupt flag is set only when the captured value is above or equal (for LOCMIN) or below or equal (for LOCMAX) to the previous captured value. So interrupt flag is set when a new relative local Minimum (for CAPTMIN) or Maximum (for CAPTMAX) value has been detected. DERIV0 is equivalent to an OR function of (LOCMIN, LOCMAX).

In CAPT operation, capture is performed on each capture event. The MCx interrupt flag is set on each new capture.

In CAPTMIN and CAPTMAX operation, capture is performed only when on capture event time, the counter value is lower (for CAPTMIN) or higher (for CAPMAX) than the last captured value. The MCx interrupt flag is set only when on capture event time, the counter value is higher or equal (for CAPTMIN) or lower or equal (for CAPTMAX) to the value captured on the previous event. So interrupt flag is set when a new absolute local Minimum (for CAPTMIN) or Maximum (for CAPTMAX) value has been detected.

Interrupt Generation

In CAPT mode, an interrupt is generated on each filtered Fault n and each dedicated CCx channel capture counter value. In other modes, an interrupt is only generated on an extreme captured value.

Figure 36-28. Capture Action "CAPTMAX"

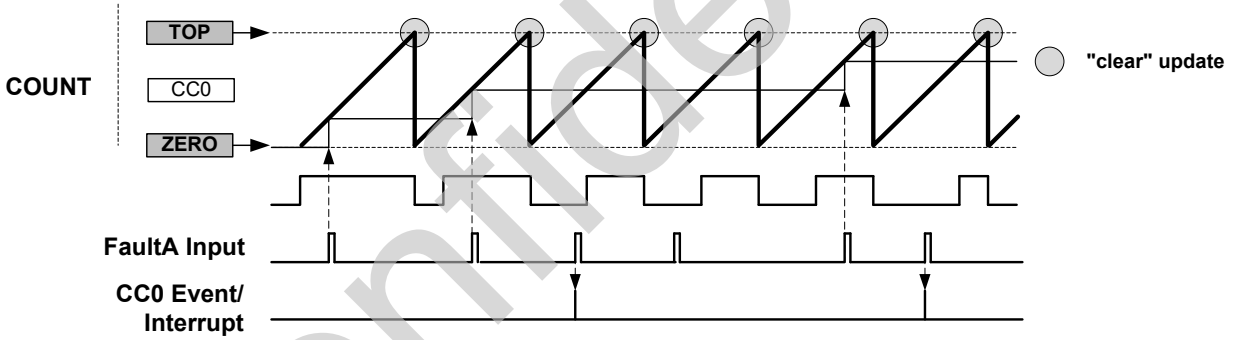
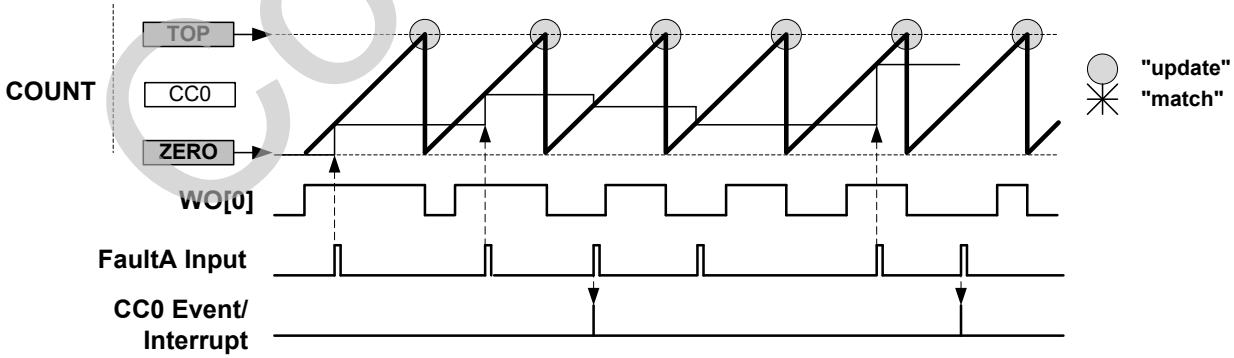


Figure 36-29. Capture Action "DERIV0"



Hardware Halt Action

This is configured by writing 0x1 to the Fault n Halt mode bits in the Recoverable Fault n Configuration register (FCTRLn.HALT). When enabled, the timer/counter is halted and the cycle is extended as long as the corresponding fault is present.

The next figure ('Waveform Generation with Halt and Restart Actions') shows an example where both restart action and hardware halt action are enabled for Fault A. The compare channel 0 output

is clamped to inactive level as long as the timer/counter is halted. The timer/counter resumes the counting operation as soon as the fault condition is no longer present. As the restart action is enabled in this example, the timer/counter is restarted after the fault condition is no longer present.

The figure after that ('Waveform Generation with Fault Qualification, and Halt) shows a similar example, but with additionally enabled fault qualification. Here, counting is resumed after the fault condition is no longer present.

Note that in RAMP2 and RAMP2A operations, when a new timer/counter cycle starts, the cycle index will automatically change.

Figure 36-30. Waveform Generation with Halt

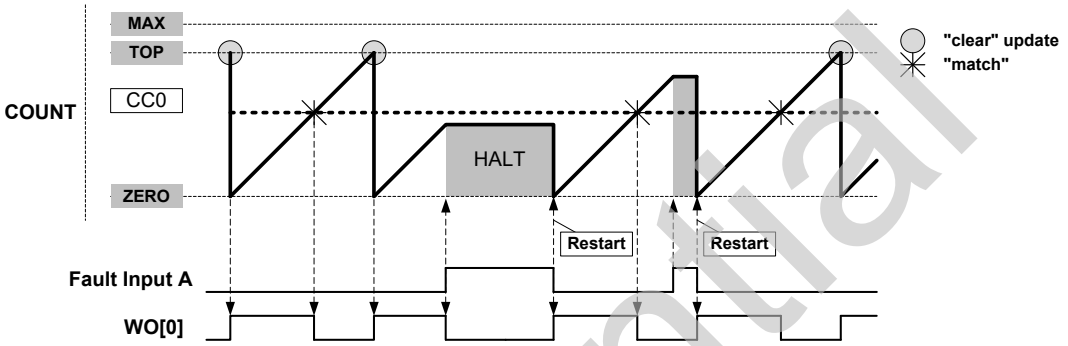
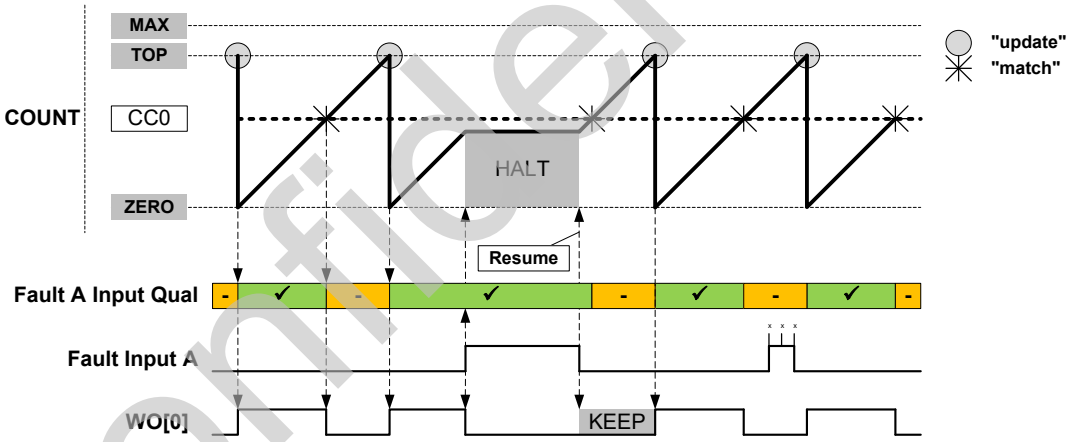


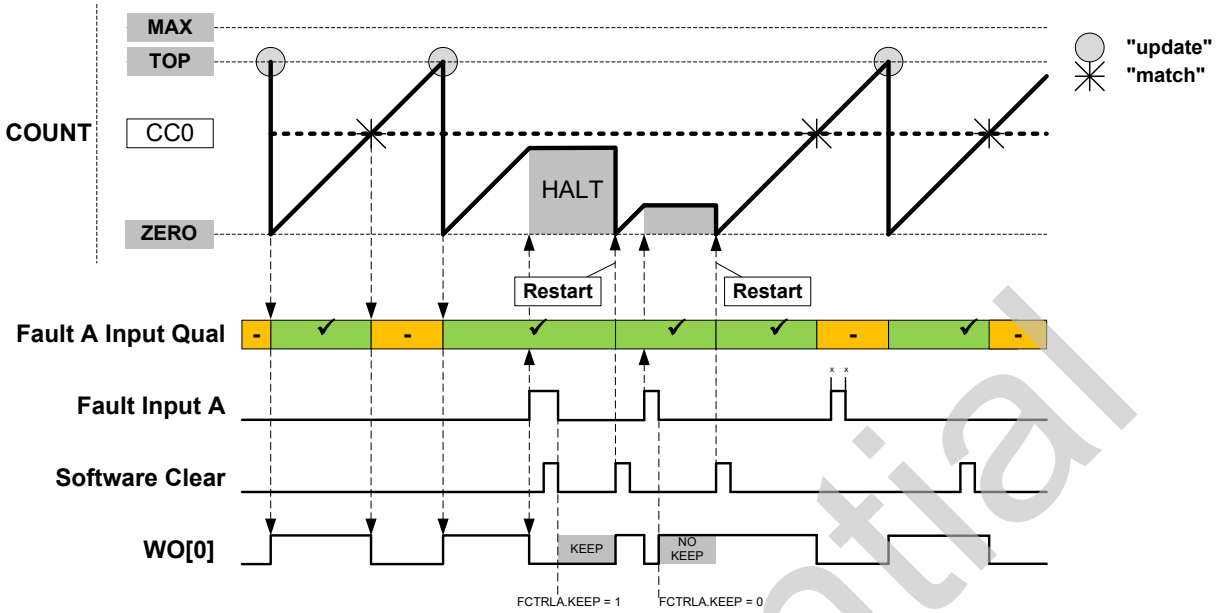
Figure 36-31. Waveform Generation with Fault Qualification and Halt



Software Halt Action

This is configured by writing 0x2 to the Fault n Halt mode bits in the Recoverable Fault n configuration register (FCTRLn.HALT). Software halt action is similar to hardware halt action, but in order to restart the timer/counter, the corresponding fault condition must not be present anymore, and the corresponding FAULT n bit in the STATUS register must be cleared by software.

Figure 36-32. Waveform Generation with Software Halt, Fault Qualification, Keep and Restart Actions



Related Links

36.6.2.7. Capture Operations

36.6.3.6 Non-Recoverable Faults

The non-recoverable fault action will force all the compare outputs to a pre-defined level programmed into the Driver Control register (DRVCTRL.NRE and DRVCTRL.NRV). The non-recoverable fault input (EV0 and EV1) actions are enabled in Event Control register (EVCTRL.EVACT0 and EVCTRL.EVACT1).

To avoid false fault detection on external events (e.g. a glitch on an I/O port) a digital filter can be enabled using Non-Recoverable Fault Input x Filter Value bits in the Driver Control register (DRVCTRL.FILTERVALn). Therefore, the event detection is synchronous, and event action is delayed by the selected digital filter value clock cycles.

When the Fault Detection on Debug Break Detection bit in Debug Control register (DGBCTRL.FDDBD) is written to '1', a non-recoverable Debug Faults State and an interrupt (DFS) is generated when the system goes in debug operation.

In RAMP2, RAMP2A, or DSBOTH operation, when the Lock Update bit in the Control B register is set by writing CTRLBSET.LUPD=1 and the ramp index or counter direction changes, a non-recoverable Update Fault State and the respective interrupt (UFS) are generated.

36.6.3.7 Time-Stamp Capture on Events or I/Os

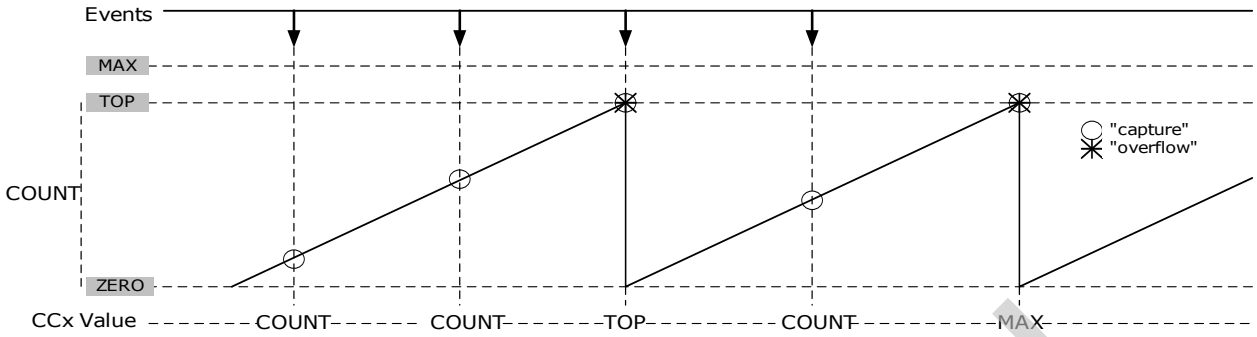
This feature is enabled when the Capture Time Stamp (STAMP) Event Action in Event Control register (EVCTRL.EVACT) is selected. The counter TOP value must be smaller than MAX.

When a capture event from the Event System or the I/O pin is detected, the COUNT value is copied into the corresponding Channel x Compare/Capture Value (CCx) register. In case of an overflow, the MAX value is copied into the corresponding CCx register.

When a valid captured value is present in the capture channel register, the corresponding Capture Channel x Interrupt Flag (INTFLAG.MCx) is set.

The timer/counter can detect capture overflow of the input capture channels: When a new capture event is detected while the Capture Channel interrupt flag (INTFLAG.MCx) is still set, the new time-stamp will not be stored and INTFLAG.ERR will be set.

Figure 36-33. Time Stamp



36.6.3.8 Waveform Extension

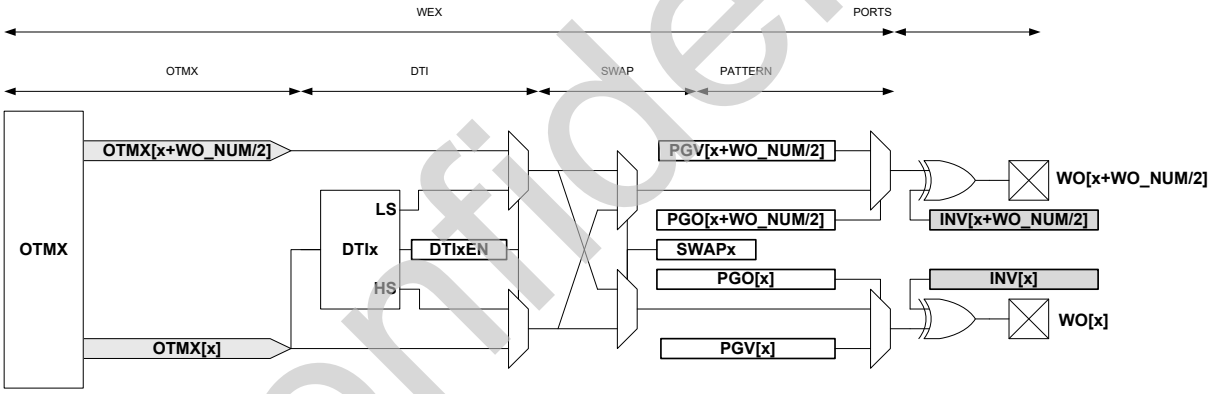
Waveform Extension Stage Details displays the schematic diagram of actions of the four optional units that follow the recoverable fault stage on a port pin pair: Output Matrix (OTMX), Dead-Time Insertion (DTI), SWAP and Pattern Generation. The DTI and SWAP units can be seen as a four port pair slices:

- Slice 0 DTI0 / SWAP0 acting on port pins (WO[0], WO[WO_NUM/2 +0])
- Slice 1 DTI1 / SWAP1 acting on port pins (WO[1], WO[WO_NUM/2 +1])

And generally:

- Slice n DTIx / SWAPx acting on port pins (WO[x], WO[WO_NUM/2 +x])

Figure 36-34. Waveform Extension Stage Details



The **output matrix (OTMX)** unit distributes compare channels, according to the selectable configurations in the following table.

Table 36-6. Output Matrix Channel Pin Routing Configuration

WEXCTRL.OTMX	OTMX[5]	OTMX[4]	OTMX[3]	OTMX[2]	OTMX[1]	OTMX[0]
0x0	CC5	CC4	CC3	CC2	CC1	CC0
0x1	CC2	CC1	CC0	CC2	CC1	CC0
0x2	CC0	CC0	CC0	CC0	CC0	CC0
0x3	CC1	CC1	CC1	CC1	CC1	CC0

- Configuration 0x0 is the default configuration. The channel location is the default one and channels are distributed on outputs modulo the number of channels. Channel 0 is routed to the Output matrix output OTMX[0], and Channel 1 to OTMX[1]. If there are more outputs than channels, then channel 0 is duplicated to the Output matrix output OTMX[CC_NUM], channel 1 to OTMX[CC_NUM+1] and so on.
- Configuration 0x1 distributes the channels on output modulo half the number of channels. This assigns twice the number of output locations to the lower channels than the default configuration. This can be used, for example, to control the four transistors of a full bridge using only two compare channels.

Using pattern generation, some of these four outputs can be overwritten by a constant level, enabling flexible drive of a full bridge in all quadrant configurations.

- Configuration 0x2 distributes compare channel 0 (CC0) to all port pins. With pattern generation, this configuration can control a stepper motor.
- Configuration 0x3 distributes the compare channel CC0 to the first output, and the channel CC1 to all other outputs. Together with pattern generation and the fault extension, this configuration can control up to seven LED strings, with a boost stage.

The table below is an example showing four compare channels on four outputs.

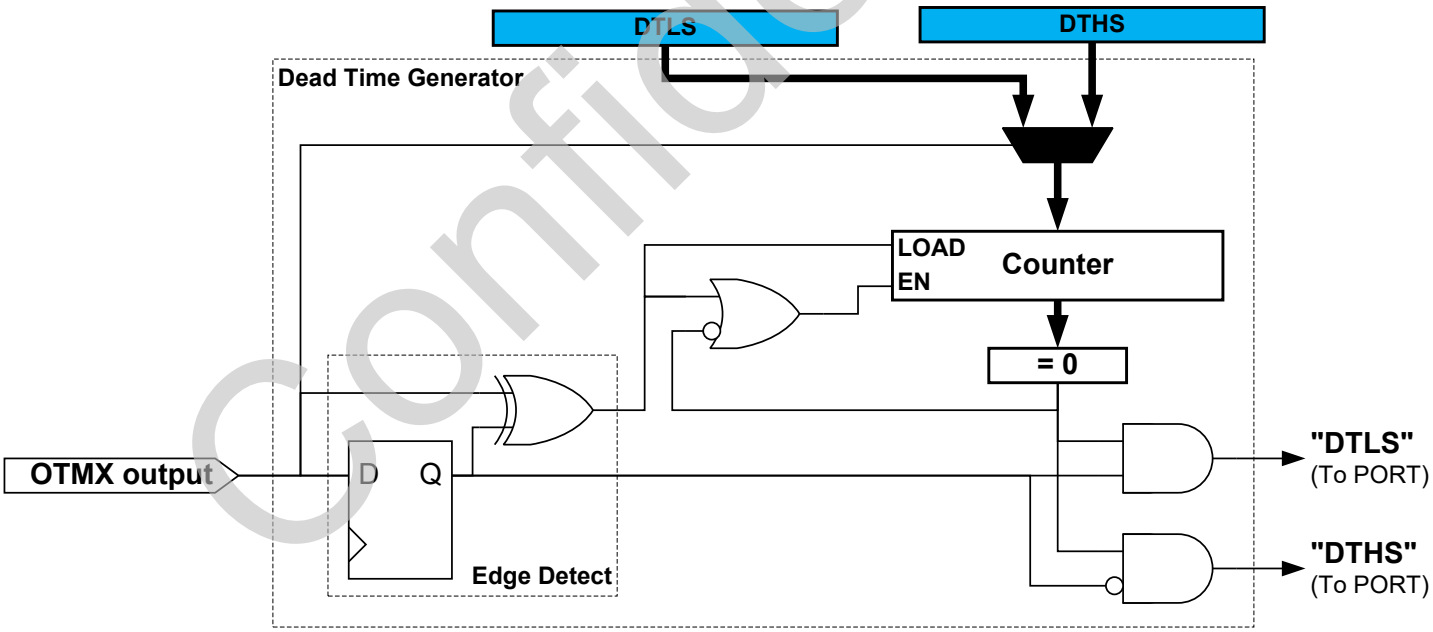
Table 36-7. Four Compare Channels on Four Outputs

WEXCTRL.OTMX	OTMX[3]	OTMX[2]	OTMX[1]	OTMX[0]
0x0	CC3	CC2	CC1	CC0
0x1	CC1	CC0	CC1	CC0
0x2	CC0	CC0	CC0	CC0
0x3	CC1	CC1	CC1	CC0

The dead-time insertion (DTI) unit generates OFF time with the non-inverted low side (LS) and inverted high side (HS) of the wave generator output forced at low level. This OFF time is called dead time. Dead-time insertion ensures that the LS and HS will never switch simultaneously.

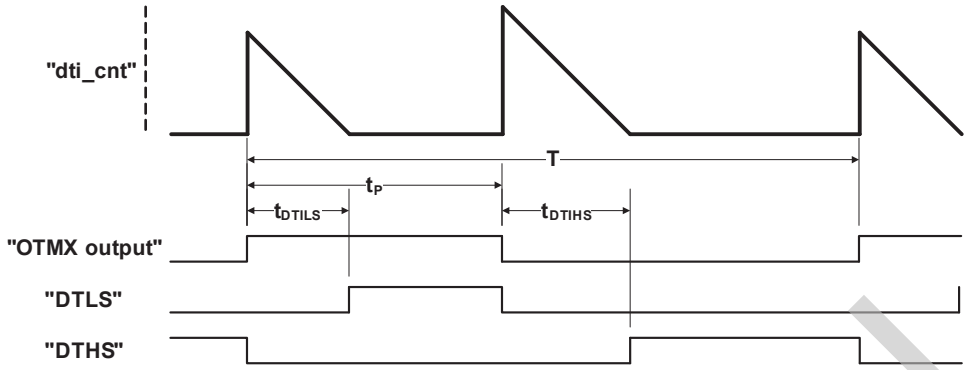
The DTI stage consists of four equal dead-time insertion generators; one for each of the first four compare channels. The following figure shows the block diagram of one DTI generator. The four channels have a common register which controls the dead time, which is independent of high side and low side setting.

Figure 36-35. Dead-Time Generator Block Diagram



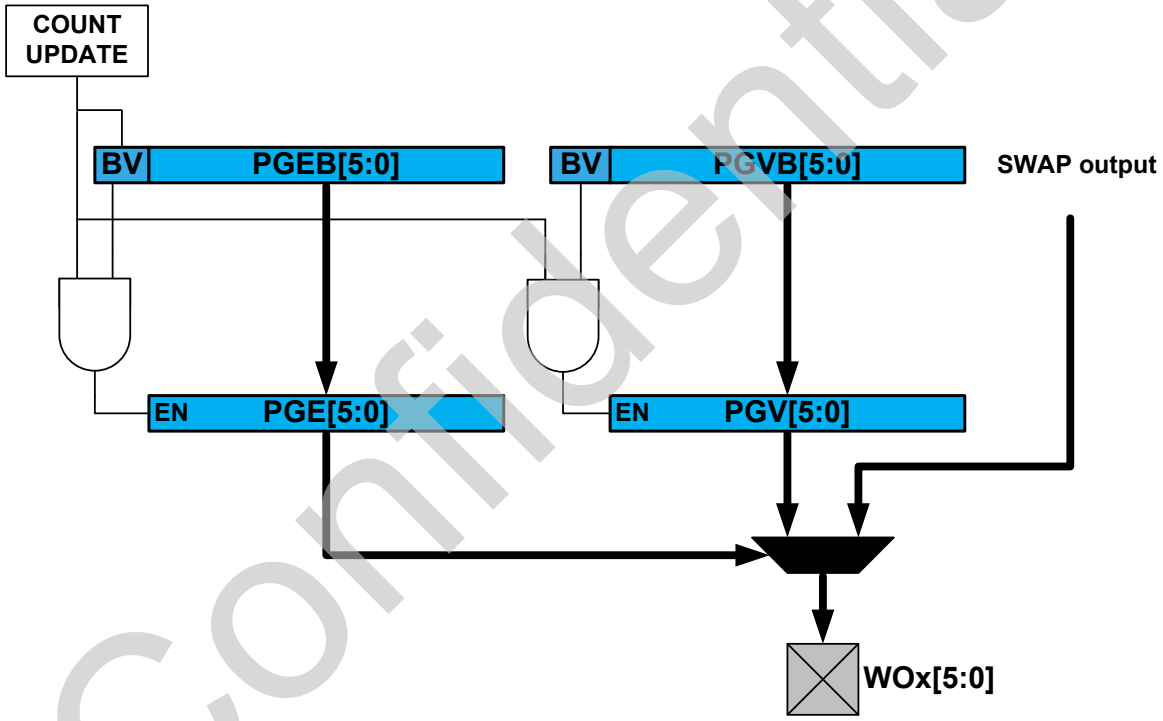
As shown in the following figure, the 8-bit dead-time counter is decremented by one for each peripheral clock cycle until it reaches zero. A non-zero counter value will force both the low side and high side outputs into their OFF state. When the output matrix (OTMX) output changes, the dead-time counter is reloaded according to the edge of the input. When the output changes from low to high (positive edge) it initiates a counter reload of the DTLS register. When the output changes from high to low (negative edge) it reloads the DTHS register.

Figure 36-36. Dead-Time Generator Timing Diagram



The pattern generator unit produces a synchronized bit pattern across the port pins it is connected to. The pattern generation features are primarily intended for handling the commutation sequence in brushless DC motors (BLDC), stepper motors, and full bridge control. For more information, refer to the following figure.

Figure 36-37. Pattern Generator Block Diagram



As with other double-buffered timer/counter registers, the register update is synchronized to the UPDATE condition set by the timer/counter waveform generation operation. If synchronization is not required by the application, the software can simply access directly the PATT.PGE, PATT.PGV bits registers.

36.6.4 Host/Client Operation

Two TCC instances sharing the same GCLK_TCC clock, can be linked to provide more synchronized CC channels. The operation is enabled by setting the Host Synchronization bit in Control A register (CTRLA.MSYNC) in the Client instance. When the bit is set, the Client TCC instance will synchronize the CC channels to the Host counter.

36.6.5 DMA, Interrupts, and Events

Table 36-8. Module Requests for TCC

Condition	Interrupt request	Event output	Event input	DMA request	DMA request is cleared
Overflow / Underflow	Yes	Yes		Yes(1)	On DMA acknowledge
Channel Compare Match or Capture	Yes	Yes	Yes(2)	Yes(3)	For circular buffering: on DMA acknowledge For capture channel: when CCx register is read
Retrigger	Yes	Yes			
Count	Yes	Yes			
Capture Overflow Error	Yes				
Debug Fault State	Yes				
Recoverable Faults	Yes				
Non-Recoverable Faults	Yes				
TCCx Event 0 input			Yes(4)		
TCCx Event 1 input			Yes(5)		

Notes:

1. DMA request set on Overflow, Underflow or Re-trigger conditions.
2. Can perform capture or generate recoverable fault on an event input.
3. In Capture or Circular modes.
4. On event input, either action can be executed:
   - re-trigger counter
   - control counter direction
   - stop the counter
   - decrement the counter
   - perform period and pulse width capture
   - generate non-recoverable fault
5. On event input, either action can be executed:
   - re-trigger counter
   - increment or decrement counter depending on direction
   - start the counter
   - increment or decrement counter based on direction
   - increment counter regardless of direction
   - generate non-recoverable fault

36.6.5.1 DMA Operation

The TCC can generate the following DMA requests:

Counter overflow (OVF) If the One-shot Trigger mode in the control A register (CTRLA.DMAOS) is written to '0', the The TCC generates a DMA request on each cycle when an update condition (Overflow, Underflow or Re-trigger) is detected.
When an update condition (Overflow, Underflow or Re-trigger) is detected while CTRLA.DMAOS=1, the TCC generates a DMA trigger on the cycle following the DMA One-Shot Command written to the Control B register (CTRLBSET.CMD=DMAOS).

	In both cases, the request is cleared by hardware on DMA acknowledge.
Channel Match (MCx)	A DMA request is set only on a compare match if CTRLA.DMAOS=0. The request is cleared by hardware on DMA acknowledge. When CTRLA.DMAOS=1, the DMA requests are not generated.
Channel Capture (MCx)	For a capture channel, the request is set when valid data is present in the CCx register, and cleared once the CCx register is read. In this operation mode, the CTRLA.DMAOS bit value is ignored.

DMA Operation with Circular Buffer

When circular buffer operation is enabled, the Buffer registers must be written in a correct order and synchronized to the update times of the timer. The DMA triggers of the TCC provide a way to ensure a safe and correct update of circular buffers.

Note: Circular buffer are intended to be used with RAMP2, RAMP2A and DSBOTH operation only.

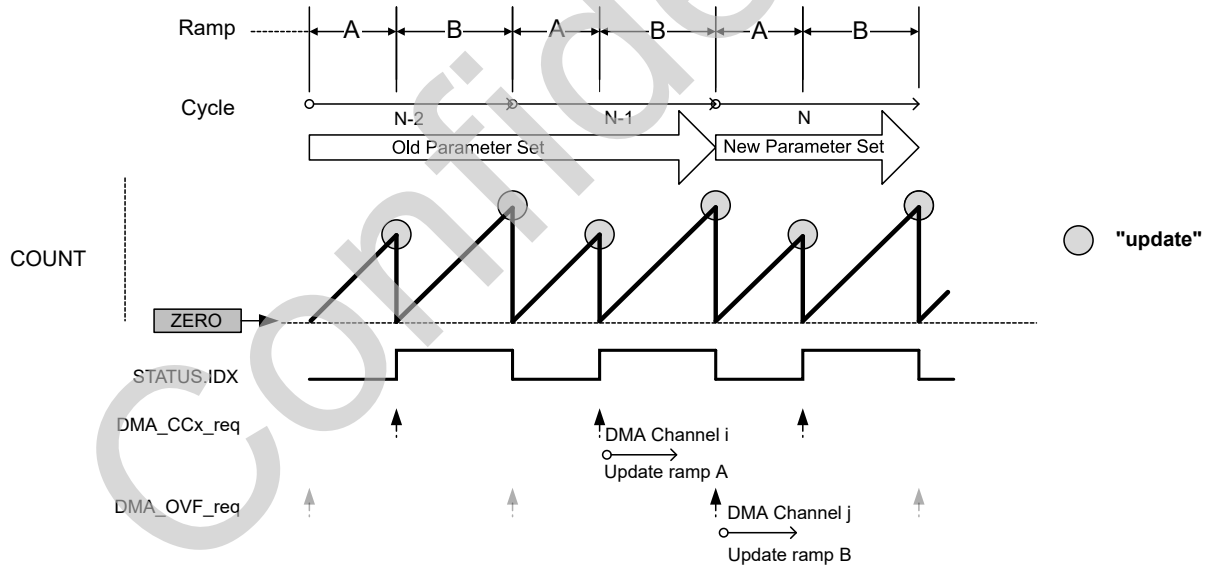
DMA Operation with Circular Buffer in RAMP2 and RAMP2A Mode

When a CCx channel is selected as a circular buffer, the related DMA request is not set on a compare match detection, but on start of ramp B.

If at least one circular buffer is enabled, the DMA overflow request is conditioned to the start of ramp A with an effective DMA transfer on previous ramp B (DMA acknowledge).

The update of all circular buffer values for ramp A can be done through a DMA channel triggered on a MC trigger. The update of all circular buffer values for ramp B, can be done through a second DMA channel triggered by the overflow DMA request.

Figure 36-38. DMA Triggers in RAMP and RAMP2 Operation Mode and Circular Buffer Enabled



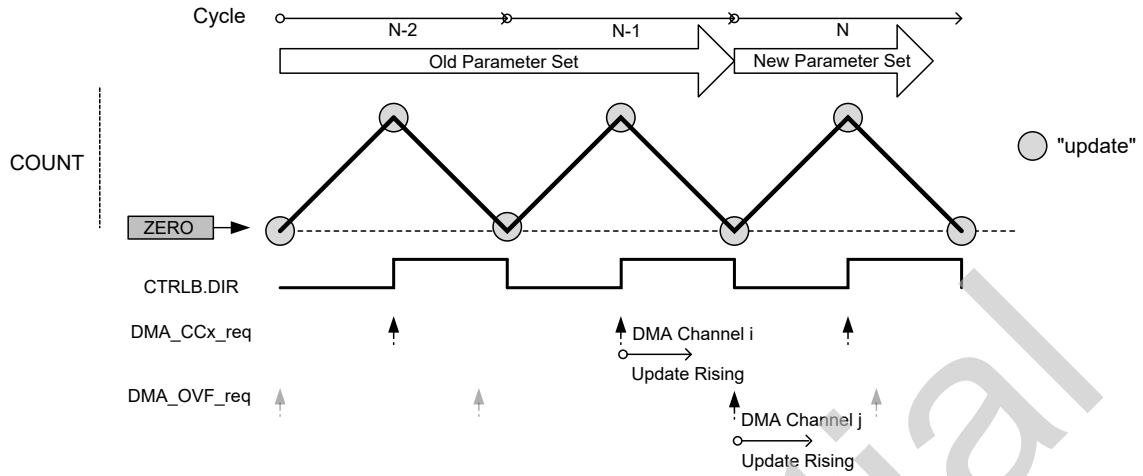
DMA Operation with Circular Buffer in DSBOTH Mode

When a CC channel is selected as a circular buffer, the related DMA request is not set on a compare match detection, but on start of down-counting phase.

If at least one circular buffer is enabled, the DMA overflow request is conditioned to the start of up-counting phase with an effective DMA transfer on previous down-counting phase (DMA acknowledge).

When up-counting, all circular buffer values can be updated through a DMA channel triggered by MC trigger. When down-counting, all circular buffer values can be updated through a second DMA channel, triggered by the OVF DMA request.

Figure 36-39. DMA Triggers in DSBOTH Operation Mode and Circular Buffer Enabled



36.6.5.2 Interrupts

The TCC has the following interrupt sources:

- Overflow/Underflow (OVF)
- Retrigger (TRG)
- Count (CNT) – Refer also to the description of EVCTRL.CNTSEL
- Capture Overflow Error (ERR)
- Non-Recoverable Update Fault (UFS)
- Debug Fault State (DFS)
- Recoverable Faults (FAULTn)
- Non-recoverable Faults (FAULTx)
- Compare Match or Capture Channels (MCx)

These interrupts are asynchronous wake-up sources.

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 '1' to the corresponding bit in the Interrupt Enable Set (INTENSET) register, and disabled by writing a '1' to the corresponding bit in the Interrupt Enable Clear (INTENCLR) register. The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the Interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the Interrupt flag is cleared, the interrupt is disabled or the TCC is reset. See *INTFLAG* from Related Links for details on how to clear Interrupt flags. The TCC has one common interrupt request line for all the interrupt sources. The user must read the INTFLAG register to determine which Interrupt condition is present.

Interrupts must be globally enabled for interrupt requests to be generated. See *Nested Vector Interrupt Controller (NVIC)* from Related Links.

Related Links

8.2. Nested Vector Interrupt Controller (NVIC)

36.8.12. INTFLAG

36.6.5.3 Events

The TCC can generate the following output events:

- Overflow/Underflow (OVF)
- Trigger (TRG)
- Counter (CNT) (For further details, refer to the EVCTRL.CNTSEL description.)

- Compare Match or Capture on compare/capture channels: MCx

Writing a '1' ('0') to an Event Output bit in the Event Control Register (EVCTRL.xxEO) enables (disables) the corresponding output event. See *Event System (EVSYS)* from Related Links.

The TCC can take the following actions on a channel input event (MCx):

- Capture event
- Generate a recoverable or non-recoverable fault

The TCC can take the following actions on counter Event 1 (TCCx EV1):

- Counter re-trigger
- Counter direction control
- Stop the counter
- Decrement the counter on event
- Period and pulse width capture
- Non-recoverable fault

The TCC can take the following actions on counter Event 0 (TCCx EV0):

- Counter re-trigger
- Count on event (increment or decrement, depending on counter direction)
- Counter start – Start counting on the event rising edge. Further events will not restart the counter; the counter will keep counting using prescaled GCLK_TCCx, until it reaches TOP or ZERO, depending on the direction.
- Counter increment on event. This will increment the counter, irrespective of the counter direction.
- Count during active state of an asynchronous event (increment or decrement, depending on counter direction). In this case, the counter will be incremented or decremented on each cycle of the prescaled clock, as long as the event is active.
- Non-recoverable fault

The counter Event Actions are available in the Event Control registers (EVCTRL.EVACT0 and EVCTRL.EVACT1). See *EVCTRL* from Related Links.

Writing a '1' ('0') to an Event Input bit in the Event Control register (EVCTRL.MCEIx or EVCTRL.TCEIx) enables (disables) the corresponding action on input event.

Note: When several events are connected to the TCC, the enabled action will apply for each of the incoming events. See *Event System (EVSYS)* from Related Links for details on how to configure the Event System.

Related Links

26. Event System (EVSYS)
36.8.9. EVCTRL

36.6.6 Sleep Mode Operation

The TCC can be configured to operate in any sleep mode (Standby Sleep, Idle). To be able to run in standby sleep mode, the RUNSTDBY bit in the Control A register (CTRLA.RUNSTDBY) must be '1'. This peripheral can wake up the device from any sleep mode using interrupts or perform actions through the Event System.

36.6.7 Synchronization

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

The following bits are synchronized when written:

- Software Reset and Enable bits in Control A register (CTRLA.SWRST and CTRLA.ENABLE)

The following registers are synchronized when written:

- Control B Clear and Control B Set registers (CTRLBCLR and CTRLBSET)
- Status register (STATUS)
- Pattern and Pattern Buffer registers (PATT and PATTBUF)
- Waveform register (WAVE)

- Count Value register (COUNT)
- Period Value and Period Buffer Value registers (PER and PERBUF)
- Compare/Capture Channel x and Channel x Compare/Capture Buffer Value registers (CCx and CCBUFx)

The following registers are synchronized when read:

- Control B Clear and Control B Set registers (CTRLBCLR and CTRLBSET)
- Count Value register (COUNT): synchronization is done on demand through READSYNC command (CTRLBSET.CMD)
- Pattern and Pattern Buffer registers (PATT and PATTBUF)
- Waveform register (WAVE)
- Period Value and Period Buffer Value registers (PER and PERBUF)
- Compare/Capture Channel x and Channel x Compare/Capture Buffer Value registers (CCx and CCBUFx)

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

Required read synchronization is denoted by the "Read-Synchronized" property in the register description.

36.7 Register Summary

See *TCCx (x = 0 to 2)* module in the *Product Memory Mapping Overview* from Related Links for base address based on the TCC instant used.

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x00	CTRLA	7:0		RESOLUTION[1:0]					ENABLE	SWRST
		15:8	MSYNC	ALOCK	PRESCYNC[1:0]		RUNSTDBY	PRESCALER[2:0]		
		23:16	DMAOS							
		31:24			CPTEN5	CPTEN4	CPTEN3	CPTEN2	CPTEN1	CPTEN0
0x04	CTRLBCLR	7:0	CMD[2:0]			IDXCMD[1:0]		ONESHOT	LUPD	DIR
0x05	CTRLBSET	7:0	CMD[2:0]			IDXCMD[1:0]		ONESHOT	LUPD	DIR
0x06 ... 0x07	Reserved									
0x08	SYNCBUSY	7:0	PER	WAVE	PATT	COUNT	STATUS	CTRLB	ENABLE	SWRST
		15:8			CC5	CC4	CC3	CC2	CC1	CC0
		23:16								
		31:24								
0x0C	FCTRLA	7:0	RESTART	BLANK[1:0]		QUAL	KEEP		SRC[1:0]	
		15:8	BLANKPRESC	CAPTURE[2:0]			CHSEL[1:0]		HALT[1:0]	
		23:16	BLANKVAL[7:0]							
		31:24					FILTERVAL[3:0]			
0x10	FCTRLB	7:0	RESTART	BLANK[1:0]		QUAL	KEEP		SRC[1:0]	
		15:8	BLANKPRESC	CAPTURE[2:0]			CHSEL[1:0]		HALT[1:0]	
		23:16	BLANKVAL[7:0]							
		31:24					FILTERVAL[3:0]			
0x14	WEXCTRL	7:0							OTMX[1:0]	
		15:8					DTIEN3	DTIEN2	DTIEN1	DTIEN0
		23:16	DTLS[7:0]							
		31:24	DTHS[7:0]							
0x18	DRVCTRL	7:0			NRE5	NRE4	NRE3	NRE2	NRE1	NRE0
		15:8			NRV5	NRV4	NRV3	NRV2	NRV1	NRV0
		23:16			INVEN5	INVEN4	INVEN3	INVEN2	INVEN1	INVEN0
		31:24	FILTERVAL1[3:0]				FILTERVAL0[3:0]			
0x1C ... 0x1D	Reserved									
0x1E	DBGCTRL	7:0						FDDBD		DBGRUN
0x1F	Reserved									
0x20	EVCTRL	7:0	CNTSEL[1:0]		EVACT1[2:0]			EVACT0[2:0]		
		15:8	TCEI1	TCEI0	TCINV1	TCINV0		CNTEO	TRGEO	OVFEO
		23:16			MCEI5	MCEI4	MCEI3	MCEI2	MCEI1	MCEI0
		31:24			MCEO5	MCEO4	MCEO3	MCEO2	MCEO1	MCEO0
0x24	INTENCLR	7:0					ERR	CNT	TRG	OVF
		15:8	FAULT1	FAULT0	FAULTB	FAULTA	DFS	UFS		
		23:16			MC5	MC4	MC3	MC2	MC1	MC0
		31:24								
0x28	INTENSET	7:0					ERR	CNT	TRG	OVF
		15:8	FAULT1	FAULT0	FAULTB	FAULTA	DFS	UFS		
		23:16			MC5	MC4	MC3	MC2	MC1	MC0
		31:24								
0x2C	INTFLAG	7:0					ERR	CNT	TRG	OVF
		15:8	FAULT1	FAULT0	FAULTB	FAULTA	DFS	UFS		
		23:16			MC5	MC4	MC3	MC2	MC1	MC0
		31:24								

.....continued

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x30	STATUS	7:0	PERBUFV		PATTBUFV	SLAVE	DFS	UFS	IDX	STOP
		15:8	FAULT1	FAULT0	FAULTB	FAULTA	FAULT1IN	FAULT0IN	FAULTBIN	FAULTAIN
		23:16			CCBUFV5	CCBUFV4	CCBUFV3	CCBUFV2	CCBUFV1	CCBUFV0
		31:24			CMP5	CMP4	CMP3	CMP2	CMP1	CMP0
0x34	COUNT	7:0	COUNT[7:0]							
		15:8	COUNT[15:8]							
		23:16	COUNT[23:16]							
		31:24								
0x38	PATT	7:0			PGE5	PGE4	PGE3	PGE2	PGE1	PGE0
		15:8			PGV5	PGV4	PGV3	PGV2	PGV1	PGV0
0x3A ... 0x3B	Reserved									
0x3C	WAVE	7:0	CIPEREN		RAMP[1:0]			WAVEGEN[2:0]		
		15:8					CICCEN3	CICCEN2	CICCEN1	CICCEN0
		23:16			POL5	POL4	POL3	POL2	POL1	POL0
		31:24					SWAP3	SWAP2	SWAP1	SWAP0
0x40	PER	7:0	PER[1:0]		DITHER[5:0]					
		15:8	PER[9:2]							
		23:16	PER[17:10]							
		31:24								
0x44	CCx0	7:0	CC[1:0]		DITHER[5:0]					
		15:8	CC[9:2]							
		23:16	CC[17:10]							
		31:24								
0x48	CCx1	7:0	CC[1:0]		DITHER[5:0]					
		15:8	CC[9:2]							
		23:16	CC[17:10]							
		31:24								
0x4C	CCx2	7:0	CC[1:0]		DITHER[5:0]					
		15:8	CC[9:2]							
		23:16	CC[17:10]							
		31:24								
0x50	CCx3	7:0	CC[1:0]		DITHER[5:0]					
		15:8	CC[9:2]							
		23:16	CC[17:10]							
		31:24								
0x54	CCx4	7:0	CC[1:0]		DITHER[5:0]					
		15:8	CC[9:2]							
		23:16	CC[17:10]							
		31:24								
0x58	CCx5	7:0	CC[1:0]		DITHER[5:0]					
		15:8	CC[9:2]							
		23:16	CC[17:10]							
		31:24								
0x5C ... 0x63	Reserved									
0x64	PATTBUF	7:0			PGEB5	PGEB4	PGEB3	PGEB2	PGEB1	PGEB0
		15:8			PGVB5	PGVB4	PGVB3	PGVB2	PGVB1	PGVB0
0x66 ... 0x6B	Reserved									
0x6C	PERBUF	7:0	PERBUF[1:0]		DITHERBUF[5:0]					
		15:8	PERBUF[9:2]							
		23:16	PERBUF[17:10]							
		31:24								

.....continued

Offset	Name	Bit Pos.	7	6	5	4	3	2	1	0
0x70	CCBUFx0	7:0	CCBUF[1:0]		DITHERBUF[5:0]					
		15:8	CCBUF[9:2]							
		23:16	CCBUF[17:10]							
		31:24								
0x74	CCBUFx1	7:0	CCBUF[1:0]		DITHERBUF[5:0]					
		15:8	CCBUF[9:2]							
		23:16	CCBUF[17:10]							
		31:24								
0x78	CCBUFx2	7:0	CCBUF[1:0]		DITHERBUF[5:0]					
		15:8	CCBUF[9:2]							
		23:16	CCBUF[17:10]							
		31:24								
0x7C	CCBUFx3	7:0	CCBUF[1:0]		DITHERBUF[5:0]					
		15:8	CCBUF[9:2]							
		23:16	CCBUF[17:10]							
		31:24								
0x80	CCBUFx4	7:0	CCBUF[1:0]		DITHERBUF[5:0]					
		15:8	CCBUF[9:2]							
		23:16	CCBUF[17:10]							
		31:24								
0x84	CCBUFx5	7:0	CCBUF[1:0]		DITHERBUF[5:0]					
		15:8	CCBUF[9:2]							
		23:16	CCBUF[17:10]							
		31:24								

Related Links

7. Product Memory Mapping Overview

36.8 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.

Some registers require synchronization when read and/or written. Synchronization is denoted by the "Read-Synchronized" and/or "Write-Synchronized" property in each individual register description.

Optional write protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write Protection" property in each individual register description.

Some registers are enable-protected, meaning they can only be written when the module is disabled. Enable protection is denoted by the "Enable-Protected" property in each individual register description.

36.8.1 Control A

Name: CTRLA
Offset: 0x00
Reset: 0x00000000
Property: PAC Write-Protection, Enable-Protected, Write-Synchronized (ENABLE, SWRST)

Bit	31	30	29	28	27	26	25	24
			CPTEN5	CPTEN4	CPTEN3	CPTEN2	CPTEN1	CPTEN0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

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

Bit	15	14	13	12	11	10	9	8
	MSYNC	ALOCK	PRESCYNC[1:0]		RUNSTDBY	PRESCALER[2:0]		
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	7	6	5	4	3	2	1	0
		RESOLUTION[1:0]					ENABLE	SWRST
Access		R/W	R/W				R/W	R/W
Reset		0	0				0	0

Bits 24, 25, 26, 27, 28, 29 – CPTEN Capture Channel x Enable
These bits are used to select the capture or compare operation on channel x.
Writing a '1' to CPTENx enables capture on channel x.
Writing a '0' to CPTENx disables capture on channel x.

Bit 23 – DMAOS DMA One-Shot Trigger Mode
This bit enables the DMA One-shot Trigger Mode.
Writing a '1' to this bit will generate a DMA trigger on TCC cycle following a TCC_CTRLBSET_CMD_DMAOS command.
Writing a '0' to this bit will generate DMA triggers on each TCC cycle.
This bit is not synchronized.

Bit 15 – MSYNC Host Synchronization (only for TCC client instance)
This bit must be set if the TCC counting operation must be synchronized on its Host TCC.
This bit is not synchronized.

Value	Description
0	The TCC controls its own counter.
1	The counter is controlled by its Host TCC.

Bit 14 – ALOCK Auto Lock
This bit is not synchronized.

Value	Description
0	The Lock Update bit in the Control B register (CTRLB.LUPD) is not affected by overflow/underflow, and re-trigger events
1	CTRLB.LUPD is set to '1' on each overflow/underflow or re-trigger event.

Bits 13:12 – PRESCYNC[1:0] Prescaler and Counter Synchronization
These bits select if on re-trigger event, the Counter is cleared or reloaded on either the next GCLK_TCCx clock, or on the next prescaled GCLK_TCCx clock. It is also possible to reset the prescaler on re-trigger event.

These bits are not synchronized.

Value	Name	Description	
		Counter Reloaded	Prescaler
0x0	GCLK	Reload or reset Counter on next GCLK	-
0x1	PRESC	Reload or reset Counter on next prescaler clock	-
0x2	RESYNC	Reload or reset Counter on next GCLK	Reset prescaler counter
0x3	Reserved		

Bit 11 – RUNSTDBY Run in Standby

This bit is used to keep the TCC running in Standby mode.
This bit is not synchronized.

Value	Description
0	The TCC is halted in standby.
1	The TCC continues to run in standby.

Bits 10:8 – PRESCALER[2:0] Prescaler

These bits select the Counter prescaler factor.
These bits are not synchronized.

Value	Name	Description
0x0	DIV1	Prescaler: GCLK_TCC
0x1	DIV2	Prescaler: GCLK_TCC/2
0x2	DIV4	Prescaler: GCLK_TCC/4
0x3	DIV8	Prescaler: GCLK_TCC/8
0x4	DIV16	Prescaler: GCLK_TCC/16
0x5	DIV64	Prescaler: GCLK_TCC/64
0x6	DIV256	Prescaler: GCLK_TCC/256
0x7	DIV1024	Prescaler: GCLK_TCC/1024

Bits 6:5 – RESOLUTION[1:0] Dithering Resolution

These bits increase the TCC resolution by enabling the dithering options.
These bits are not synchronized.

Table 36-9. Dithering

Value	Name	Description
0x0	NONE	The dithering is disabled.
0x1	DITH4	Dithering is done every 16 PWM frames. PER[3:0] and CCx[3:0] contain dithering pattern selection.
0x2	DITH5	Dithering is done every 32 PWM frames. PER[4:0] and CCx[4:0] contain dithering pattern selection.
0x3	DITH6	Dithering is done every 64 PWM frames. PER[5:0] and CCx[5:0] contain dithering pattern selection.

Bit 1 – ENABLE Enable

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

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

Bit 0 – SWRST Software Reset

Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets all registers in the TCC (except DBGCTRL) to their initial state, and the TCC will be disabled.

Writing a '1' to CTRLA.SWRST will always take precedence; all other writes in the same write-operation will be discarded.
Due to synchronization there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the reset is complete.

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

36.8.2 Control B Clear

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

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

Bit	7	6	5	4	3	2	1	0
	CMD[2:0]			IDXCMD[1:0]		ONESHOT	LUPD	DIR
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 7:5 – CMD[2:0] TCC Command
Writing a '0' to these bits has no effect.
Writing a '1' to any of these bits will clear the pending command.

Value	Name	Description
0x0	NONE	No action
0x1	RETRIGGER	Clear start, restart or retrigger
0x2	STOP	Force stop
0x3	UPDATE	Force update of double buffered registers
0x4	READSYNC	Force COUNT read synchronization
0x5	DMAOS	One-shot DMA trigger

Bits 4:3 – IDXCMD[1:0] Ramp Index Command
These bits can be used to force cycle A and cycle B changes in RAMP2 and RAMP2A operation. On timer/counter update condition, the command is executed, the IDX flag in STATUS register is updated and the IDXCMD command is cleared.
Writing zero to these bits has no effect.
Writing a '1' to any of these bits will clear the pending command.

Value	Name	Description
0x0	DISABLE	DISABLE Command disabled: IDX toggles between cycles A and B
0x1	SET	Set IDX: cycle B will be forced in the next cycle
0x2	CLEAR	Clear IDX: cycle A will be forced in next cycle
0x3	HOLD	Hold IDX: the next cycle will be the same as the current cycle.

Bit 2 – ONESHOT One-Shot
This bit controls one-shot operation of the TCC. When one-shot operation is enabled, the TCC will stop counting on the next overflow/underflow condition or on a stop command.
Writing a '0' to this bit has no effect
Writing a '1' to this bit will disable the one-shot operation.

Value	Description
0	The TCC will update the counter value on overflow/underflow condition and continue operation.
1	The TCC will stop counting on the next underflow/overflow condition.

Bit 1 – LUPD Lock Update
This bit controls the update operation of the TCC buffered registers.
When CTRLB.LUPD is cleared, the hardware UPDATE registers with value from their buffered registers is enabled.
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 enable the registers updates on hardware UPDATE condition.

Value	Description
0	The CCBx, PERB, PGVB, and PGEB buffer registers values *are* copied into the corresponding CCx, PER, PGV, and PGE registers on hardware update condition.

Value	Description
1	The CCBx, PERB, PGVB, and PGEB buffer registers values are *not* copied into the corresponding CCx, PER, PGV, and PGE registers on hardware update condition.

Bit 0 – DIR Counter Direction

This bit is used to change the direction of the counter.
Writing a '0' to this bit has no effect
Writing a '1' to this bit will clear the bit and make the counter count up.

Value	Description
0	The timer/counter is counting up (incrementing).
1	The timer/counter is counting down (decrementing).

36.8.3 Control B Set

Name: CTRLBSET
Offset: 0x05
Reset: 0x00
Property: PAC Write-Protection, Write-Synchronized, Read-Synchronized

This register allows the user to change this register without doing a read-modify-write operation. Changes in this register will also be reflected in the Control B Clear (CTRLBCLR) register.

Bit	7	6	5	4	3	2	1	0
	CMD[2:0]			IDXCMD[1:0]		ONESHOT	LUPD	DIR
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 7:5 – CMD[2:0] TCC Command
These bits can be used for software control of re-triggering and stop commands of the TCC. When a command has been executed, the CMD bit field will be read back as zero. The commands are executed on the next prescaled GCLK_TCCx clock cycle.
Writing a '0' to this bit has no effect
Writing a valid value to this bit group, as shown in the following table, will set the associated command.

Value	Name	Description
0x0	NONE	No action
0x1	RETRIGGER	Force start, restart or retrigger
0x2	STOP	Force stop
0x3	UPDATE	Force update of double buffered registers
0x4	READSYNC	Force a read synchronization of COUNT
0x5	DMAOS	One-shot DMA trigger

Bits 4:3 – IDXCMD[1:0] Ramp Index Command
These bits can be used to force cycle A and cycle B changes in RAMP2 and RAMP2A operation. On timer/counter update condition, the command is executed, the IDX flag in STATUS register is updated and the IDXCMD command is cleared.
Writing a '0' to this bit has no effect
Writing a valid value to these bits will set a command.

Value	Name	Description
0x0	DISABLE	Command disabled: IDX toggles between cycles A and B
0x1	SET	Set IDX: cycle B will be forced in the next cycle
0x2	CLEAR	Clear IDX: cycle A will be forced in next cycle
0x3	HOLD	Hold IDX: the next cycle will be the same as the current cycle.

Bit 2 – ONESHOT One-Shot
This bit controls one-shot operation of the TCC. When in one-shot operation, the TCC will stop counting on the next overflow/underflow condition or a stop command.
Writing a '0' to this bit has no effect
Writing a '1' to this bit will enable the one-shot operation.

Value	Description
0	The TCC will count continuously.
1	The TCC will stop counting on the next underflow/overflow condition.

Bit 1 – LUPD Lock Update
This bit controls the update operation of the TCC buffered registers.
When CTRLB.LUPD is set, the hardware UPDATE registers with value from their buffered registers is disabled. Disabling 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 disable the registers updates on hardware UPDATE condition.

Value	Description
0	The CCBx, PERB, PGVB, and PGEB buffer registers values *are* copied into the corresponding CCx, PER, PGV, and PGE registers on hardware update condition.
1	The CCBx, PERB, PGVB, and PGEB buffer registers values are *not* copied into CCx, PER, PGV, and PGE registers on hardware update condition.

Bit 0 – DIR Counter Direction

This bit is used to change the direction of the counter.
Writing a '0' to this bit has no effect
Writing a '1' to this bit will set the bit and make the counter count down.

Value	Description
0	The timer/counter is counting up (incrementing).
1	The timer/counter is counting down (decrementing).

36.8.4 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
			CC5	CC4	CC3	CC2	CC1	CC0
Access			R	R	R	R	R	R
Reset			0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	PER	WAVE	PATT	COUNT	STATUS	CTRLB	ENABLE	SWRST
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 8, 9, 10, 11, 12, 13 – CC Compare/Capture Channel x Synchronization Busy
This bit is cleared when the synchronization of Compare/Capture Channel x register between the clock domains is complete.
This bit is set when the synchronization of Compare/Capture Channel x register between clock domains is started.
CCx bit is available only for existing Compare/Capture Channels. For details on CC channels number, refer to each TCC feature list.
This bit is set when the synchronization of CCx register between clock domains is started.

Bit 7 – PER PER Synchronization Busy
This bit is cleared when the synchronization of PER register between the clock domains is complete.
This bit is set when the synchronization of PER register between clock domains is started.

Bit 6 – WAVE WAVE Synchronization Busy
This bit is cleared when the synchronization of WAVE register between the clock domains is complete.
This bit is set when the synchronization of WAVE register between clock domains is started.

Bit 5 – PATT PATT Synchronization Busy
This bit is cleared when the synchronization of PATTERN register between the clock domains is complete.
This bit is set when the synchronization of PATTERN register between clock domains is started.

Bit 4 – COUNT COUNT Synchronization Busy
This bit is cleared when the synchronization of COUNT register between the clock domains is complete.
This bit is set when the synchronization of COUNT register between clock domains is started.

Bit 3 – STATUS STATUS Synchronization Busy
This bit is cleared when the synchronization of STATUS register between the clock domains is complete.
This bit is set when the synchronization of STATUS register between clock domains is started.

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

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

Bit 0 – SWRST SWRST Synchronization Busy
This bit is cleared when the synchronization of SWRST bit between the clock domains is complete.
This bit is set when the synchronization of SWRST bit between clock domains is started.

36.8.5 Fault Control A and B

Name: FCTRLn
Offset: 0x0C + n*0x04 [n=0..1]
Reset: 0x00000000
Property: PAC Write-Protection, Enable-Protected

Bit	31	30	29	28	27	26	25	24
					FILTERVAL[3:0]			
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	23	22	21	20	19	18	17	16
	BLANKVAL[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
	BLANKPRESC	CAPTURE[2:0]			CHSEL[1:0]		HALT[1:0]	
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	7	6	5	4	3	2	1	0
	RESTART	BLANK[1:0]		QUAL	KEEP		SRC[1:0]	
Access	R/W	R/W	R/W	R/W	R/W		R/W	R/W
Reset	0	0	0	0	0		0	0

Bits 27:24 – FILTERVAL[3:0] Recoverable Fault n Filter Value
These bits define the filter value applied on MCEx (x=0,1) event input line. The value must be set to zero when MCEx event is used as synchronous event.

Bits 23:16 – BLANKVAL[7:0] Recoverable Fault n Blanking Value
These bits determine the duration of the blanking of the fault input source. Activation and edge selection of the blank filtering are done by the BLANK bits (FCTRLn.BLANK).
When enabled, the fault input source is internally disabled for BLANKVAL* prescaled GCLK_TCCx periods after the detection of the waveform edge.

Bit 15 – BLANKPRESC Recoverable Fault n Blanking Value Prescaler
This bit enables a factor 64 prescaler factor on used as base frequency of the BLANKVAL value.

Value	Description
0	Blank time is BLANKVAL* prescaled GCLK_TCCx.
1	Blank time is BLANKVAL* 64 * prescaled GCLK_TCCx.

Bits 14:12 – CAPTURE[2:0] Recoverable Fault n Capture Action
These bits select the capture and Fault n interrupt/event conditions.

Table 36-10. Fault n Capture Action

Value	Name	Description
0x0	DISABLE	Capture on valid recoverable Fault n is disabled
0x1	CAPT	On rising edge of a valid recoverable Fault n, capture counter value on channel selected by CHSEL[1:0]. INTFLAG.FAULTn flag rises on each new captured value.
0x2	CAPTMIN	On rising edge of a valid recoverable Fault n, capture counter value on channel selected by CHSEL[1:0], if COUNT value is lower than the last stored capture value (CC). INTFLAG.FAULTn flag rises on each local minimum detection.

.....continued		
Value	**Name**	**Description**
0x3	CAPTMAX	On rising edge of a valid recoverable Fault n, capture counter value on channel selected by CHSEL[1:0], if COUNT value is higher than the last stored capture value (CC). INTFLAG.FAULTn flag rises on each local maximun detection.
0x4	LOCMIN	On rising edge of a valid recoverable Fault n, capture counter value on channel selected by CHSEL[1:0]. INTFLAG.FAULTn flag rises on each local minimum value detection.
0x5	LOCMAX	On rising edge of a valid recoverable Fault n, capture counter value on channel selected by CHSEL[1:0]. INTFLAG.FAULTn flag rises on each local maximun detection.
0x6	DERIV0	On rising edge of a valid recoverable Fault n, capture counter value on channel selected by CHSEL[1:0]. INTFLAG.FAULTn flag rises on each local maximun or minimum detection.
0x7	CAPTMARK	Capture with ramp index as MSB value.

Bits 11:10 – CHSEL[1:0] Recoverable Fault n Capture Channel

These bits select the channel for capture operation triggered by recoverable Fault n.

Value	Name	Description
0x0	CC0	Capture value stored into CC0
0x1	CC1	Capture value stored into CC1
0x2	CC2	Capture value stored into CC2
0x3	CC3	Capture value stored into CC3

Bits 9:8 – HALT[1:0] Recoverable Fault n Halt Operation

These bits select the halt action for recoverable Fault n.

Value	Name	Description
0x0	DISABLE	Halt action disabled
0x1	HW	Hardware halt action
0x2	SW	Software halt action
0x3	NR	Non-recoverable fault

Bit 7 – RESTART Recoverable Fault n Restart

Setting this bit enables restart action for Fault n.

Value	Description
0	Fault n restart action is disabled.
1	Fault n restart action is enabled.

Bits 6:5 – BLANK[1:0] Recoverable Fault n Blanking Operation

These bits, select the blanking start point for recoverable Fault n.

Value	Name	Description
0x0	START	Blanking applied from start of the Ramp period
0x1	RISE	Blanking applied from rising edge of the waveform output
0x2	FALL	Blanking applied from falling edge of the waveform output
0x3	BOTH	Blanking applied from each toggle of the waveform output

Bit 4 – QUAL Recoverable Fault n Qualification

Setting this bit enables the recoverable Fault n input qualification.

Value	Description
0	The recoverable Fault n input is not disabled on CMPx value condition.
1	The recoverable Fault n input is disabled when output signal is at inactive level (CMPx == 0).

Bit 3 – KEEP Recoverable Fault n Keep

Setting this bit enables the Fault n keep action.

Value	Description
0	The Fault n state is released as soon as the recoverable Fault n is released.
1	The Fault n state is released at the end of TCC cycle.

Bits 1:0 – SRC[1:0] Recoverable Fault n Source

These bits select the TCC event input for recoverable Fault n.

Event system channel connected to MCEx event input, must be configured to route the event asynchronously, when used as a recoverable Fault n input.

Value	Name	Description
0x0	DISABLE	Fault input disabled
0x1	ENABLE	MCEx (x=0,1) event input
0x2	INVERT	Inverted MCEx (x=0,1) event input
0x3	ALTFAULT	Alternate fault (A or B) state at the end of the previous period.

36.8.6 Waveform Extension Control

Name: WEXCTRL
Offset: 0x14
Reset: 0x00000000
Property: PAC Write-Protection, Enable-Protected

Bit	31	30	29	28	27	26	25	24
	DTHS[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	23	22	21	20	19	18	17	16
	DTLS[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
					DTIEN3	DTIEN2	DTIEN1	DTIEN0
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

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

Bits 31:24 – DTHS[7:0] Dead-Time High Side Outputs Value
This register holds the number of GCLK_TCCx clock cycles for the dead-time high side.

Bits 23:16 – DTLS[7:0] Dead-time Low Side Outputs Value
This register holds the number of GCLK_TCCx clock cycles for the dead-time low side.

Bits 8, 9, 10, 11 – DTIENx Dead-time Insertion Generator x Enable [x=0..3]
Setting any of these bits enables the dead-time insertion generator for the corresponding output matrix. This will override the output matrix [x] and [x+WO_NUM/2], with the low side and high side waveform respectively.

Value	Description
0	No dead-time insertion override.
1	Dead time insertion override on signal outputs[x] and [x+WO_NUM/2], from matrix outputs[x] signal.

Bits 1:0 – OTMX[1:0] Output Matrix
These bits define the matrix routing of the TCC waveform generation outputs to the port pins, according to Waveform Extension. See *Waveform Extension* from Related Links.

Related Links
36.6.3.8. Waveform Extension

36.8.7 Driver Control

Name: DRVCTRL
Offset: 0x18
Reset: 0x00000000
Property: PAC Write-Protection, Enable-Protected

Bit	31	30	29	28	27	26	25	24
	FILTERVAL1[3:0]				FILTERVAL0[3:0]			
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	23	22	21	20	19	18	17	16
			INVEN5	INVEN4	INVEN3	INVEN2	INVEN1	INVEN0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
			NRV5	NRV4	NRV3	NRV2	NRV1	NRV0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
			NRE5	NRE4	NRE3	NRE2	NRE1	NRE0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bits 31:28 – FILTERVAL1[3:0] Non-Recoverable Fault Input 1 Filter Value
These bits define the filter value applied on TCE1 event input line. When the TCE1 event input line is configured as a synchronous event, this value must be 0x0.

Bits 27:24 – FILTERVAL0[3:0] Non-Recoverable Fault Input 0 Filter Value
These bits define the filter value applied on TCE0 event input line. When the TCE0 event input line is configured as a synchronous event, this value must be 0x0.

Bits 16, 17, 18, 19, 20, 21 – INVENx Waveform Output x Inversion [x=0..5]
These bits are used to select inversion on the output of channel x.
Writing a '1' to INVENx inverts output from WO[x].
Writing a '0' to INVENx disables inversion of output from WO[x].

Bits 8, 9, 10, 11, 12, 13 – NRVx NRVx Non-Recoverable State x Output Value [x=0..5]
These bits define the value of the enabled override outputs, under non-recoverable fault condition.

Bits 0, 1, 2, 3, 4, 5 – NREx Non-Recoverable State x Output Enable [x=0..5]
These bits enable the override of individual outputs by NRVx value, under non-recoverable fault condition.

Value	Description
0	Non-recoverable fault tri-state the output.
1	Non-recoverable faults set the output to NRVx level.

36.8.8 Debug control

Name: DBGCTRL
Offset: 0x1E
Reset: 0x00
Property: PAC Write-Protection

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

Bit 2 – FDDBD Fault Detection on Debug Break Detection
This bit is not affected by software Reset and should not be changed by software while the TCC is enabled.
By default this bit is zero, and the on-chip debug (OCD) fault protection is disabled. When this bit is written to '1', OCD break request from the OCD system will trigger non-recoverable fault.
When this bit is set, OCD fault protection is enabled and OCD break request from the OCD system will trigger a non-recoverable fault.

Value	Description
0	No faults are generated when TCC is halted in Debug mode.
1	A non recoverable fault is generated and FAULTD flag is set when TCC is halted in Debug mode.

Bit 0 – DBGRUN Debug Running State
This bit is not affected by software Reset and should not be changed by software while the TCC is enabled.

Value	Description
0	The TCC is halted when the device is halted in Debug mode.
1	The TCC continues normal operation when the device is halted in Debug mode.

36.8.9 Event Control

Name: EVCTRL
Offset: 0x20
Reset: 0x00000000
Property: PAC Write-Protection, Enable-Protected

Bit	31	30	29	28	27	26	25	24
			MCEO5	MCEO4	MCEO3	MCEO2	MCEO1	MCEO0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
			MCEI5	MCEI4	MCEI3	MCEI2	MCEI1	MCEI0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	TCEI1	TCEI0	TCINV1	TCINV0		CNTEO	TRGEO	OVFEO
Access	R/W	R/W	R/W	R/W		R/W	R/W	R/W
Reset	0	0	0	0		0	0	0

Bit	7	6	5	4	3	2	1	0
	CNTSEL[1:0]		EVACT1[2:0]			EVACT0[2:0]		
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 24, 25, 26, 27, 28, 29 – MCEOx Match or Capture Channel x Event Output Enable [x=0..5]
These bits control if the match/capture event on channel x is enabled and will be generated for every match or capture.

Value	Description
0	Match/capture x event is disabled and will not be generated.
1	Match/capture x event is enabled and will be generated for every compare/capture on channel x.

Bits 16, 17, 18, 19, 20, 21 – MCEIx Match or Capture Channel x Event Input Enable [x=0..3]
These bits indicate if the match/capture x incoming event is enabled
These bits are used to enable match or capture input events to the CCx channel of TCC.

Value	Description
0	Incoming events are disabled.
1	Incoming events are enabled.

Bits 14, 15 – TCEIx Timer/Counter Event Input x Enable [x=0..1]
This bit is used to enable input event x to the TCC.

Value	Description
0	Incoming event x is disabled.
1	Incoming event x is enabled.

Bits 12, 13 – TCINVx Timer/Counter Event x Invert Enable [x=0..1]
This bit inverts the event x input.

Value	Description
0	Input event source x is not inverted.
1	Input event source x is inverted.

Bit 10 – CNTEO Timer/Counter Event Output Enable
This bit is used to enable the counter cycle event. When enabled, an event will be generated on begin or end of counter cycle depending on CNTSEL[1:0] settings.

Value	Description
0	Counter cycle output event is disabled and will not be generated.
1	Counter cycle output event is enabled and will be generated depending on CNTSEL[1:0] value.

Bit 9 – TRGEO Retrigger Event Output Enable

This bit is used to enable the counter retrigger event. When enabled, an event will be generated when the counter retriggers operation.

Value	Description
0	Counter retrigger event is disabled and will not be generated.
1	Counter retrigger event is enabled and will be generated for every counter retrigger.

Bit 8 – OVFEO Overflow/Underflow Event Output Enable

This bit is used to enable the overflow/underflow event. When enabled an event will be generated when the counter reaches the TOP or the ZERO value.

Value	Description
0	Overflow/underflow counter event is disabled and will not be generated.
1	Overflow/underflow counter event is enabled and will be generated for every counter overflow/ underflow.

Bits 7:6 – CNTSEL[1:0] Timer/Counter Interrupt and Event Output Selection

These bits define on which part of the counter cycle the counter event output is generated.

Value	Name	Description
0x0	BEGIN	An interrupt/event is generated at begin of each counter cycle
0x1	END	An interrupt/event is generated at end of each counter cycle
0x2	BETWEEN	An interrupt/event is generated between each counter cycle.
0x3	BOUNDARY	An interrupt/event is generated at begin of first counter cycle, and end of last counter cycle.

Bits 5:3 – EVACT1[2:0] Timer/Counter Event Input 1 Action

These bits define the action the TCC will perform on TCE1 event input.

Value	Name	Description
0x0	OFF	Event action disabled.
0x1	RETRIGGER	Start, restart or re-trigger TC on event
0x2	DIR (asynch)	Direction control
0x3	STOP	Stop TC on event
0x4	DEC	Decrement TC on event
0x5	PPW	Period captured into CC0 Pulse Width on CC1
0x6	PWP	Period captured into CC1 Pulse Width on CC0
0x7	FAULT	Non-recoverable Fault

Bits 2:0 – EVACT0[2:0] Timer/Counter Event Input 0 Action

These bits define the action the TCC will perform on TCE0 event input.

Value	Name	Description
0x0	OFF	Event action disabled.
0x1	RETRIGGER	Start, restart or re-trigger TC on event
0x2	COUNTEV	Count on event.
0x3	START	Start TC on event
0x4	INC	Increment TC on EVENT
0x5	COUNT (async)	Count on active state of asynchronous event
0x6	STAMP	Capture overflow times (Max value)
0x7	FAULT	Non-recoverable Fault

36.8.10 Interrupt Enable Clear

Name: INTENCLR
Offset: 0x24
Reset: 0x00000000
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 (INTENSET) register.

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
			MC5	MC4	MC3	MC2	MC1	MC0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	FAULT1	FAULT0	FAULTB	FAULTA	DFS	UFS		
Access	R/W	R/W	R/W	R/W	R/W	R/W		
Reset	0	0	0	0	0	0		

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

Bits 16, 17, 18, 19, 20, 21 – MCx Match or Capture Channel x Interrupt Disable [x=0..5]
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the corresponding Match or Capture Channel x Interrupt Disable/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 15 – FAULT1 Non-Recoverable Fault 1 Interrupt Disable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Non-Recoverable Fault 1 Interrupt Disable/Enable bit, which disables the Non-Recoverable Fault 1 interrupt.

Value	Description
0	The Non-Recoverable Fault 1 interrupt is disabled.
1	The Non-Recoverable Fault 1 interrupt is enabled.

Bit 14 – FAULT0 Non-Recoverable Fault 0 Interrupt Disable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Non-Recoverable Fault 0 Interrupt Disable/Enable bit, which disables the Non-Recoverable Fault 0 interrupt.

Value	Description
0	The Non-Recoverable Fault 0 interrupt is disabled.
1	The Non-Recoverable Fault 0 interrupt is enabled.

Bit 13 – FAULTB Recoverable Fault B Interrupt Disable
Writing a '0' to this bit has no effect.

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

Value	Description
0	The Recoverable Fault B interrupt is disabled.
1	The Recoverable Fault B interrupt is enabled.

Bit 12 – FAULTA Recoverable Fault A Interrupt Disable

Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Recoverable Fault A Interrupt Disable/Enable bit, which disables the Recoverable Fault A interrupt.

Value	Description
0	The Recoverable Fault A interrupt is disabled.
1	The Recoverable Fault A interrupt is enabled.

Bit 11 – DFS Non-Recoverable Debug Fault State Interrupt Disable

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 10 – UFS Non-Recoverable Update Fault Interrupt Disable

Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Non-Recoverable Update Fault Interrupt Disable/Enable bit, which disables the Non-Recoverable Update Fault interrupt.

Value	Description
0	The Non-Recoverable Update Fault interrupt is disabled.
1	The Non-Recoverable Update Fault interrupt is enabled.

Bit 3 – ERR Error Interrupt Disable

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 Error interrupt.

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

Bit 2 – CNT Counter Interrupt Disable

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 Disable

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 Disable

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.

Value	Description
0	The Overflow interrupt is disabled.

Value	Description
1	The Overflow interrupt is enabled.

36.8.11 Interrupt Enable Set

Name: INTENSET
Offset: 0x28
Reset: 0x00000000
Property: PAC Write-Protection

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

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
			MC5	MC4	MC3	MC2	MC1	MC0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	FAULT1	FAULT0	FAULTB	FAULTA	DFS	UFS		
Access	R/W	R/W	R/W	R/W	R/W	R/W		
Reset	0	0	0	0	0	0		

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

Bits 16, 17, 18, 19, 20, 21 – MC Match or Capture Channel x Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the corresponding Match or Capture Channel x Interrupt Disable/Enable bit, which enables 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 15 – FAULT1 Non-Recoverable Fault 1 Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Non-Recoverable Fault 1 Interrupt Disable/Enable bit, which enables the Non-Recoverable Fault 1 interrupt.

Value	Description
0	The Non-Recoverable Fault 1 interrupt is disabled.
1	The Non-Recoverable Fault 1 interrupt is enabled.

Bit 14 – FAULT0 Non-Recoverable Fault 0 Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Non-Recoverable Fault 0 Interrupt Disable/Enable bit, which enables the Non-Recoverable Fault 0 interrupt.

Value	Description
0	The Non-Recoverable Fault 0 interrupt is disabled.
1	The Non-Recoverable Fault 0 interrupt is enabled.

Bit 13 – FAULTB Recoverable Fault B Interrupt Enable
Writing a '0' to this bit has no effect.

Writing a '1' to this bit will set the Recoverable Fault B Interrupt Disable/Enable bit, which enables the Recoverable Fault B interrupt.

Value	Description
0	The Recoverable Fault B interrupt is disabled.
1	The Recoverable Fault B interrupt is enabled.

Bit 12 – FAULTA Recoverable Fault A Interrupt Enable

Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Recoverable Fault A Interrupt Disable/Enable bit, which enables the Recoverable Fault A interrupt.

Value	Description
0	The Recoverable Fault A interrupt is disabled.
1	The Recoverable Fault A interrupt is enabled.

Bit 11 – DFS Non-Recoverable Debug Fault State Interrupt Enable

Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Debug Fault State Interrupt Disable/Enable bit, which enables the Debug Fault State interrupt.

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

Bit 10 – UFS Non-Recoverable Update Fault Interrupt Enable

Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Non-Recoverable Update Fault Interrupt Disable/Enable bit, which enables the Non-Recoverable Update Fault interrupt.

Value	Description
0	The Non-Recoverable Update Fault interrupt is disabled.
1	The Non-Recoverable Update Fault 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 set the Error Interrupt Disable/Enable bit, which enables the Error 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 set the Counter Interrupt Disable/Enable bit, which enables 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 set the Retrigger Interrupt Disable/Enable bit, which enables 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 set the Overflow Interrupt Disable/Enable bit, which enables the Overflow interrupt.

Value	Description
0	The Overflow interrupt is disabled.

Value	Description
1	The Overflow interrupt is enabled.

36.8.12 Interrupt Flag Status and Clear

Name: INTFLAG
Offset: 0x2C
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
			MC5	MC4	MC3	MC2	MC1	MC0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	FAULT1	FAULT0	FAULTB	FAULTA	DFS	UFS		
Access	R/W	R/W	R/W	R/W	R/W	R/W		
Reset	0	0	0	0	0	0		

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

Bits 16, 17, 18, 19, 20, 21 – MCx Match or Capture Channel x Interrupt Flag [x=0..5]
This flag is set on the next CLK_TCC_COUNT cycle after a match with the compare condition or once the CCx register contains a valid capture value.
Writing a '0' to one of these bits has no effect.
Writing a '1' to one of these bits will clear the corresponding Match or Capture Channel x interrupt flag.
In the Capture operation, this flag is automatically cleared when the CCx register is read.

Bit 15 – FAULT1 Non-Recoverable Fault 1 Interrupt Flag
This flag is set on the next CLK_TCC_COUNT cycle after a Non-Recoverable Fault 1 occurs.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Non-Recoverable Fault 1 interrupt flag.

Bit 14 – FAULT0 Non-Recoverable Fault 0 Interrupt Flag
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Non-Recoverable Fault 0 interrupt flag.

Bit 13 – FAULTB Recoverable Fault B Interrupt Flag
This flag is set on the next CLK_TCC_COUNT cycle after a Recoverable Fault B occurs.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Recoverable Fault B interrupt flag.

Bit 12 – FAULTA Recoverable Fault A Interrupt Flag
This flag is set on the next CLK_TCC_COUNT cycle after a Recoverable Fault A occurs.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Recoverable Fault A interrupt flag.

Bit 11 – DFS Non-Recoverable Debug Fault State Interrupt Flag
This flag is set on the next CLK_TCC_COUNT cycle after a Debug Fault State occurs.

Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Debug Fault State interrupt flag.

Bit 10 – UFS Non-Recoverable Update Fault Interrupt Flag
This flag is set when the RAMP index changes and the Lock Update bit is set (CTRLBSET.LUPD).
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Non-Recoverable Update Fault Interrupt Flag.

Bit 3 – ERR Error Interrupt Flag
This flag is set if a new capture occurs on a channel while the corresponding Match or Capture Channel x interrupt flag is one. In which case, there is no place to store the new capture.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Error interrupt flag.

Bit 2 – CNT Counter Interrupt Flag
This flag is set on the next CLK_TCC_COUNT cycle after a counter event occurs.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the CNT interrupt flag.

Bit 1 – TRG Retrigger Interrupt Flag
This flag is set on the next CLK_TCC_COUNT cycle after a counter retrigger occurs.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Retrigger interrupt flag.

Bit 0 – OVF Overflow Interrupt Flag
This flag is set on the next CLK_TCC_COUNT cycle after an overflow condition occurs.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Overflow interrupt flag.

36.8.13 Status

Name: STATUS
Offset: 0x30
Reset: 0x00000001
Property: -

Note: Clear STATUS register bits by 32-bits write access only.

Bit	31	30	29	28	27	26	25	24
			CMP5	CMP4	CMP3	CMP2	CMP1	CMP0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
			CCBUFV5	CCBUFV4	CCBUFV3	CCBUFV2	CCBUFV1	CCBUFV0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	FAULT1	FAULT0	FAULTB	FAULTA	FAULT1IN	FAULT0IN	FAULTBIN	FAULTAIN
Access	R/W	R/W	R/W	R/W	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	PERBUFV		PATTBUFV	SLAVE	DFS	UFS	IDX	STOP
Access	R/W		R/W	R	R/W	R/W	R	R
Reset	0		0	0	0	0	0	1

Bits 24, 25, 26, 27, 28, 29 – CMPx Channel x Compare Value [x=0..5]
This bit reflects the channel x output compare value.

Value	Description
0	Channel compare output value is 0.
1	Channel compare output value is 1.

Bits 16, 17, 18, 19, 20, 21 – CCBUFVx Channel x Compare or Capture Buffer Valid [x=0..5]
For a compare channel, this bit is set when a new value is written to the corresponding CCBUFx register. The bit is cleared either by writing a '1' to the corresponding location when CTRLB.LUPD is set, or automatically on an UPDATE condition.
For a capture channel, the bit is set when a valid capture value is stored in the CCBUFx register. The bit is automatically cleared when the CCx register is read.

Bits 14, 15 – FAULTx Non-recoverable Fault x State [x=0..1]
This bit is set by hardware as soon as non-recoverable Fault x condition occurs.
This bit is cleared by writing a one to this bit and when the corresponding FAULTxIN status bit is low.
Once this bit is clear, the timer/counter will restart from the last COUNT value. To restart the timer/counter from BOTTOM, the timer/counter restart command must be executed before clearing the corresponding FAULTx bit. For further details on timer/counter commands, refer to available commands description (CTRLBSET.CMD).

Bit 13 – FAULTB Recoverable Fault B State
This bit is set by hardware as soon as recoverable Fault B condition occurs.
This bit can be cleared by hardware when Fault B action is resumed, or by writing a '1' to this bit when the corresponding FAULTBIN bit is low. If software halt command is enabled (FAULTB.HALT=SW), clearing this bit will release the timer/counter.

Bit 12 – FAULTA Recoverable Fault A State
This bit is set by hardware as soon as recoverable Fault A condition occurs.

This bit can be cleared by hardware when Fault A action is resumed, or by writing a '1' to this bit when the corresponding FAULTAIN bit is low. If software halt command is enabled (FAULTA.HALT=SW), clearing this bit will release the timer/counter.

Bit 11 – FAULT1IN Non-Recoverable Fault 1 Input
This bit is set while an active Non-Recoverable Fault 1 input is present.

Bit 10 – FAULT0IN Non-Recoverable Fault 0 Input
This bit is set while an active Non-Recoverable Fault 0 input is present.

Bit 9 – FAULTBIN Recoverable Fault B Input
This bit is set while an active Recoverable Fault B input is present.

Bit 8 – FAULTAIN Recoverable Fault A Input
This bit is set while an active Recoverable Fault A input is present.

Bit 7 – PERBUFV Period Buffer Valid
This bit is set when a new value is written to the PERBUF register. This bit is automatically cleared by hardware on UPDATE condition when CTRLB.LUPD is set, or by writing a '1' to this bit.

Bit 5 – PATTBUFV Pattern Generator Value Buffer Valid
This bit is set when a new value is written to the PATTBUF register. This bit is automatically cleared by hardware on UPDATE condition when CTRLB.LUPD is set, or by writing a '1' to this bit.

Bit 4 – SLAVE Client Mode Enable
This bit is set when TCC is set in Client mode. This bit follows the CTRLA.MSYNC bit state.

Bit 3 – DFS Debug Fault State
This bit is set by hardware in Debug mode when DDBGCTRL.FDDBD bit is set. The bit is cleared by writing a '1' to this bit and when the TCC is not in Debug mode.
When the bit is set, the counter is halted and the Waveforms state depend on DRVCTRL.NRE and DRVCTRL.NRV registers.

Bit 2 – UFS Non-recoverable Update Fault State
This bit is set by hardware when the RAMP index changes and the Lock Update bit is set (CTRLBSET.LUPD). The bit is cleared by writing a one to this bit.
When the bit is set, the waveforms state depend on DRVCTRL.NRE and DRVCTRL.NRV registers.

Bit 1 – IDX Ramp Index
In RAMP2 and RAMP2A operation, the bit is cleared during the cycle A and set during the cycle B. In RAMP1 operation, the bit always reads zero. See *Ramp Operations* from Related Links.

Bit 0 – STOP Stop
This bit is set when the TCC is disabled either on a STOP command or on an UPDATE condition when One-Shot operation mode is enabled (CTRLBSET.ONESHOT=1).
This bit is cleared on the next incoming counter increment or decrement.

Value	Description
0	Counter is running.
1	Counter is stopped.

Related Links
36.6.3.4. Ramp Operations

36.8.14 Counter Value

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

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).

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 MSBs 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

36.8.15 Pattern

Name: PATT
Offset: 0x38
Reset: 0x0000
Property: Write-Synchronized

Bit	15	14	13	12	11	10	9	8
			PGV5	PGV4	PGV3	PGV2	PGV1	PGV0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
			PGE5	PGE4	PGE3	PGE2	PGE1	PGE0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bits 8, 9, 10, 11, 12, 13 – PGVx Pattern Generation Output Value [x=0..5]
This register holds the values of pattern for each waveform output.

Bits 0, 1, 2, 3, 4, 5 – PGEx Pattern Generation Output Enable [x=0..5]
This register holds the enable status of pattern generation for each waveform output. A bit written to '1' will override the corresponding SWAP output with the corresponding PGVx value.

36.8.16 Waveform

Name: WAVE
Offset: 0x3C
Reset: 0x00000000
Property: Write-Synchronized

Bit	31	30	29	28	27	26	25	24
					SWAP3	SWAP2	SWAP1	SWAP0
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	23	22	21	20	19	18	17	16
			POL5	POL4	POL3	POL2	POL1	POL0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
					CICCEN3	CICCEN2	CICCEN1	CICCEN0
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

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

Bits 24, 25, 26, 27 – SWAPx Swap DTI Output Pair x [x=0..3]
Setting these bits enables output swap of DTI outputs [x] and [x+WO_NUM/2]. Note the DTIxEN settings will not affect the swap operation.

Bits 16, 17, 18, 19, 20, 21 – POLx Channel Polarity x [x=0..5]
Setting these bits enables the output polarity in single-slope and dual-slope PWM operations.

Value	Name	Description
0	(single-slope PWM waveform generation)	Compare output is initialized to ~DIR and set to DIR when TCC counter matches CCx value
1	(single-slope PWM waveform generation)	Compare output is initialized to DIR and set to ~DIR when TCC counter matches CCx value.
0	(dual-slope PWM waveform generation)	Compare output is set to ~DIR when TCC counter matches CCx value
1	(dual-slope PWM waveform generation)	Compare output is set to DIR when TCC counter matches CCx value.

Bits 8, 9, 10, 11 – CICCENx Circular CC Enable x [x=0..3]
Setting these bits enables the compare circular buffer option on the first four Compare/Capture channels. When the bit is set, CCx register value is copied-back into the CCBUFx register on UPDATE condition.

Bit 7 – CIPEREN Circular Period Enable
Setting this bit enables the period circular buffer option. When the bit is set, the PER register value is copied-back into the PERBUF register on UPDATE condition.

Bits 5:4 – RAMP[1:0] Ramp Operation
These bits select Ramp operation (RAMP). These bits are not synchronized.

Value	Name	Description
0x0	RAMP1	RAMP1 operation
0x1	RAMP2A	Alternative RAMP2 operation

Value	Name	Description
0x2	RAMP2	RAMP2 operation
0x3	RAMP2C	Critical RAMP2 operation

Bits 2:0 – WAVEGEN[2:0] Waveform Generation Operation

These bits select the waveform generation operation. The settings impact the top value and control if frequency or PWM waveform generation should be used. These bits are not synchronized.

Value	Name	Description						
		Operation	Top	Update	Waveform Output On Match	Waveform Output On Update	OVFIF/Event Up Down	
0x0	NFRQ	Normal Frequency	PER	TOP/Zero	Toggle	Stable	TOP	Zero
0x1	MFRQ	Match Frequency	CC0	TOP/Zero	Toggle	Stable	TOP	Zero
0x2	NPWM	Normal PWM	PER	TOP/Zero	Set	Clear	TOP	Zero
0x3	Reserved	-	-	-	-	-	-	-
0x4	DSCRITICAL	Dual-slope PWM	PER	Zero	~DIR	Stable	–	Zero
0x5	DSBOTTOM	Dual-slope PWM	PER	Zero	~DIR	Stable	–	Zero
0x6	DSBOTH	Dual-slope PWM	PER	TOP & Zero	~DIR	Stable	TOP	Zero
0x7	DSTOP	Dual-slope PWM	PER	Zero	~DIR	Stable	TOP	–

36.8.17 Period Value

Name: PER
Offset: 0x40
Reset: 0x00FFFFFF
Property: Write-Synchronized

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	PER[17:10]							
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	15	14	13	12	11	10	9	8
	PER[9:2]							
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
	PER[1:0]		DITHER[5:0]					
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 23:6 – PER[17:0] Period Value
These bits hold the value of the TCC period count. The number of bits in this field corresponds to the size of the counter.
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 bits in the Control A register (CTRLA.RESOLUTION):

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

Bits 5:0 – DITHER[5:0] Dithering Cycle Number
These bits hold the number of extra cycles that are added on the PWM pulse period every 64 PWM frames.
Note: This bit field consists of the n LSB of the register. n is dependent on the value of the Resolution bits in the Control A register (CTRLA.RESOLUTION):

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

36.8.18 Compare/Capture Channel x

Name: CCx
Offset: 0x44 + x*0x04 [x=0..5]
Reset: 0x00000000
Property: Write-Synchronized, Read-Synchronized

The CCx register represents the 16-, 24- bit value, CCx. The register has two functions, depending of the mode of operation.

For capture operation, this register represents the second buffer level and access point for the CPU and DMA.

For compare operation, this register is continuously compared to the counter value. Normally, the output form the comparator is then used for generating waveforms.

CCx register is updated with the buffer value from their corresponding CCBUFx register when an UPDATE condition occurs.

In addition, in match frequency operation, the CC0 register controls the counter period.

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	CC[17:10]							
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[9:2]							
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[1:0]		DITHER[5:0]					
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 23:6 – CC[17:0] Channel x Compare/Capture Value
These bits hold the value of the Channel x compare/capture register.
Notes:

1. When the TCC is configured as a 16-bit timer/counter, the excess bits are read as zero.
2. 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
0x1 - DITH4	23:4
0x2 - DITH5	23:5
0x3 - DITH6	23:6 (depicted)

Bits 5:0 – DITHER[5:0] Dithering Cycle Number
These bits hold the number of extra cycles that are added on the PWM pulse width every 64 PWM frames.

Note: This bit field consists of the n LSB of the register. n is dependent on the value of the Resolution bits in the Control A register (CTRLA.RESOLUTION):

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

36.8.19 Pattern Buffer

Name: PATTBUF
Offset: 0x64
Reset: 0x0000
Property: Write-Synchronized, Read-Synchronized

Bit	15	14	13	12	11	10	9	8
			PGVB5	PGVB4	PGVB3	PGVB2	PGVB1	PGVB0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
			PGEB5	PGEB4	PGEB3	PGEB2	PGEB1	PGEB0
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bits 8, 9, 10, 11, 12, 13 – PGVBx Pattern Generation Output Value Buffer [x=0..5]
This register is the buffer for the PGV register. If double buffering is used, valid content in this register is copied to the PGVx register on an UPDATE condition.

Bits 0, 1, 2, 3, 4, 5 – PGEBx Pattern Generation Output Enable Buffer [x=0..5]
This register is the buffer of the PGE register. If double buffering is used, valid content in this register is copied into the PGEx register at an UPDATE condition.

36.8.20 Period Buffer Value

Name: PERBUF
Offset: 0x6C
Reset: 0x00FFFFFF
Property: Write-Synchronized, Read-Synchronized

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	PERBUF[17:10]							
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	15	14	13	12	11	10	9	8
	PERBUF[9:2]							
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
	PERBUF[1:0]		DITHERBUF[5:0]					
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 23:6 – PERBUF[17:0] Period Buffer Value
These bits hold the value of the Period Buffer register. The value is copied to PER register on UPDATE condition.
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 bits in the Control A register (CTRLA.RESOLUTION):

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

Bits 5:0 – DITHERBUF[5:0] Dithering Buffer Cycle Number
These bits represent the PER.DITHER bits buffer. When the double buffering is enabled, the value of this bit field is copied to the PER.DITHER bits on an UPDATE condition.
Note: This bit field consists of the n LSB of the register. n is dependent on the value of the Resolution bits in the Control A register (CTRLA.RESOLUTION):

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

36.8.21 Channel x Compare/Capture Buffer Value

Name: CCBUFx
Offset: 0x70 + x*0x04 [x=0..5]
Reset: 0x00000000
Property: Write-Synchronized, Read-Synchronized

CCBUFx is copied into CCx at TCC update time

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	CCBUF[17:10]							
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
	CCBUF[9:2]							
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
	CCBUF[1:0]		DITHERBUF[5:0]					
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 23:6 – CCBUF[17:0] Channel x Compare/Capture Buffer Value [x=0..5]
These bits hold the value of the Channel x Compare/Capture Buffer Value register. The register serves as the buffer for the associated compare or capture registers (CCx). Accessing this register using the CPU or DMA will affect the corresponding CCBUFVx status bit.
Notes:

1. When the TCC is configured as a 16-bit timer/counter, the excess bits are read as zero.
2. This bit field occupies the MSB of the register, [23:m]. m is dependent on the Resolution bits in the Control A register (CTRLA.RESOLUTION):

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

Bits 5:0 – DITHERBUF[5:0] Dithering Buffer Cycle Number
These bits represent the CCx.DITHER bits buffer. When the double buffering is enable, DITHERBUF bits value is copied to the CCx.DITHER bits on an UPDATE condition.
Note: This bit field consists of the n LSB of the register. n is dependent on the value of the Resolution bits in the Control A register (CTRLA.RESOLUTION):

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

37. Zigbee Bluetooth Radio Subsystem (ZBT)

37.1 Overview

The PIC32CX-BZ3 and the WBZ35x provides an on-chip IEEE 802.15.4 and 802.15.3 compliant Zigbee, Bluetooth Low Energy 5.2 interface with integrated transceivers. The Wireless Subsystem block comprises of the following modules:

- Single ultra-low power 2.4 GHz ISM band RF transceiver
- Radio arbiter
- Bluetooth modem
- Bluetooth link layer
- ZigBee modem
- ZigBee MAC

The radio arbiter hardware allows to establish simultaneous Bluetooth and ZigBee links with dynamic switching and programmable QoS. The RF transceiver includes a switching power amplifier architecture, and a transmit/receive switch. Therefore, medium to high power application use cases are supported without an external front-end module (FEM).

With integrated Ultra Low Power 2.4 GHz ISM band single transceiver, dual modem, and dual MAC, the Radio supports dual Zigbee and Bluetooth 5.2 link protocols. An on board intelligent Radio Arbiter hardware module establishes both the links simultaneously using single Radio Transmit/Receive with programmable QoS. The RF transceiver includes Switching Power Amplifiers architecture, and TR Switch. Therefore medium to high power application use cases are supported without external FEM.

The RTOS which is running on the Cortex M4F CPU handles the arbitration between the application, the Bluetooth link stack, the ZigBee link stack, and miscellaneous maintenance tasks.

37.2 Features

2.4 GHz RF Transceiver

- Integrated 2.4 GHz Ultra-low Power RF Transceiver Shared between Bluetooth and Zigbee Modems and Link (MAC) Controllers
- Integrated Crystal Oscillator with Support for 16 MHz Crystal
- Internal Parallel PA Paths, which may be Shut Down to Control Pout as well as Current Consumption to Improve TX Power Efficiency
- Low BOM Single-ended TRX RFFE Architecture
  - Integrated balun (single ended RF output) and TRX switch
- Hardware Radio Arbiter with Programmable QoS:
  - Resolution: up to per packet level
  - Time-division coexistence between Bluetooth and 802.15.4
  - Based on shared transceiver and antenna

Bluetooth

- Bluetooth Low Energy 5.2 Certified
- Up to +10 dBm Programmable Transmit Output Power
- Typical Receiver Power Sensitivity:
  - -97 dBm for Bluetooth low energy 1 Mbps
  - -95 dBm for Bluetooth low energy 2 Mbps
  - -108 dBm for Bluetooth low energy 125 Kbps
  - -102 dBm for Bluetooth low energy 500 Kbps
- Wideband RSSI

- Enables Interference Robustness and Higher Tolerance to Out-of-Band Blockers
- Bluetooth Supported Features:
  - 2M uncoded PHY
  - Long range (Coded PHY)
  - Channel selection algorithm #2
  - Advertising extensions, offloads CPU with hardware based scheduler
  - High duty cycle non-connectible advertising
  - Data length extensions
  - Secure connections
  - Privacy upgrades (with hardware white-list support)
- ECDH P256 Hardware Engine for Link Key Generation When Bluetooth Pairing
- AES128 Hardware Module for Real-Time Bluetooth Payload Data Encryption
- HCI Interface via UART
- LE Power Control
- Bluetooth Low Energy Profiles/Services:
  - Bluetooth Low Energy peripheral and central roles
  - Bluetooth Low Energy APIs for application layer to implement standard or customize GATT based profiles/services
  - Alert notification service
  - Proximity reporter (PXP)
  - Time information

Zigbee

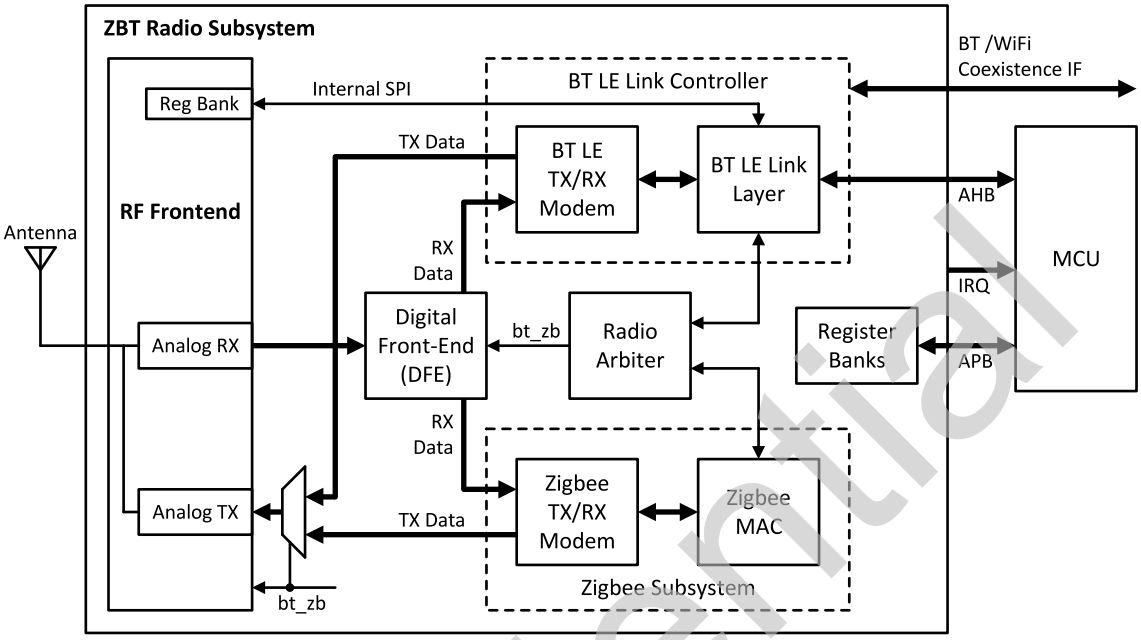
- PSDU Data Rate: 250 Kbps
- Programmable RX Mode
  - -103 dBm RX sensitivity (typical) in the Continuous mode
  - -96 dBm sensitivity in the RPC mode
  - RPC mode provides lower power consumption in the RX mode to support California Green Energy Specification at the system level

Proprietary Modulation Schemes

- Typical sensitivity level for the proprietary data rates:
  - 500 Kbps: -98 dBm
  - 1 Mbps: -96 dBm
  - 2 Mbps: -90 dBm
- TX Output Power up to +10 dBm
- Hardware Assisted MAC
  - Auto acknowledge
  - Auto retry
  - Channel access back-off
- SFD Detection; Spreading; De-spreading; Framing; CRC-16 Computation
- Independent TX/RX Buffers for Improved CPU Offloading while Handling Zigbee Data
  - 128-byte TX and 128-byte RX frame buffer
- Hardware Security
  - Advanced Encryption Standard (AES)
  - True Random Number Generator (TRNG)
- Zigbee Stack Support
  - Zigbee 3.0 ready
  - Zigbee Pro 2017
  - Zigbee green power support (proxy, sink, and multi sensor)

37.3 Wireless Subsystem Top Level Diagram

Figure 37-1. Wireless Subsystem Top Level Diagram



37.4 Analog RF Front-end

The analog front-end contains the following:

- RF antenna switch
- XTO, connecting to a 16 MHz crystal oscillator
- RF synthesizer
- Analog RX path (LNA, Mixer, IF amplifier)
- ADC
- Analog TX path (Power amplifier)
- System clock PLL
- RF-related power management (RF-LDOs)
- Analog control logic

37.5 Digital Front-end

The digital front-end (DFE) connects to the analog front-end on one side and the Bluetooth or Zigbee baseband on the other side. In the receive path, the I/Q data from the ADC are filtered and down-converted. Automatic gain control is also implemented in the DFE. In the transmit path, the Zigbee power control is done in the DFE.

37.6 Bluetooth Low-Energy Link Controller

The Bluetooth Low-Energy Link Controller consists of the TX/RX modem and the link layer. The modem is responsible for modulation and demodulation of the digital IF data for both, Bluetooth classic and Bluetooth Low-Energy.

The baseband link layer carries out all Bluetooth operations as transmit or receive tasks. There are several task masters:

1. Firmware – Firmware can trigger tasks by writing the task controller registers
2. Hardware Schedule Controller – This is Bluetooth low-energy advertisement scheduler controller (Advertiser role)
3. Hardware Scanner – This is Bluetooth LE advertisement scanner (Central role)
4. Bluetooth Low-Energy advertisement controller: This is Bluetooth 4.2 advertisement controller (Advertiser role)

The requests from the task masters are arbitrated and carried out by the task controller.

The RX and TX data are written to common memory via DMA over an AHB interface.

37.7 Zigbee Subsystem

The Zigbee Subsystem deals with the physical/modem layer and the MAC layer of the Zigbee transceiver. A digital baseband processor down-converts and demodulates the received IF data, the subsequent modules implement basic MAC functionality and hardware accelerators for features, such as automatic acknowledgment, CSMA_CA and retransmission or automatic FCS check. The flow control is done by a Finite State Machine (FSM). The data is stored in integrated 128-bytes RX and TX frame buffers.

37.8 Radio Arbiter

The Radio Arbiter provides a low-level arbitration for using the single RF frontend for both, Zigbee and Bluetooth links. The arbiter supports the following modes:

- BT static – Radio ownership is with the Bluetooth link controller.
- ZB static – Radio ownership is with the Zigbee subysystem.
- Dynamic – Radio ownership is decided dynamically at every arbitration event.

In the static modes, either the Bluetooth link controller or the Zigbee subsystem continuously own the radio. No arbitration is done.

Every firmware command or transaction which is issued to the Bluetooth link layer or the ZigBee MAC results in a radio request from the corresponding controller to the arbiter hardware. The firmware stack must accompany each request by the following three main attributes:

- Base priority (4-bit)
- Time sensitivity (transaction is timing critical)
- Atomic nature (burst mode)

In the dynamic mode, the ownership of the radio is decided at every arbitration event. In between events the radio is retained by the current owner.

37.9 Register Banks

The ZBT subsystem contains various register banks to configure and control the digital baseband hardware and state machines. The firmware accesses the registers by the ARM APB bus.

37.10 Coexistence Interface

Bluetooth and Wi-Fi partly operate in the same 2.4 GHz band. Therefore, transmissions can interfere with each other and impact the performance and reliability of the wireless systems. The Coexistence interface is built of a 3-wire bus that allows to mutually signalize RF activity when Bluetooth and Wi-Fi chips are available in the same hardware module. Interference can thus be avoided.

38. Electrical Characteristics

This chapter provides the Electrical Specification and Characteristic of PIC32CX-BZ3 and WBZ35x Module across the operating temperature range of the product.

38.1 Absolute Maximum Electrical Characteristics

Exposure to these maximum rating conditions for extended periods may affect device reliability. Functional operation of the device at these or any other conditions above the parameters indicated in the operation listings of this specification is not implied.

Table 38-1. Absolute Maximum Ratings

Parameter	Value
Ambient temperature under bias(Note)	-40°C to +125°C
Storage temperature	-65°C to +150°C
Voltage on V_{DD} / V_{DDIO} with respect to GND	-0.3V to +4.0V
Voltage on any tolerant pins, with respect to GND	-0.3V to (V_{DD} + 0.3V)
Maximum current out of GND pins	300 mA
Maximum current into V_{DD} pins(2)	300 mA
Maximum output current sourced/sunk by any Low Current Mode I/O pin (4x drive strength)	10 mA
Maximum output current sourced/sunk by any High Current Mode I/O pin (8x drive strength)	15 mA
Maximum current sink by all ports	120 mA
Maximum current sourced by all ports(2)	120 mA
ESD Qualification	
Human Body Model (HBM) per JESD22-A114	2000V
Charged Device Model (CDM) (ANSI/ESD STM 5.3.1)…(All pins / Corner pins)	+500V/- 500V

Notes:

1. Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.
2. Maximum allowable current is a function of device maximum power dissipation (See the *Thermal Operating Conditions* table in the *Thermal Specifications* from Related Links).

Related Links

38.3. Thermal Specifications

38.2 DC Electrical Characteristics

Table 38-2. Operating Frequency VS. Voltage

Param. No.	V_{DDIO}, V_{DDANA} Range	Temp. Range (in °C)	Max. MCU Frequency	Comments
DC_5	1.9V to 3.6V	-40°C to +85°C	64 MHz	Industrial

.....continued				
Param. No.	**V_{DDIO}, V_{DDANA} Range**	**Temp. Range (in °C)**	**Max. MCU Frequency**	**Comments**
DC_7	1.9V to 3.6V	-40°C to +125°C	64 MHz	Extended

Note: The same voltage must be applied to V_{DDIN} and V_{DDANA}.

38.3 Thermal Specifications

Table 38-3. Thermal Operating Conditions

Rating	Symbol	Min.	Typ	Max.	Unit
Industrial Temperature Devices:					
Operating ambient temperature range	T_A	-40	—	+85	°C
Operating junction temperature range	T_J	-40	—	+105	°C
V-Temp Temperature Devices:					
Operating ambient temperature range	T_A	-40	—	+105	°C
Operating junction temperature range	T_J	-40	—	+125	°C
Extended Temperature Range:					
Operating ambient temperature range	T_A	-40	—	+125	°C
Operating junction temperature range	T_J	-40	—	+135	°C
Maximum allowed power dissipation	P_{DMAX}	$(T_J - T_A)/\Theta_{JA}$			W

38.4 Active Current Consumption DC Electrical Specifications

Table 38-4. Active Current Consumption DC Electrical Specifications

DC Characteristics				**Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp**			
Param. No.	**Symbol**	**Characteristics**	**Clock/Freq**	**Typ (1)**	**Max.**	**Units**	**Conditions**
APWR_1	I_{DD_ACTIVE} (2,3)	MCU I_{DD} in Active mode w/LDO mode selected	PLL 64 MHz	131	—	µA/MHz	V_{DD} = V_{DDANA} = 3.3V
APWR_2			FRC 8 MHz	443			
APWR_3		MCU I_{DD} in Active mode w/ BUCK mode selected	PLL 64 MHz	76.8			
APWR_4			FRC 8 MHz	300			

Figure 38-1. Run Mode Current Consumption in Buck Mode at PLL 64 MHz

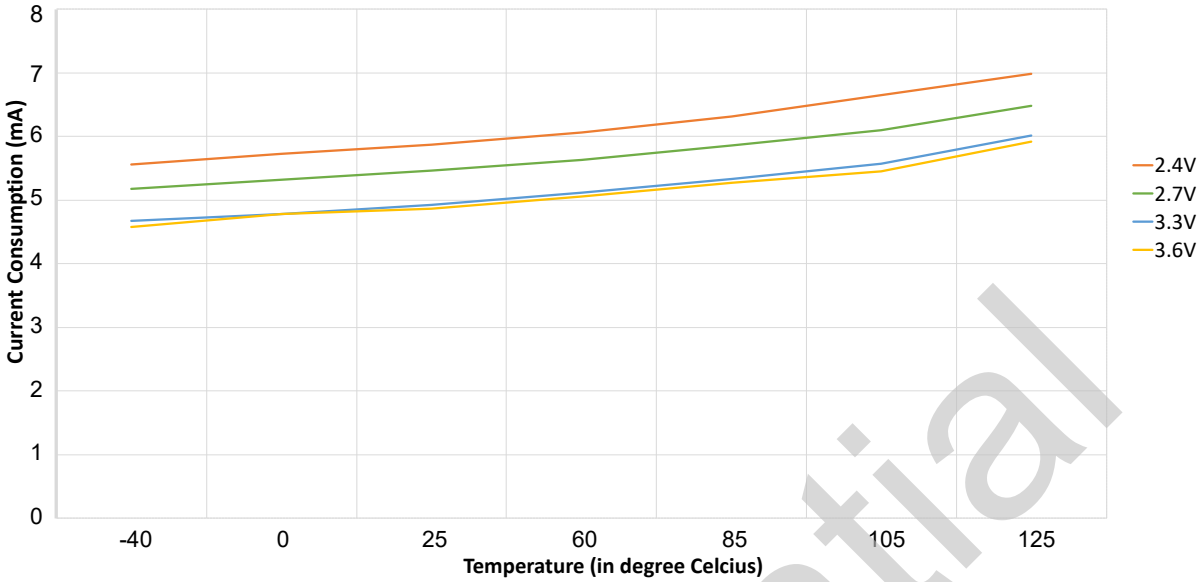
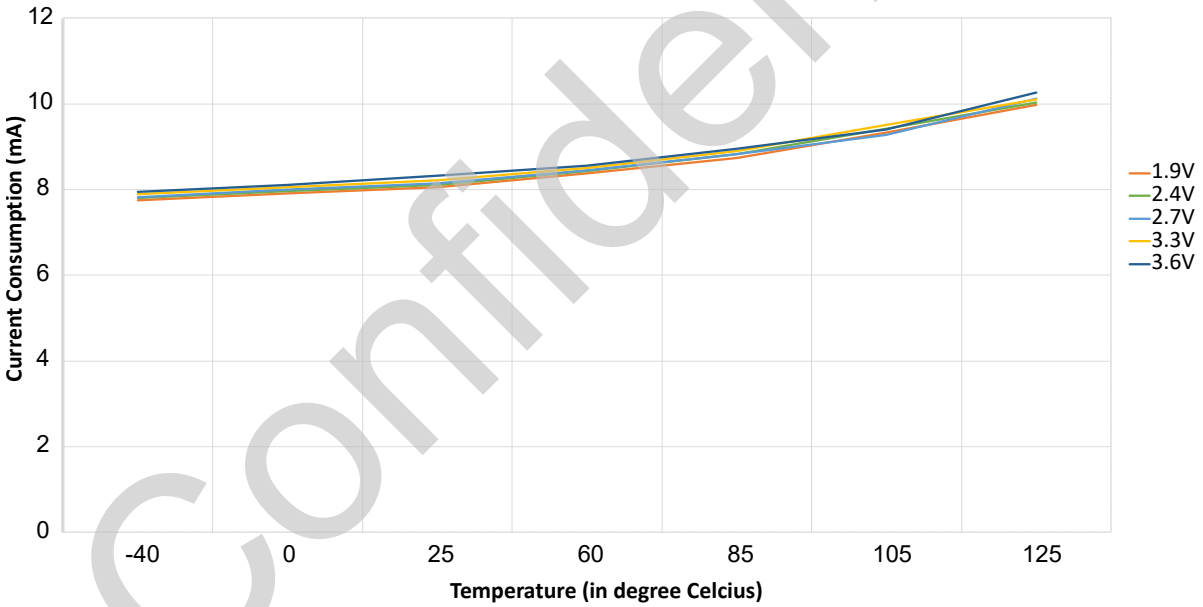


Figure 38-2. Run Mode Current Consumption in MLDO Mode at PLL 64 MHz



Notes:

1. Typical value measured at 3.3V and 25℃.
2. The test conditions are as follows:
   – All GPIO except RPA3 pulled up. RPA3 is pulled down.
   – RF system in IDLE state. Not transmitting or receiving packets.
   – All peripherals are disabled.
   – All PB clocks are divided by 16.
   – LPRC is set as LPCLK.
   – SOSC is disabled.
   – PMU 1 MHz clock is derived from FRC. FRC is divided by 8.
   – Cache is enabled and configured to wait states PFMWS[3:0] as 0xF.
   – Backup RAM in the Retention mode.
   – DSU is disconnected.
3. MCU running while(1) loop with 50 NOP instructions.

38.5 Idle Current Consumption DC Electrical Specifications

Table 38-5. Idle Current Consumption DC Electrical Specifications

DC Characteristics				Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp			
Param. No.	**Symbol**	**Characteristics**	**Clock/Freq**	**Typ(1)**	**Max.**	**Units**	**Conditions (3)**
IPWR_1	I_{DD_IDLE} (2)	MCU I_{DD} in IDLE mode w/LDO mode selected	PLL 64 MHz	83	—	µA/MHz	V_{DDIO} = V_{DDANA} = 3.3V
IPWR_2			FRC 8 MHz	336			
IPWR_3		MCU I_{DD} in IDLE mode w/BUCK mode selected	PLL 64 MHz	52.8	—		
IPWR_4			FRC 8 MHz	282			

Figure 38-3. Idle Mode Current Consumption in Buck Mode at PLL 64 MHz

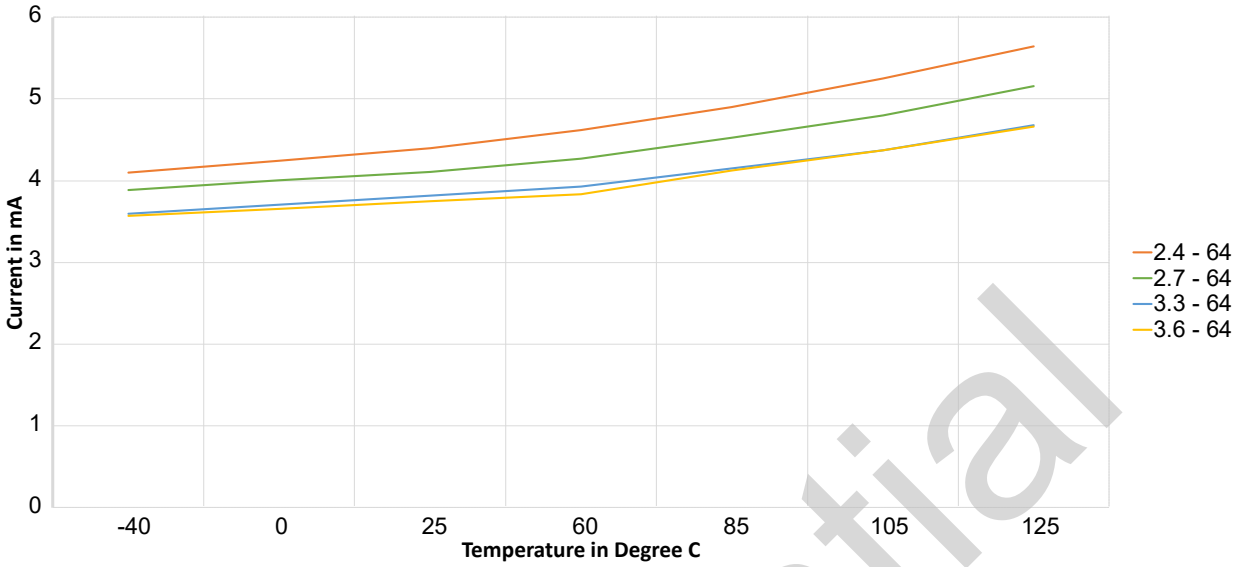
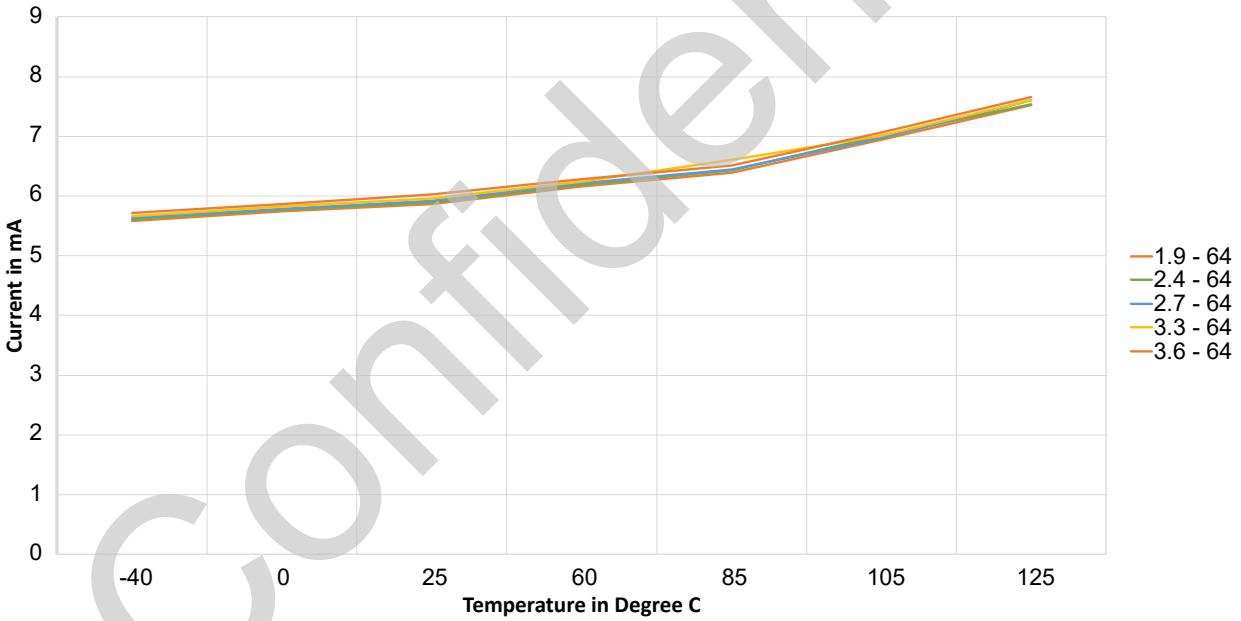


Figure 38-4. Idle Mode Current Consumption in MLDO Mode at PLL 64 MHz



Notes:

1. Typical value measured during characterization at 3.3V and 25℃.
2. The test conditions are as follows:
   - All GPIO except RPA3 are pulled up. RPA3 is pulled down.
   - All peripherals are disabled.
   - All PB clocks are divided by 16.
   - LPRC is set as LPCLK.
   - SOSC is disabled.
   - PMU 1 MHz clock is derived from FRC. FRC is divided by 8.
   - Cache is enabled and configured to wait states PFMWS[3:0] as 0xF.
   - Backup RAM in the Retention mode.
   - DSU is disconnected.
   - RF system clock is gated.
   - Entry to the Sleep mode is disabled and WFI instruction is executed.
   - No active RF transmission or reception during current measurement.

38.6 Sleep Current Consumption DC Electrical Specifications

Table 38-6. Sleep Current Consumption DC Electrical Specifications

DC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**V_{DDIO}**	**Typ(1)**	**Max.**	**Units**	**Conditions**
SPWR_1	I_{DD_SLEEP}	MCU I_{DD} in Sleep mode w/LDO mode selected	3.3V	950	—	µA	XTAL ON
			3.3V	620	—	µA	XTAL OFF
SPWR_29		MCU I_{DD} in Sleep mode w/ BUCK mode selected	3.3V	470	—	µA	XTAL ON
			3.3V	450	—	µA	XTAL OFF

Figure 38-5. Sleep Current with XTAL_OFF_MLDO

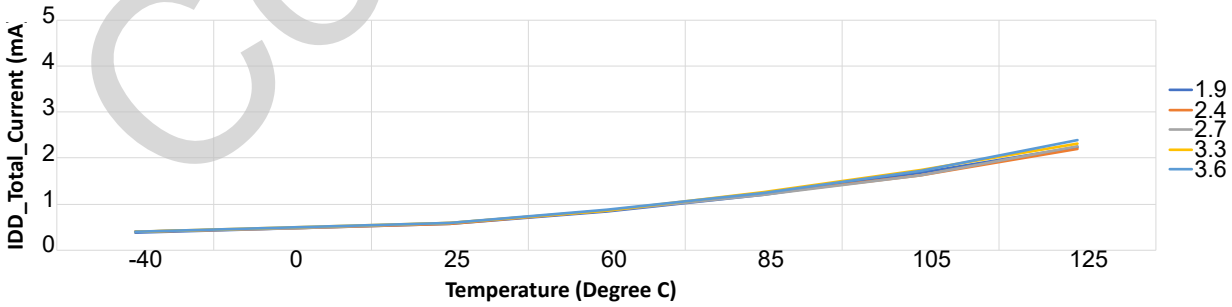


Figure 38-6. Sleep Current with XTAL_OFF_DC_DC

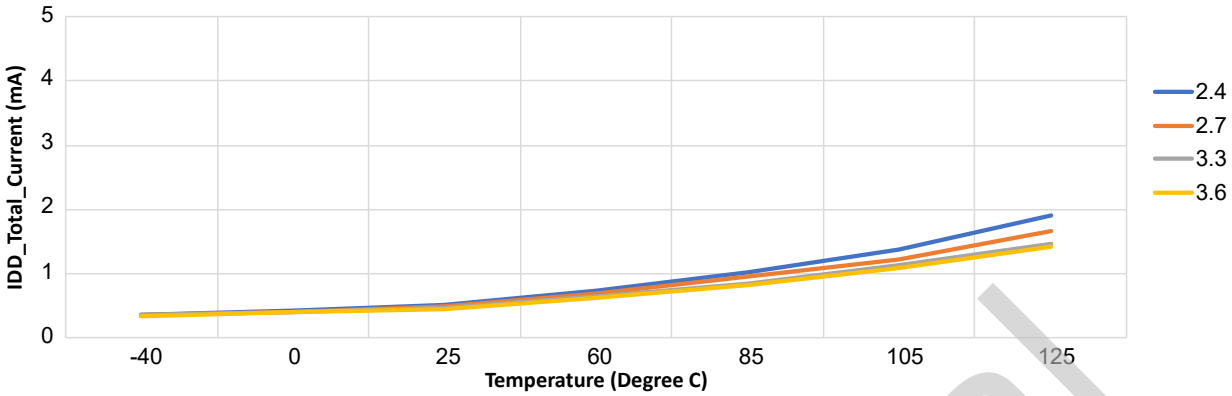


Figure 38-7. Sleep Current with XTAL_ON_MLDO

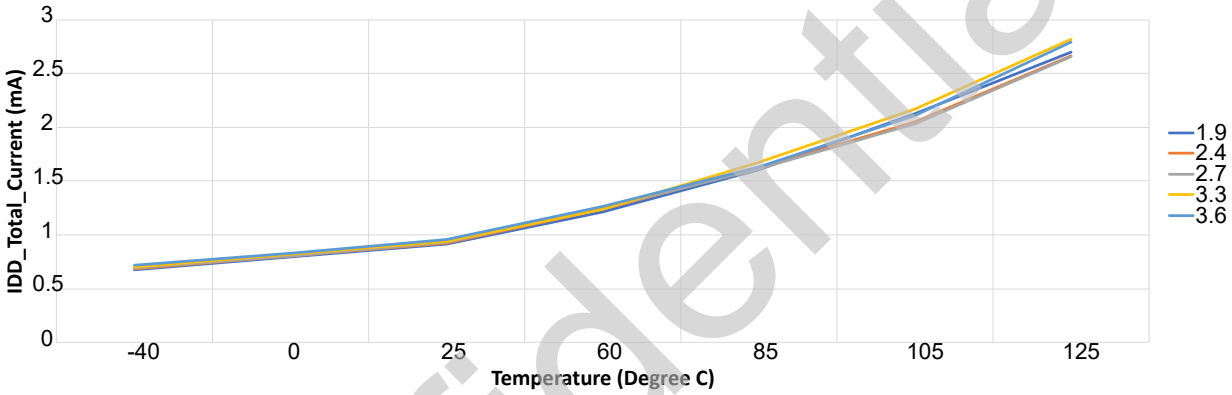
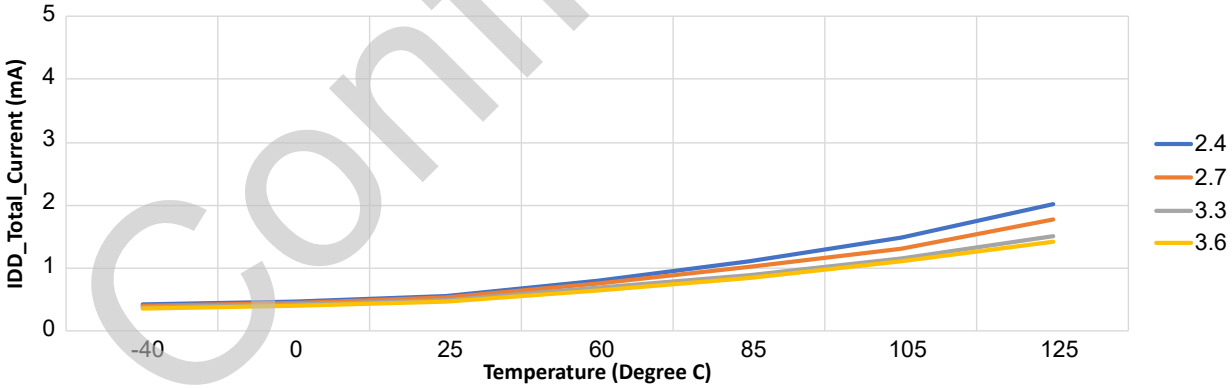


Figure 38-8. Sleep Current with XTAL_ON_DC_DC



Notes:

1. Typical value measured during characterization at 3.3V and 25℃.
2. The test conditions are as follows:
   - All GPIO except RPA3 are pulled up. RPA3 is pulled down.
   - All peripherals are disabled (except EIC).
   - All PB clocks are divided by 16.
   - LPRC is set as LPCLK.
   - SOSC is disabled.
   - CLDO is configured at lowest possible voltage (VREG Trim = 0x07).
   - PMU is configured to the Buck PSM mode on the Sleep Mode Entry with current sense is disabled.
   - Cache is enabled and configured to wait states PFMWS[3:0] as 0xF.
   - Backup RAM in the Retention mode.
   - DSU is disconnected.
   - RF system is in low power configuration.
   - Entry to the Sleep mode is enabled and WFI instruction is executed.
   - In XTAL ON mode - PMU Buck clock is derived from POSC 16 MHz and scaled to 1 MHz. FRC is OFF.
   - In XTAL OFF mode - PMU is clocked from FRC and XTAL 16 MHz clock is disabled and clock configuration in CRU changed to FRC.

38.7 Deep Sleep Current Consumption DC Electrical Specifications

Table 38-7. Deep Sleep Current Consumption DC Electrical Specifications

DC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**V_{DDIO}**	**Typ(1)**	**Max.**	**Units**	**Conditions**
BPWR_1	I_{DD_BACKUP}(2)	MCU I_{DD} in Deep Sleep mode powered from V_{DDIO}	3.3V	1.2	—	µA	No backup RAM retained
BPWR_2			3.3V	1.5	—	µA	8 KB backup RAM retained

Figure 38-9. Deep Sleep Current RAM ON

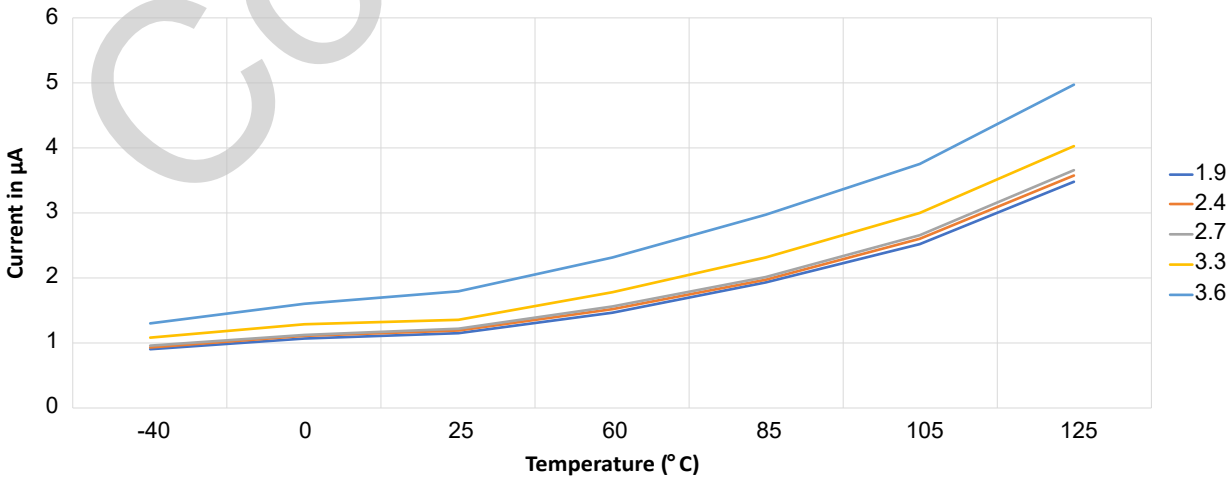
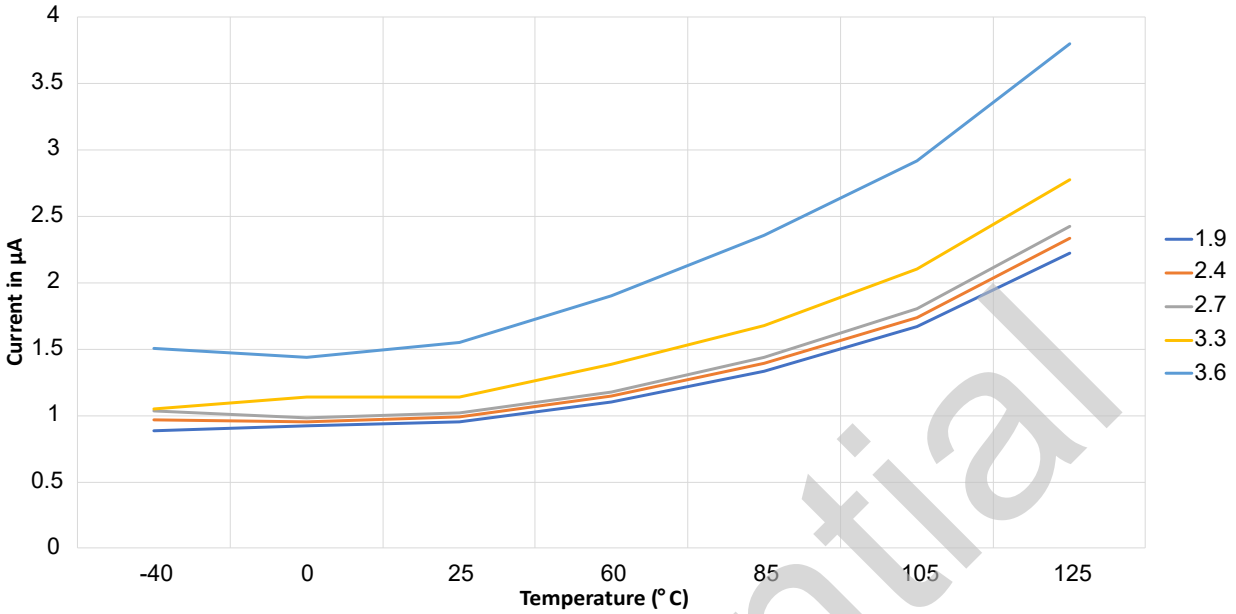


Figure 38-10. Deep Sleep Current RAM OFF



Notes:

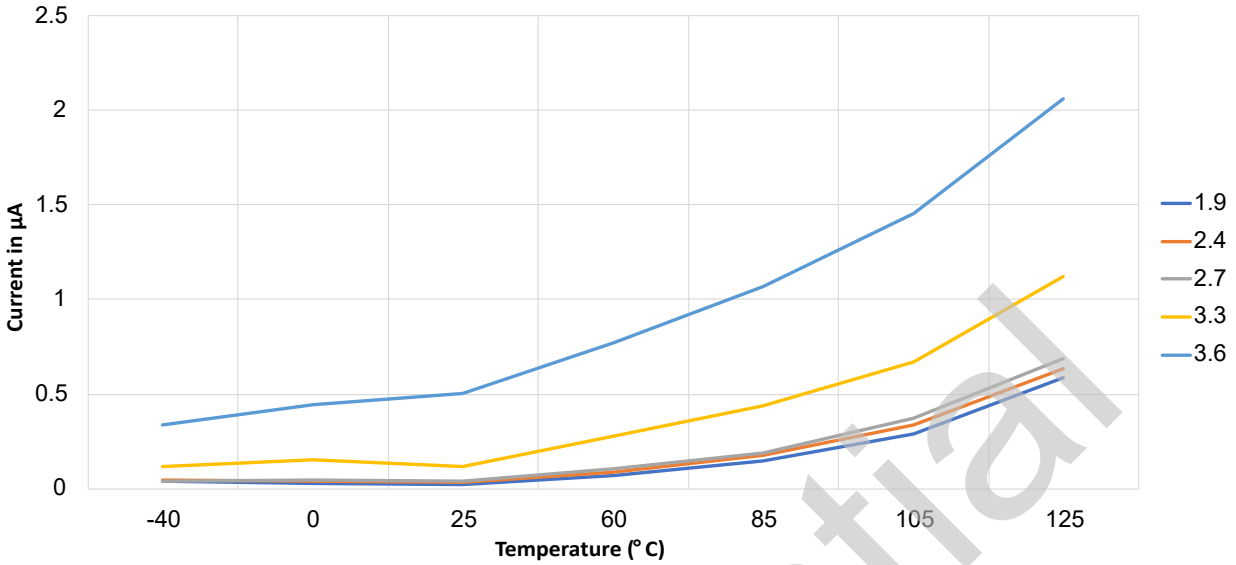
1. Typical value measured during characterization at 3.3V and 25°C.
2. The test conditions are as follows:
   - All GPIO except RPA3 are pulled up. RPA3 is pulled down.
   - All peripherals are disabled.
   - All PB clocks are divided by 16.
   - LPRC is set as LPCLK.
   - SOSC and POSC is disabled.
   - DSU is disconnected.
   - RF system is in low power configuration.
   - DSWDT is enabled and configured for wake-up.
   - Deep sleep entry is configured and WFI instruction is executed.

38.8 XDS (Extreme Deep Sleep) Current Consumption DC Electrical Specifications

Table 38-8. XDS (Extreme Deep Sleep) Current Consumption DC Electrical Specifications

DC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**V_{DDIO}**	**Typ(1)**	**Max.**	**Units**	**Conditions**
OPWR_1	I_{DD_OFF}(2)	MCU I_{DD} in OFF mode powered from V_{DDIOx}	3.3V	0.12	—	µA	In OFF mode, the device is entirely powered-off. **Note:** This mode is left by pulling the RESET pin low, or when a power Reset is done.

Figure 38-11. Extreme Deep Sleep Current



Notes:

1. Typical value measured during characterization at 3.3V and 25°C.
2. The test conditions are as follows:
   - All GPIO except RPA3 are pulled up. RPA3 is pulled down.
   - All peripherals are disabled.
   - DSU is disconnected.
   - RF system is in low power configuration.
   - DSWDT is disabled.
   - RTCC and POSC is disabled.
   - Deep sleep entry is configured and WFI instruction is executed.

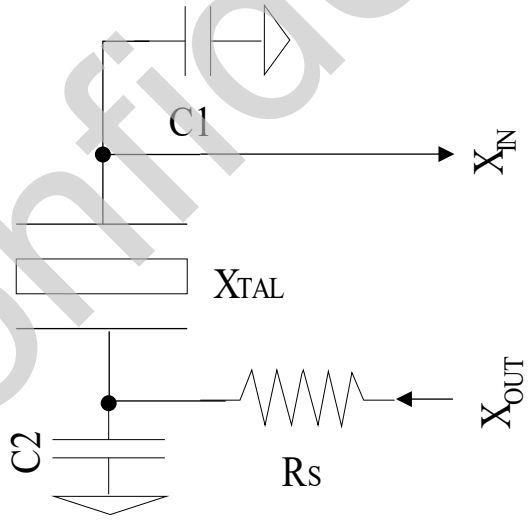
38.9 External XTAL and Clock AC Electrical Specifications

Table 38-9. External XTAL and Clock AC Electrical Specifications

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ**	**Max.**	**Units**	**Conditions(1)**
XOSC_1	FOSC_XOSC	XOSC crystal frequency	—	16	—	MHz	—
XOSC_1A	TOSC	TOSC = 1/ FOSC_XOSC	—	0.0625	—	ns	See parameter XOSC1 for FOSC_XOSC value
XOSC_2	XOSC_ST(2)	XOSC crystal start-up time	—	—	2.5	ms	Crystal stabilization time only not oscillator ready

.....continued							
AC Characteristics			**Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp**				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ**	**Max.**	**Units**	**Conditions(1)**
XOSC_3	CXIN	XOSC XIN parasitic pin capacitance	—	0.35	—	pF	With default crystal trim settings(2)
XOSC_5	CXOUT	XOSC XOUT parasitic pin capacitance	—	0.35	—	pF	With default crystal trim settings
XOSC_11	CLOAD(3)	XOSC crystal FOSC = 16 MHz	—	9	—	pF	—
XOSC_21	ESR	XOSC crystal FOSC = 16 MHz	—	—	100	Ω	—
XOSC_33	DLEVEL	MCU crystal oscillator power drive level	—	—	100	µW	—
XOSC_34	FREQ ERROR	Center frequency error	-30	—	+30	ppm	Across temperature

Figure 38-12. External XTAL and Clock Diagram



Notes:

1. V_{DDIO} = V_{DDANA} = 3.3V.
2. Refer RDP for information related to crystal trimming.
3. This is for guidance only. A major component of crystal start-up time is based on the second party crystal MFG parasitic that are outside the scope of this specification.
4. The test conditions for crystal load capacitor calculation are as follows:
   – Standard PCB trace capacitance = 1.5 pF per 12.5 mm (0.5 inches) (in other words, PCB STD TRACE W = 0.175 mm, H = 36 µm, T = 113 µm).
   – Xtal PCB capacitance typical; therefore, ~= 2.5 pF for a tight PCB xtal layout
   – For CXIN and CXOUT within 4 pF of each other, assume CXTAL_EFF = ((CXIN+CXOUT) / 2).
   – **Note:** Averaging CXIN and CXOUT will affect the final calculated CLOAD value by less than 0.25 pF.

Equation 38-1. Equation 1:

$$MFG\ \ CLOAD\ Spec = \{([CXIN + C1] * [CXOUT + C2])/[CXIN + C1 + C2 + CXOUT]\} + estimated\ oscillator\ PCB\ stray\ capacitance$$

Assuming C1 = C2 and CXin ~= CXout, the formula can be further simplified and restated to solve for C1 and C2 by:

Equation 38-2. Equation 2 (In other words: Simplified Equation 1)

$$C1 = C2 = ((2 * MFG\ \ CLOAD\ Spec) - CXTAL\_EFF - (2 * PCB\ \ capacitance))$$

Example:

- XTAL Mfg CLOAD Data Sheet Spec = 12 pF
- PCB XTAL trace Capacitance = 2.5 pF
- CXIN pin = 6.5 pF, CXOUT pin = 4.5 pF; therefore, CXTAL_EFF = ((CXIN+CXOUT) / 2)

CXTAL_EFF = ((6.5 + 4.5)/2) = 5.5 pF

C1 = C2 = ((2 * MFG Cload spec) - CXTAL_EFF - (2 * PCB capacitance))

C1 = C2 = (24 - 5.5 - (2 * 2.5))

C1 = C2 = 13.5 pF (Always rounded down)

C1 = C2 = 13 pF (in other words, for hypothetical example crystal external load capacitors)

User C1 = C2 = 13 pF CLOAD (max) spec

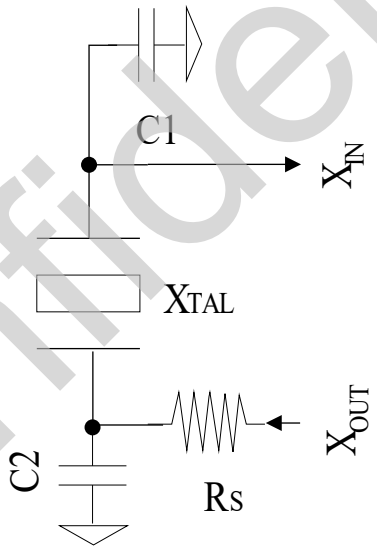
38.10 XOSC32 AC Electrical Specifications

Table 38-10. XOSC32 AC Electrical Specifications

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions(1)**
XOSC32_1	FOSC_XOSC32	XOSC32 oscillator crystal frequency	—	32.768	—	kHz	XIN32, XOUT32 secondary oscillator
XOSC32_3	CXIN32	XOSC32 XIN32 parasitic pin capacitance	—	0.4 2.4	—	pF	At the SOC pins on the Module
XOSC32_5	CXOUT32	XOSC32 XOUT32 parasitic pin capacitance on PIC32CX-BZ2	—	0.4 2.4	—	pF	At the SOC pins on the Module

.....continued							
AC Characteristics			**Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp**				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions(1)**
XOSC32_11	CLOAD_X32(3)	32.768 kHz crystal load capacitance	—	11	—	pF	—
XOSC32_13	ESR_X32	32.768 kHz crystal ESR	—	75	—	KΩ	—
XOSC32_15	TOSC32	TOSC32 = 1/ FOSC_XOSC32	—	30.518	—	µs	See parameter XOSC32_1 for FOSC_XOSC32 value
XOSC32_17	XOSC32_ST(2)	XOSC32 crystal start-up time	—	23	—	ms	This includes system delay and cannot be considered as the exact start-up time of SOSC clock as it is brought out on REFOx

Figure 38-13. XOSC32 Block Diagram



Notes:

1. V_{DDIO} = V_{DDANA} = 3.3V.
2. This is for guidance only. A major component of crystal start-up time is based on the second party crystal MFG parasitic that is outside the scope of this specification. If this is a major concern, the customer might need to characterize this based on their design choices.
3. The test conditions for crystal load capacitor calculation are as follows:
   – Standard PCB trace capacitance = 1.5 pF per 12.5 mm (0.5 inches) (in other words, PCB STD TRACE W = 0.175 mm, H = 36 µm, T = 113 µm)
   – Xtal PCB capacitance typical; therefore, ~= 2.5 pF for a tight PCB xtal layout
   – For CXIN and CXOUT within 4 pF of each other, assume CXTAL_EFF = ((CXIN / 2)
   – **Note:** Averaging CXIN and CXOUT will affect the final calculated CLOAD value by less than the tolerance of the capacitor selection.
4. **Equation 38-3. Equation 1:**

   $$MFG\ \ CLOAD\ Spec = \{([CXIN + C1] * [CXOUT + C2])/[CXIN + C1 + C2 + CXOUT]\} + estimated\ oscillator\ PCB\ stray\ capacitance$$

   Assuming C1 = C2 and CXin ~= CXout, the formula can be further simplified and restated to solve for C1 and C2 by:

   Equation 38-4. Equation 2 (In other words: Simplified Equation 1)

   $$C1 = C2 = ((2 * MFG\ \ CLOAD\ Spec) - CXTAL\_EFF - (2 * PCB\ \ capacitance))$$

   Example:

   - XTAL Mfg CLOAD Data Sheet Spec = 12 pF
   - PCB XTAL trace Capacitance = 2.5 pF
   - CXIN pin = 6.5 pF, CXOUT pin = 4.5 pF therefore CXTAL_EFF = ((CXIN / 2)

   CXTAL_EFF = ((6.5 + 4.5)/2) = 5.5 pF

   C1 = C2 = ((2 * MFG Cload spec) - CXTAL_EFF - (2 * PCB capacitance))

   C1 = C2 = (24 - 5.5 - (2 * 2.5))

   C1 = C2 = (24 - 5.5 - 5)

   C1 = C2 = 13.5 pF (Always rounded down)

   C1 = C2 = 13 pF (in other words, for hypothetical example crystal external load capacitors)

   User C1 = C2 = 13 pF ≤ CLOAD_X32 (max.) spec

38.11 Low Power Internal 32 kHz RC Oscillator AC Electrical Specifications

Table 38-11. Low Power Internal 32 kHz RC Oscillator AC Electrical Specifications

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
LP32K_1	FOSC_LPRC32K	Output frequency	—	32.7	—	kHz	Factory default calibration
LP32K_3	LPRC32K_ACC	Accuracy	—	—	—	kHz	—
LP32K_9	RC32K_Duty	LPRC32K OSC duty cycle	—	50	—	%	—

38.12 FRC AC Electrical Specifications

Table 38-12. FRC AC Electrical Specifications

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	Symbol	Characteristics	Min.	Typ.	Max.	Units	Conditions
FRC1	FINTFREQ	Internal FRC frequency	—	8	—	MHz	Factory calibrated and user trim bits set to zero
FRC2	TSUFRC	Start-up time of internal FRC	—	15	—	µs	Not characterized
FRC3	FACCU	FRC accuracy	—	—	—	%	—
FRC5	DUTY_CYCLE	FRC duty cycle	—	50	—	%	—
FRC6	C_USE	FRC user tune frequency drift	—	—	—	%	Frequency drift = [(Maximum frequency measured - Minimum frequency measured)/ (Default frequency of 8 MHz)] * 100; Maximum frequency drift possible by using the frequency trim bits
FRC7	C_USER_STEP_SIZE	FRC user tune step size	—	—	—	%	Step size = (% Drift across Osc tune)/(Total number of trim bit combinations); Change in frequency with incremental trim bit

38.13 Frequency AC Electrical Specifications

Table 38-13. Frequency AC Electrical Specifications

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	Symbol	Characteristics	Min.	Typ.	Max.	Units	Conditions
MCU_1	FCY	MCU frequency of operation	—	—	64	MHz	$V_{DDIO(min)}$ to $V_{DDIO(max)}$
MCU_3	TCY	MCU clock period	1/FCY			µs	

PLL (Phase Locked Loop) AC Electrical Specifications

Table 38-14. PLL MHz (Phase Locked Loop)

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	Symbol	Characteristics	Min.	Typ.	Max.	Units	Conditions
PLL_1	PLL_FIN	PLL input frequency	—	16	—	MHz	—
PLL_3	PLL_FOUT	PLL output clock frequency	—	96	—	MHz	—
PLL_4	PLL_VCO_FREQ	PLL VCO frequency (Fixed)	—	480	—	MHz	—

Note:

1. PLL input clock is 16 MHz XOSC.

38.14 QSPI Module Electrical Specifications

Table 38-15. QSPI Module Electrical Specifications

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	Symbol	Characteristics	Min.	Typ.(1)	Max.	Units	Conditions
QSPI_1	FCLK	SQI serial clock frequency • SDR Host mode 0 and 2	—	—	32	MHz	—
QSPI_2	FCLK	SQI serial clock frequency • SDR Host mode 1 and 3	—	—	32	MHz	—
QSPI_3	TSCKH	Serial clock high time	—	8.9	—	ns	—
QSPI_5	TSCKL	Serial clock low time	—	7.8	—	ns	—
QSPI_7	TSCKR	Serial clock rise time	—	6.9	—	ns	—
QSPI_9	TSCKF	Serial clock fall time	—	7.6	—	ns	—
QSPI_11	TCSS	CS active setup time	—	7.3	—	ns	—
QSPI_13	TCSH	CS active hold time	—	10.4	—	ns	—
QSPI_19	TDIS	Data in setup time	—	7.4	—	ns	—

.....continued							
AC Characteristics			**Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp**				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.(1)**	**Max.**	**Units**	**Conditions**
QSPI_21	TDIH	Data in hold time	—	19.8	—	ns	—
QSPI_23	TDOH	Data out hold	—	18	—	ns	—
QSPI_25	TDOV	Data out valid	—	2.4	—	ns	—

Note:

1. V_{DDIO} = 3.3V, C_{LOAD} = 30 pF (Typ.).

38.15 Power Supply DC Module Electrical Specifications

Table 38-16. Power Supply DC Electrical Specifications

AC Characteristics			**Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp**				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
REG_1	VDDCORE_CIN	VDDCORE (CLDO_OUT) input bypass parallel capacitor pair(5)	—	1	—	µF	Bulk ceramic or solid tantalum with ESR <0.5Ω. Min and max represent absolute values including cap tolerances
			—	100	—	nF	Ceramic XR7/X5R with ESR <0.5Ω depending on temperature
REG_5	VDD33	VDD33 input bypass parallel capacitor pair(5)	—	10	—	µF	Bulk ceramic or solid tantalum with ESR <0.5Ω(6)
			—	100	—	nF	Ceramic XR7/X5R with ESR <0.5Ω depending on temperature on all VDDIO pins(6)
REG_6	PMU_VDDIO	Input bypass parallel capacitor pair for the PMU power section(5)	—	4.7	—	µF	Bulk Ceramic or solid Tantalum with ESR <0.5Ω(6)
			—	—	—	nF	Ceramic XR7/X5R with ESR <0.5Ω depending on temperature on all VDDIO pins(6)
REG_7	PMU_VDDP	Input bypass parallel capacitor pair for the PMU power section(5)	—	1	—	µF	Bulk ceramic or solid tantalum with ESR <0.5Ω(6)
			—	—	—	nF	Ceramic XR7/X5R with ESR <0.5Ω depending on temperature on all VDDIN pins(6)

.....continued

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
REG_9	VDDFLASH_C_{IN}	VDD_FLASH bypass parallel capacitor pair(5)	—	10	—	µF	Bulk ceramic or solid tantalum with ESR <0.5Ω(6)
			—	100	—	nF	Ceramic XR7/X5R with ESR <0.5Ω depending on temperature on all VDDFLASH pins(6)
REG_17	VDDANA_C_{IN}	VDDANA input bypass parallel capacitor pair(5)	—	10	—	µF	Bulk ceramic or solid tantalum with ESR <0.5Ω(6)
			—	0.1	—	nF	Ceramic XR7/X5R with ESR <0.5Ω
REG_18	VDDANA_LEXT	VDDANA series ferrite bead DCR (DC resistance)	—	—	0.1	Ω	≥600Ω at 100 MHz
REG_19		Ferrite bead current Rating	100	—	—	mA	—
REG_20	BUCK_PLL_CIN	VDD bypass capacitor on the BUCK_PLL input	—	1	—	µF	Ceramic XR7/X5R with ESR <0.5Ω
REG_21	BUCK_BB_CIN	VDD bypass capacitor on the BUCK_BB input	—	1	—	µF	Ceramic XR7 with ESR <0.5Ω
REG_22	BUCK_MPA_CIN	VDD bypass capacitor on the BUCK_LPA input	—	1	—	µF	Ceramic XR7 with ESR <0.5Ω
REG_23	BUCK_LPA_CIN	VDD bypass capacitor on the BUCK_MPA input	—	1	—	µF	Ceramic XR7 with ESR <0.5Ω
REG_24	BUCK_CLDO_CIN	VDD bypass capacitor on the BUCK_CLDO input	—	1	—	µF	Ceramic XR7 with ESR <0.5Ω
REG_25	R_EXT	Bias for the reference current generation	—	30	—	kΩ	—

.....continued

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
REG_27	VSW_L_{EXT}(2,3)	Buck Switch mode regulator inductor inductance	—	4.7	—	µH	Shielded inductor only
REG_29		Inductor DCR (DC resistance)	—	—	0.22	Ω	(7)
REG_31		Inductor I_{SAT} Rating(2)	250	—	—	mA	—
REG_32	VSW_CAP EXT	Buck Switch mode regulator bulk capacitor capacitance	10	—	—	µF	—
REG_32A		Buck Switch mode regulator filtering capacitor capacitance	100	—	—	nF	—
REG_36	VDDCORE	VDDCORE voltage range	—	1.2	—	V	MCU Active, cache and prefetch disabled while executing from Flash(1)
REG_37	VDD33(4)	VDD33 input voltage range	1.9	3.3	3.6	V	—
REG_39	VDDANA(4)	VDDANA input voltage range	1.9	3.3	3.6	V	—
REG_40	VDDREG	VDDREG input voltage range	—	1.35	—	V	PMU output voltage
REG_43	SVDDIO_R	VDDIO rise ramp rate to ensure internal Power-on Reset signal	—	0.1	—	V/µs	Failure to meet this specification may lead to start-up or unexpected behaviors
REG_44	SVDDIO_F	VDDIO falling ramp rate to ensure internal Power-on Reset signal	—	—	—	V/µs	Failure to meet this specification may cause the device to not detect reset
REG_45	VP0R+	Power-on Reset	—	1.5	—	V	VDDIO power up/Down (See Param REG43, VDDIO Ramp Rate)
REG_45_A	VP0R-	Power-on Reset	—	1.5	—	V	VDDIO Power up/Down (See Param REG43, VDDIO Ramp Rate)
REG_47	VBOR33(4)	VDDIO BOD (All modes)(4)	—	1.7	—	V	—

.....continued

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
REG_48	VBOR12	BOR of the 1.2V regulator	—	1.0	—	V	—
REG_48A	VZBPOR33	Zero power BOR	—	1.8	—	V	—
REG_53	TRST(6)	External RESET valid active pulse width	—	11	—	µs	Minimum Reset active time to guarantee MCU Reset for the module. Reset filter circuit inside Module
			—	2.5	—	µs	Minimum Reset active time to guarantee MCU Reset for SoC with no Reset filter circuit
REG_54	MLDO_DELAY	PMU MLDO and CLDO power-up delay	—	260	—	µs	—
REG_55	SYS_DELAY	System delay before fetching first instruction	—	—	—	µs	—
REG_56	CLK_DELAY	Crystal start-up delay	—	XOSC_2	—	µs	Refer to XOSC_2 parameter from the crystal oscillator section
REG_57	POWER_ON_DELAY	Power-up period	—	—	—	µs	—
REG_58	BOR_DELAY	Width of the BOR event	—	488	—	µs	Includes system delay since this is measured with a CPU event

Figure 38-14. Power-On Reset Timing Characteristics

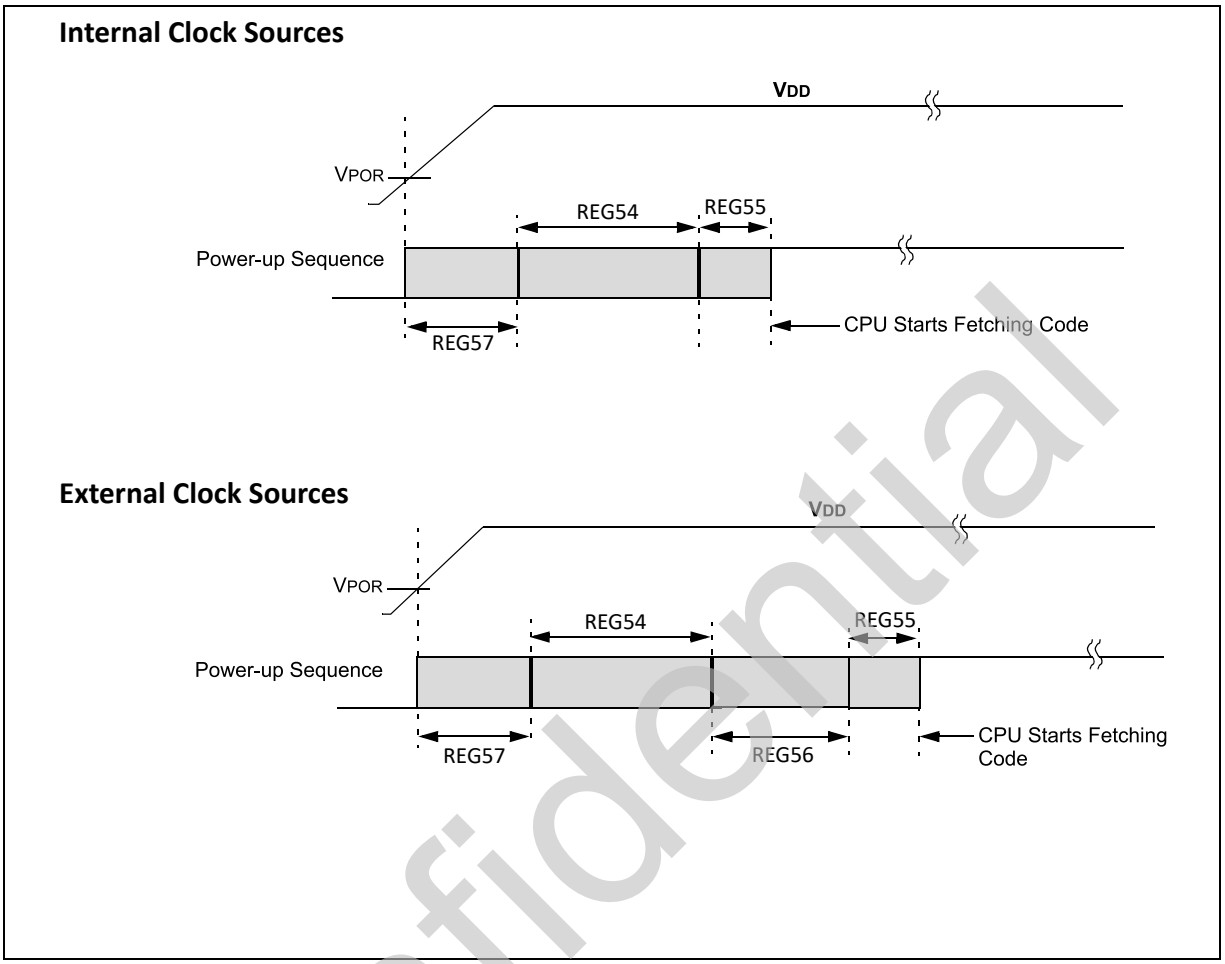
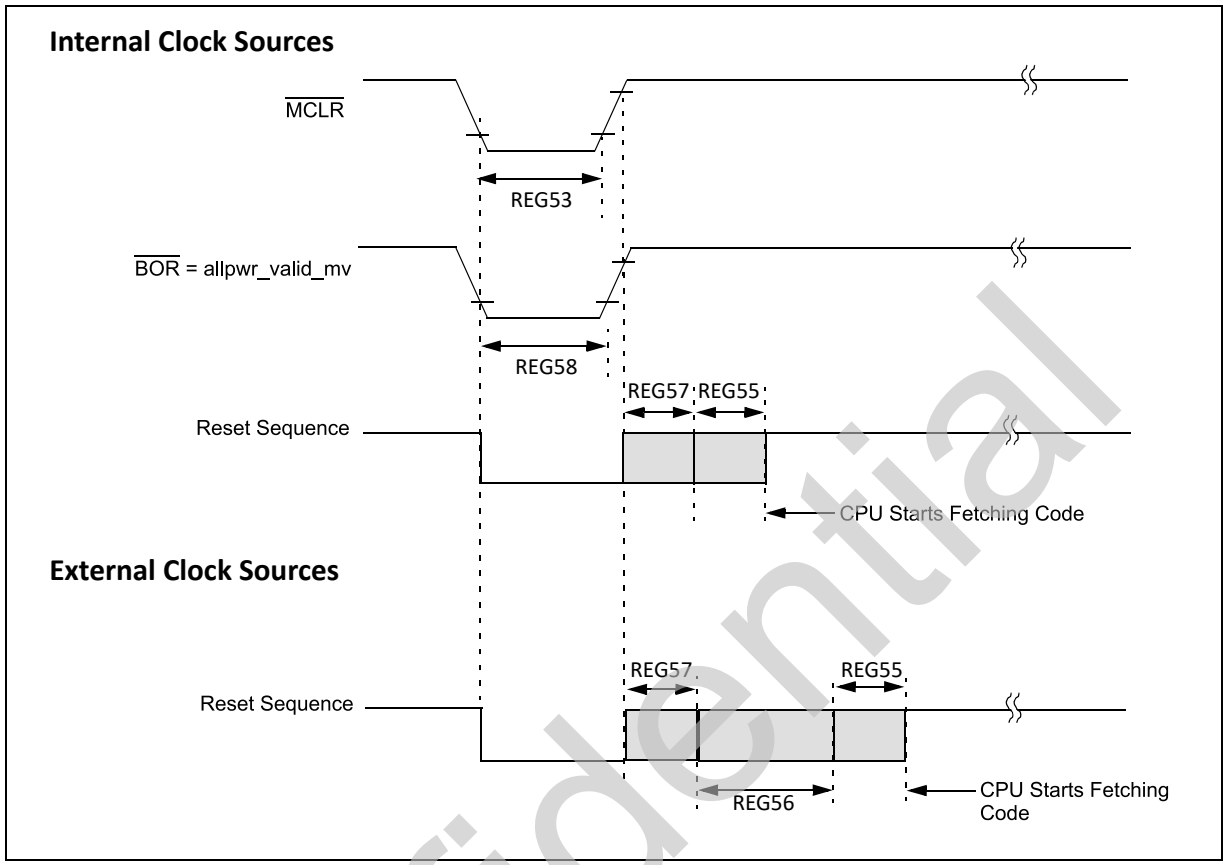


Figure 38-15. External Reset Timing Characteristics



Notes:

1. Ferrite Bead ISAT(min) ≥ (IDDANA(max) * 1.15).
2. Buck Inductor ISAT(min) ≥ ((ICAPACITOR + IVDDCORE_MAX) * 1.2) when the BUCK mode is enabled (shielded inductor only).
3. User must select either LDO or BUCK Mode. The modes are exclusive to each other.
4. VDD33 and VDDANA must be at the same voltage level.
5. All bypass caps must be located immediately adjacent to pin(s) and on the same side of the PCB as the MCU. Each primary power supply group VDDIO, VDDANA, VDDCORE must have one bulk capacitor and all power pins with a 100 nF bypass cap.
6. The RESET pulse width is the minimum pulse width required on the I/O pin after any filtering on the NMCLR pin.
7. Keep the DCR as low as possible to improve efficiency.

38.16 I/O PIN AC/DC Electrical Specifications

Table 38-17. I/O PIN AC/DC Electrical Specifications

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.(1)**	**Max.**	**Units**	**Conditions**
DI_1	VOL	Output voltage low (Drive strength, 8x)	—	0.1	—	V	VDDIO = 3.3V at IOL= 10 mA
		Output voltage low (Drive strength, 4x)		0.2	—		VDDIO = 3.3V at IOL= 10 mA
DI_2	VOL	Output voltage low (Drive strength, 8x)	—	0.2	—	V	VDDIO = 3.3V at IOL= 13 mA
		Output voltage low (Drive strength, 4x)		0.3	—		VDDIO = 3.3V at IOL= 13 mA
DI_3	VOL	Output voltage low (Drive strength, 8x)	—	0.2	—	V	VDDIO = 3.3V at IOL= 15 mA
DI_4	VOH	Output voltage low (Drive strength, 8x)	—	2.3	—	V	VDDIO = 3.3V at IOH= 5 mA
		Output voltage low (Drive strength, 4x)	—	2.1	—		VDDIO = 3.3V at IOH= 5 mA
DI_5	VOH	Output voltage low (Drive strength, 8x)	—	2.2	—	V	VDDIO = 3.3V at IOH= 7 mA
		Output voltage low (Drive strength, 4x)	—	2.0	—		VDDIO = 3.3V at IOH= 7 mA
DI_6	VOH	Output voltage High (Drive strength, 8x)	—	2.1	—	V	VDDIO = 3.3V at IOH= 10 mA
DI_7	VIL	Input voltage low (Drive strength, 8x)	—	0.7	—	V	VDDIO = 3.3V
		Input voltage low (Drive strength, 4x)	—	0.5	—		VDDIO = 3.3V

.....continued

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.(1)**	**Max.**	**Units**	**Conditions**
DI_8	VIH	Input voltage high (Drive strength, 8x)	—	2.9	—	V	VDDIO = 3.3V
		Input voltage high (Drive strength, 4x)	—	2.9	—		VDDIO = 3.3V
DI_13	IIL	Input pin leakage current	—	71.8	—	nA	—
DI_15(5)	RPDWN	Internal pull-down (Drive strength, 8x)	—	21.4	—	kΩ	VDDIO = 3.3V
		Internal pull-down (Drive strength, 4x)	—	3.9	—	kΩ	
DI_17	RPUP	Internal pull-up (Drive strength, 8x)	—	18.6	—	kΩ	
		Internal pull-up (Drive strength, 4x)	—	3.3	—	kΩ	
DI_25	TRISE	I/O pin rise time (High drive strength, high load)	—	1.97	—	ns	VDDIO = 3.3V
		I/O pin rise time (Low drive strength, high load)	—	10.7	—	ns	—
		I/O pin rise time (High drive strength, standard load)	—	2.0	—	ns	—
		I/O pin rise time (Low drive strength, standard load)	—	7.6	—	ns	—

.....continued							
AC Characteristics			**Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp**				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.(1)**	**Max.**	**Units**	**Conditions**
DI_27	TFALL	I/O pin fall time (High drive strength, high load)	—	1.6	—	ns	—
		I/O pin fall time (Low drive strength, high load)	—	8.1	—	ns	—
		I/O pin fall time (High drive strength, standard load)	—	1.6	—	ns	—
		I/O pin fall time (Low drive strength, standard load)	—	5.1	—	ns	—

Notes:

1. All measurements done at 3.3V at 25℃.
2. VIL source < (GND – 0.3). Characterized but not tested.
3. Any number and/or combination of I/O pins not excluded under IICL or IICH conditions are permitted provided the absolute instantaneous sum of the input injection currents from all pins do not exceed the specified ∑IICT limit. To limit the injection current, the user must insert a resistor in series RSERIES (in other words, RS), between the input source voltage and device pin. The resistor value is calculated according to:
   - For negative Input voltages less than (GND – 0.3): RS ≥ absolute value of | ((VIL source – (GND – 0.3)) / IICL) |
   - For positive input voltages greater than (VDDIO + 0.3): RS ≥ ((VIH source – (VDDIO +0.3))/ IICH)
   - For Vpin voltages greater than VDDIO + 0.3 and less than GND – 0.3: RS = the larger of the values calculated above
4. High load = 50 pF and Standard load = 30 pF.
5. The drive strength of pads are as follows:
   - 8x pads - PA0, PA1, PA3, PA4, PA5, PA6, PA7, PA8, PA9, PA10, PA13, PA14, PB10, PB11, PB12, PB13
   - 4x pads - PA2, PB0, PB1, PB2, PB3, PB4, PB5, PB6, PB7, PB8, PB9

38.17 I²C Module Electrical Specifications

Note: Traditional Serial Communication Interface documentation uses the terminology “Master” and “Slave”. The equivalent Microchip terminology used in this document is “Host” and “Client”, respectively.

Table 38-18. I²C Module Host Mode Electrical Specifications

AC Characteristics				Standard Operating Conditions: $V_{DDIO} = V_{DDANA}$ 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial				
Param. No.	**Symbol**	**Characteristics**		**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
I2CM_1	TL0:SCL	Host clock low time	100 kHz mode	—	5020	—	ns	VDDIOx = 3.3V,
			400 kHz mode	—	1280	—	ns	
			1 MHz mode	—	712	—	ns	VDDIOx = 3.3V, IPULL-UP = 20 mA, CLOAD = 550 pf
I2CM_5	TF:SCL	SDAx and SCLx fall time	100 kHz mode	—	19	—	ns	VDDIOx = 3.3V, IPULL-UP = 3 mA, CLOAD = 400 pF
			400 kHz mode	—	5	—	ns	
			1 MHz mode	—	5	—	ns	VDDIOx = 3.3V, IPULL-UP = 20 mA, CLOAD = 550 pF
I2CM_7	TR:SCL	SDAx and SCLx rise time	100 kHz mode	—	166	—	ns	VDDIOx = 3.3V, IPULL-UP = 3 mA, CLOAD = 400 pF
			400 kHz mode	—	154	—	ns	
			1 MHz mode	—	154	—	ns	VDDIOx = 3.3V, IPULL-UP = 20 mA, CLOAD = 550 pF
I2CM_9	TSU:DAT	Data input setup time	100 kHz mode	—	4900	—	ns	VDDIOx = 3.3V, IPULL-UP = 3 mA, CLOAD = 400 pF
			400 kHz mode	—	1200	—	ns	
			1 MHz mode	—	592	—	ns	VDDIOx = 3.3V, IPULL-UP = 20 mA, CLOAD = 550 pF
I2CM_11	THD:DAT	Data input hold time	100 kHz mode	—	159	—	ns	VDDIOx = 3.3V, IPULL-UP = 3 mA, CLOAD = 400 pF
			400 kHz mode	—	93	—	ns	
			1 MHz mode	—	140	—	ns	VDDIOx = 3.3V, IPULL-UP = 20 mA, CLOAD = 550 pF

.....continued

AC Characteristics				Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial				
Param. No.	Symbol	Characteristics		Min.	Typ.	Max.	Units	Conditions
I2CM_13	TSU:STA	Start condition setup time	100 kHz mode	—	5180	—	µs	VDDIOx = 3.3V, IPULL-UP = 3 mA, CLOAD = 400 pF
			400 kHz mode	—	1450	—	ns	
			1 MHz mode	—	867	—	ns	VDDIOx = 3.3V, IPULL-UP = 20 mA, CLOAD = 550 pF
I2CM_15	THD:STA	Start condition hold time	100 kHz mode	—	4990	—	µs	VDDIOx = 3.3V, IPULL-UP = 3 mA, CLOAD = 400 pF
			400 kHz mode	—	1250	—	µs	
			1 MHz mode	—	666	—	µs	VDDIOx = 3.3V, IPULL-UP = 20 mA, CLOAD = 550 pF
I2CM_21	TAA:SCL	Output valid from clock	100 kHz mode	—	—	—	ns	VDDIOx = 3.3V, IPULL-UP = 3 mA, CLOAD = 400 pF
			400 kHz mode	—	—	—	ns	
			1 MHz mode	—	—	—	ns	VDDIOx = 3.3V, IPULL-UP = 20 mA, CLOAD = 550 pF
I2CM_23	TBF:SDA	Bus free time(1)	100 kHz mode	—	TL0:SCL	—	ns	VDDIOx = 3.3V, IPULL-UP = 3 mA, CLOAD = 400 pF
			400 kHz mode	—	TL0:SCL	—	ns	
			1 MHz mode	—	TL0:SCL	—	ns	VDDIOx = 3.3V, IPULL-UP = 20 mA, CLOAD = 550 pF

Note:

1. The amount of time the bus must be free before a new transmission can start (STOP condition to START condition).

Figure 38-16. I²C Start/Stop Bits Host Mode AC Timing Diagram

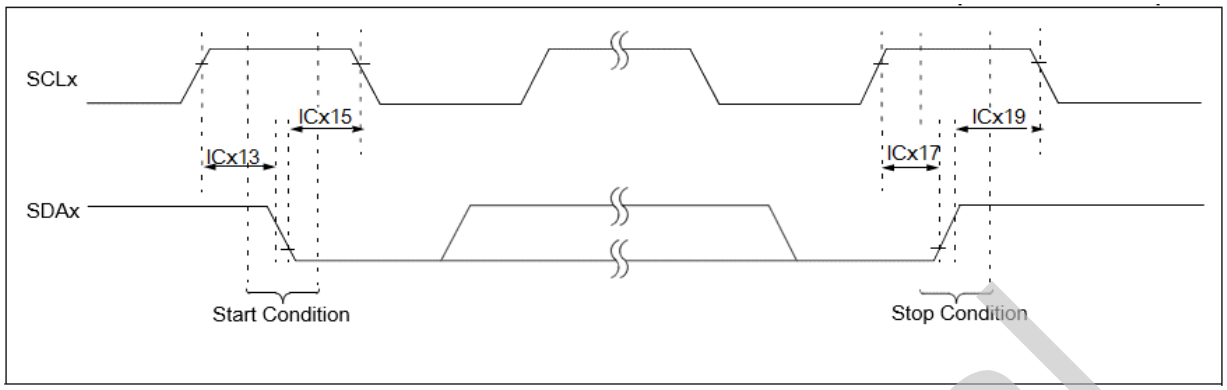


Figure 38-17. I²C Bus Data Host Mode AC Timing Diagram

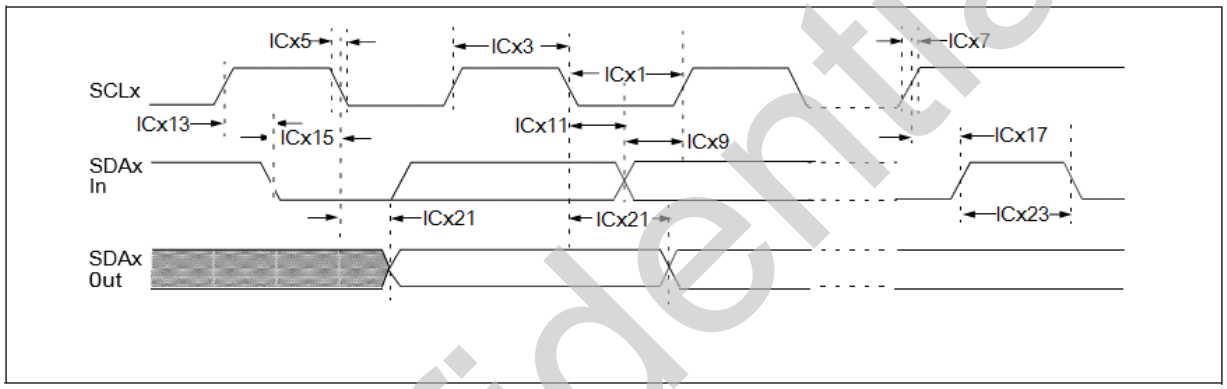


Table 38-19. I²C Module Client Mode Electrical Specifications

AC Characteristics				Standard Operating Conditions: $V_{DDIO} = V_{DDANA}$ 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended Temp				
Param. No.	**Symbol**	**Characteristics**		**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
I2CS_5	TF:SCL	SDAx and SCLx fall time	100 kHz mode	—	19	—	ns	VDDIOx = 3.3V, IPULL-UP = 3 mA, CLOAD = 400 pF
			400 kHz mode	—	5	—	ns	
			1 MHz mode	—	5	—	ns	VDDIOx = 3.3V, IPULL-UP = 20 mA, CLOAD = 550 pF
I2CS_7	TR:SCL	SDAx and SCLx rise time	100 kHz mode	—	166	—	ns	VDDIOx = 3.3V, IPULL-UP = 3 mA, CLOAD = 400 pF
			400 kHz mode	—	154	—	ns	
			1 MHz mode	—	154	—	ns	VDDIOx = 3.3V, IPULL-UP = 20 mA, CLOAD = 550 pF

.....continued

AC Characteristics				Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial -40°C ≤ T_A ≤ +125°C for Extended Temp				
Param. No.	**Symbol**	**Characteristics**		**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
I2CS_9	TSU:DAT	Data input setup time	100 kHz mode	—	5580	—	ns	VDDIOx = 3.3V, IPULL-UP = 3 mA, CLOAD = 400 pF
			400 kHz mode	—	1470	—	ns	
			1 MHz mode	—	—	—	ns	VDDIOx = 3.3V, IPULL-UP = 20 mA, CLOAD = 550 pF
I2CS_11	THD:DAT	Data input hold time	100 kHz mode	—	490	—	ns	VDDIOx = 3.3V, IPULL-UP = 3 mA, CLOAD = 400 pF
			400 kHz mode	—	490	—	ns	
			1 MHz mode	—	—	—	µs	VDDIOx = 3.3V, IPULL-UP = 20 mA, CLOAD = 550 pF

Note:

1. The amount of time the bus must be free before a new transmission can start (STOP condition to START condition).

Figure 38-18. I^2C Start/Stop Bits Client Mode AC Timing Diagram

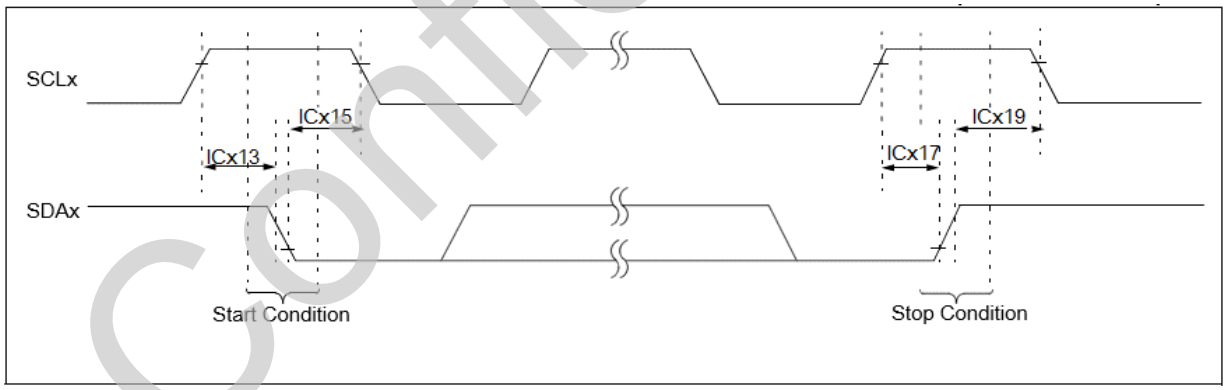
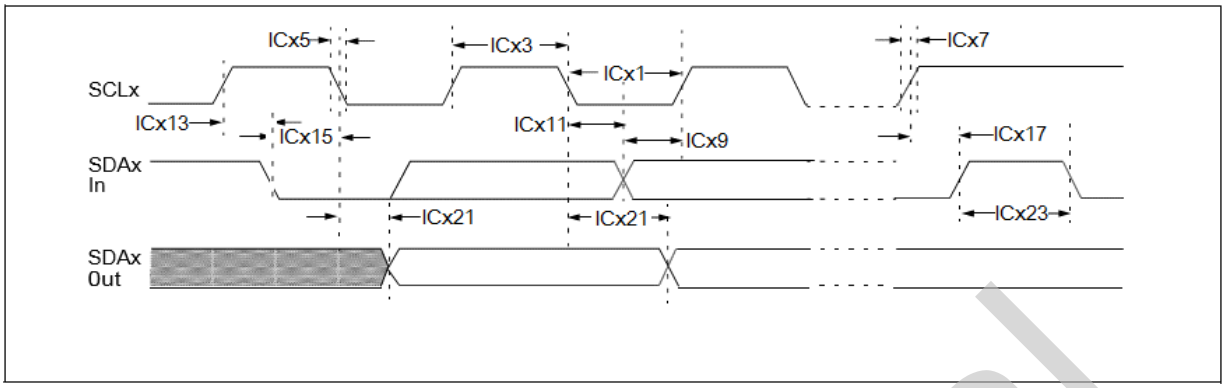


Figure 38-19. I²C Bus Data Client Mode AC Timing Diagram



38.18 SPI Module Electrical Specifications

Note: Traditional Serial Communication Interface documentation uses the terminology “Master” and “Slave”. The equivalent Microchip terminology used in this document is “Host” and “Client”, respectively.

Table 38-20. SPI Module Host Mode Electrical Specifications

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
MSP_1	FSCK	SCK frequency	—	32	—	MHz	Fixed pins
			—	24	—		Remappable pins
MSP_3	TSCL	SCK output low time	—	31.2	—	ns	—
MSP_5	TSCH	SCK output high time	—	31.2	—	ns	—
MSP_7	TSCF	SCK and MOSI output fall time	—	1.8	—	ns	See parameter DI27 I/O spec
MSP_9	TSCR	SCK and MOSI output rise time	—	1.8	—	ns	See parameter DI25 I/O spec
MSP_11	TMOV	MOSI Data output valid after SCK	—	11	—	ns	VDDIOx (Min.), CLOAD = 30 pF (Typ.)
MSP_13	TMOH	MOSI hold after SCK	—	15	—	ns	
MSP_15	TMIS	MISO setup time of data input to SCK	—	23	—	ns	
MSP_17	TMIH	MISO hold time of data input to SCK	—	8	—	ns	

Figure 38-20. SPI Host Module CPHA=0 Timing Diagrams

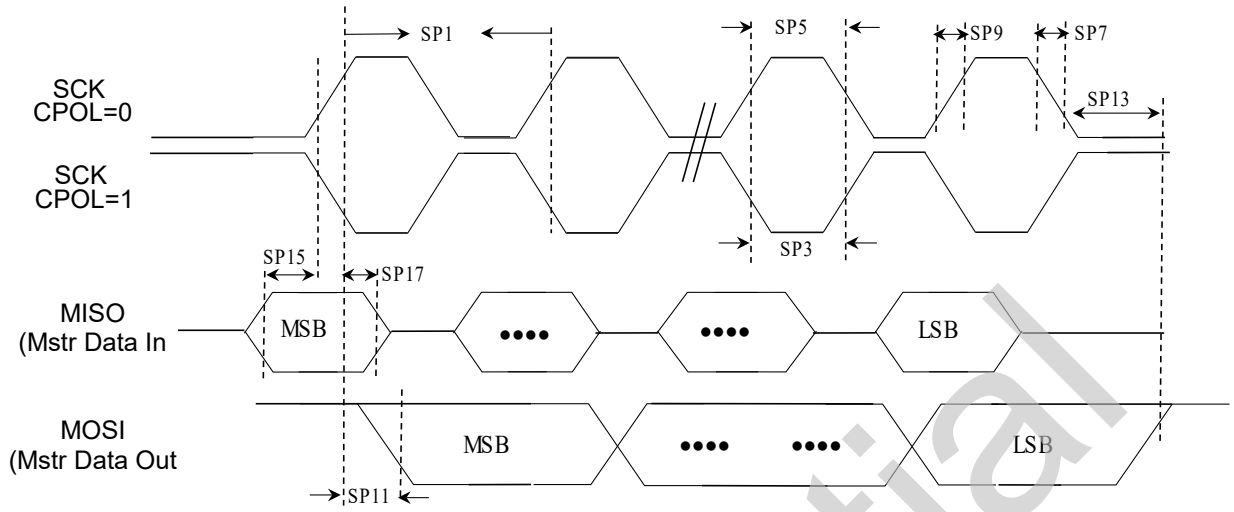
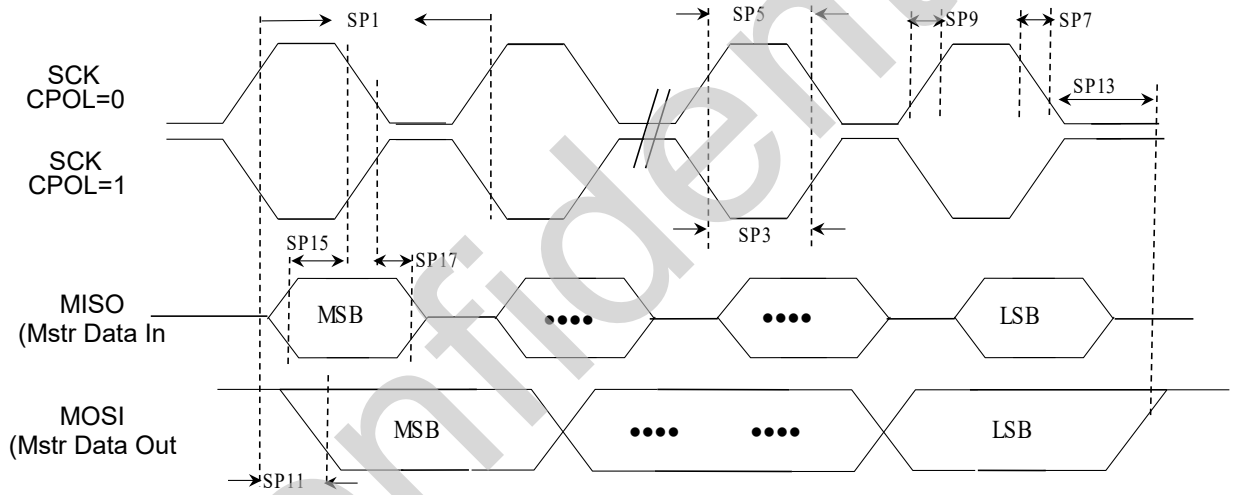


Figure 38-21. SPI Host Module CPHA=1 Timing Diagrams



Note:

1. Assumes VDDIOx(min) and 30 pF external load on all SPIx pins unless otherwise noted.

Table 38-21. SPI Module Client Mode Electrical Specifications

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ**	**Max.**	**Units**	**Conditions**
SSP_1	FSCK	SCK frequency	—	16	16	MHz	VDDIOx = 1.9V or VDDIO(min) whichever is greater, CLOAD = 30 pF (Min) fixed pins
			—	12	12		Remappable pins
SSP_3	TSCL	SCK output low time	—	62.5	—	ns	—

.....continued							
AC Characteristics			**Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp**				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ**	**Max.**	**Units**	**Conditions**
SSP_5	TSCH	SCK output high time	—	62.5	—	ns	—
SSP_7	TSCF	SCK and MOSI output fall time	—	3	—	ns	See parameter DI27 I/O spec
SSP_9	TSCR	SCK and MOSI output rise time	—	26	—	ns	See parameter DI25 I/O spec
SSP_11	TSOV	MOSI data output valid after SCK	—	21	—	ns	VDDIOx = 3.3V, CLOAD = 30 pF (Min)
SSP_15	TSIS	MISO setup time of data input to SCK	—	23	—	ns	
SSP_17	TSIH	MISO hold time of data input to SCK	—	22	—	ns	

Figure 38-22. SPI Client Module CPHA=0 Timing Diagrams

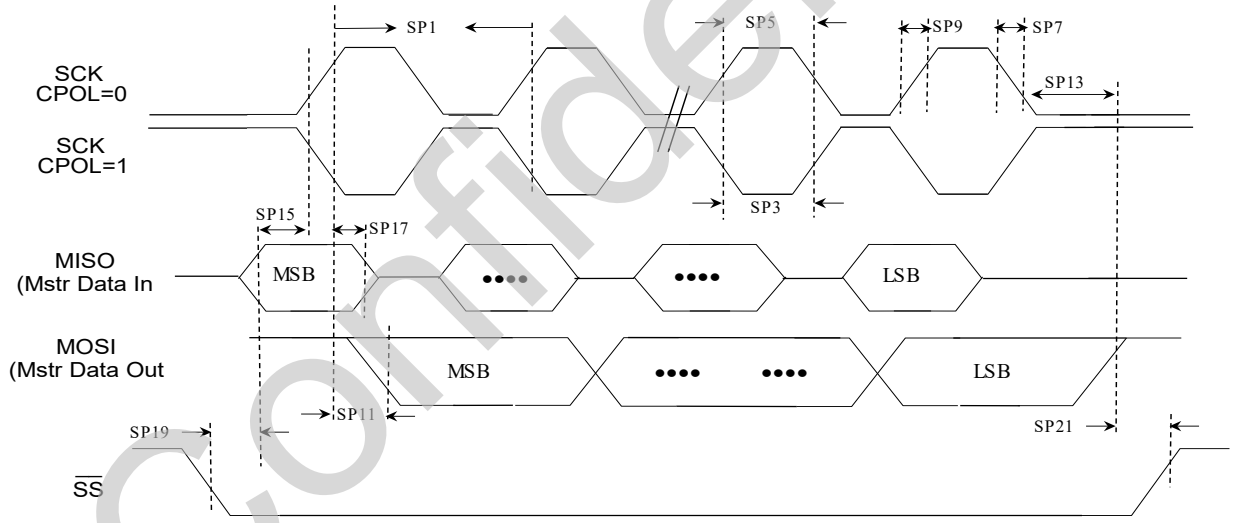
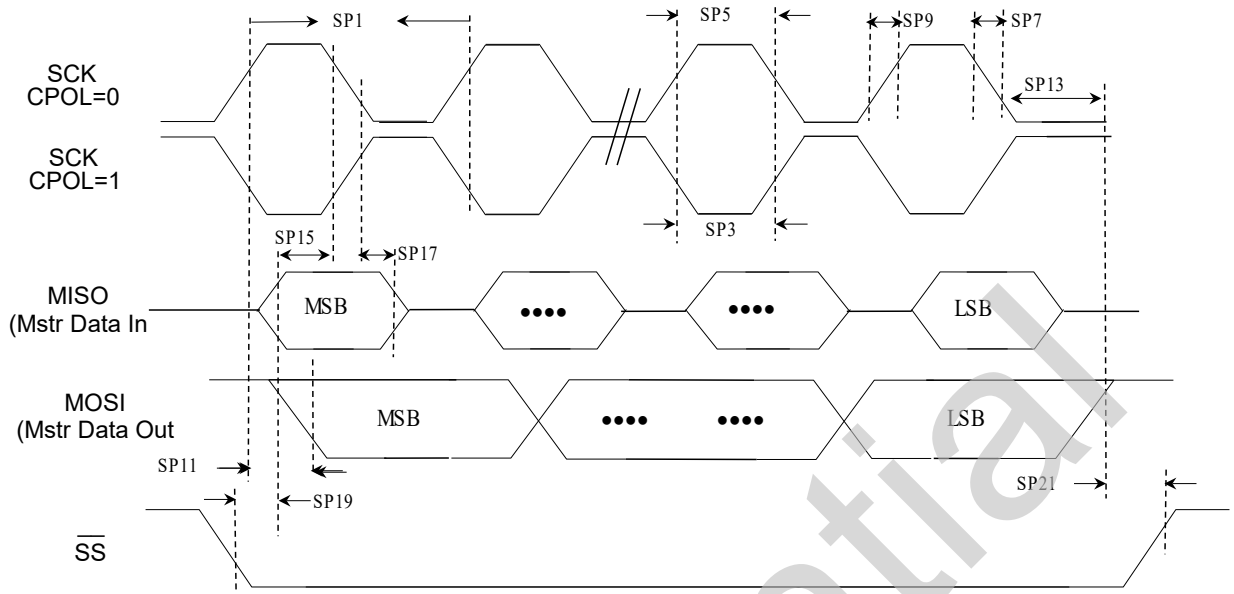


Figure 38-23. SPI Client Module CPHA=1 Timing Diagrams



38.19 ADC Electrical Specifications

Table 38-22. ADC AC Electrical Specifications

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
Device Supply							
ADC_1	V_{DDANA}	ADC supply	$V_{DDANA(min)}$	—	$V_{DDANA(max)}$	V	V_{DDIO} = V_{DDANA}
Reference Inputs							
ADC_3	V_{REF}(6)	ADC reference voltage(4)	—	—	V_{DDANA}	V	External reference voltage
Analog Input Range							
ADC_7	AFS	Full-scale analog input signal range (Single-ended)	0	—	V_{DDANA}	V	V_{REF} = $V_{DDANA(max)}$
ADC_9		Full-scale analog input signal range (Differential)	V_{DDANA}/2	—	V_{DDANA}/2	V	

Table 38-23. ADC Single-Ended Mode AC Electrical Specifications

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
Single-Ended Mode ADC Accuracy							
SADC_11	Res	Resolution	6	—	12	bits	Selectable 6, 8, 10, 12 bit resolution ranges
SADC_13	EN0B(3)	Effective number of bits	—	9.3	—	bits	1.6 Msps, Internal V_{REF}, V_{DDANA} = V_{DDIO} = 3.3V
SADC_19	INL(3)	Integral non linearity	—	-5.2 2.7	—	LSb	1.6 Msps, Internal V_{REF}, V_{DDANA} = V_{DDIO} = 3.3V
SADC_25	DNL(3)	Differential non linearity	—	-1 1.8	—	LSb	1.6 Msps, Internal V_{REF}, V_{DDANA} = V_{DDIO} = 3.3V
SADC_31	GERR(3)	Gain error	—	-0.7	—	LSb	1.6 Msps, Internal V_{REF}, V_{DDANA} = V_{DDIO} = 3.3V
SADC_37	E0FF(3)	Offset error	—	3.2	—	LSb	Internal V_{REF}, V_{DDANA} = V_{DDIO} = 3.3V
Single-Ended Mode ADC Dynamic Performance							
SADC_49	SINAD(1,2,3)	Signal to noise and distortion	—	57.8	—	dB	V_{REF} = V_{DDANA} = V_{DDIO} = 3.3V at 12-bit resolution, max sampling rate(1,2)
SADC_51	SNR(1,2,3))	Signal to noise ratio	—	58.2	—		
SADC_53	SFDR(1,2,3)	Spurious free dynamic range	—	66.2	—		
SADC_55	THD(1,2,3,5)	Total harmonic distortion	—	-71.3	—		

Notes:

1. Characterized with an analog input sine wave = (FTP(maximum)/100). Example: FTP(maxium) = 1 Msps/100 = 10 KHz sine wave.
2. Sinewave peak amplitude = 96% ADC_ Full Scale amplitude input with 12-bit resolution.
3. Spec values collected under the following additional conditions:
   a. 12-bit resolution mode.
   b. All registers at reset default value otherwise not mentioned.
4. ADC Measurements done with 3.3V V_{REF} Voltage.
5. Value taken over 7 harmonics.
6. Referred to as AVDD in the pinout.

Table 38-24. ADC Conversion AC Electrical Requirements

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
ADC_ Clock Requirements							
ADC_57	TAD	ADC clock period	—	20.8	—	ns	V_{REF} = V_{DDANA} = 3.3V
ADC Single-Ended Throughput Rates							
ADC_59	FTPR (Single-ended Mode)	Throughput rate(2) (Single-ended)	—	2	—	Msps	12-bit resolution, DIV_SHR = 2
			—	0.7	—		12-bit resolution, DIV_SHR = 4

Notes:

1. Conversion_time = (SAMC_SHR+15)*TAD.
2. FTPR = 1/((SAMC_SHR + Resolution + 1) * (1/(ControlClk/DIV_SHR))).

Table 38-25. ADC Sample AC Electrical Requirements

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
ADC_65	TCNV	Conversion time(1) (Single-ended Mode)		14		TAD	12-bit resolution
				12			10-bit resolution
				10			8-bit resolution
ADC_67		Conversion time(1) (Differential Mode)		14		TAD	12-bit resolution
				12			10-bit resolution
				10			8-bit resolution
ADC_69	CSAMPLE	ADC internal sample cap	—	5	—	pf	—
ADC_71	RSAMPLE	ADC internal impedance	—	—	200	Ω	—

Note:

1. ADC Throughput Rate FTP = ((1/((TSAMP + TCNV) * TAD))/(# of user active analog inputs in use on specific target ADC module)).

38.20 Bluetooth Low Energy RF Characteristics

Table 38-26. Bluetooth Low Energy RF Characteristics

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ(1)**	**Max.**	**Units**	**Conditions**
BTG1	FREQ	Frequency range of operation	2400	—	2480	MHz	—
BTTX1	TXPWR:MPA	Bluetooth transmit power MPA	—	11.5	—	dBm	—
BTTX2	TXPWR:LPA	Bluetooth transmit power LPA	—	4.0	—	dBm	—
BTX3	TXIB:1MBPS	In-band emission for FTX ± -2 MHz	—	-32	—	dBm	—
		In-band emission for FTX ± -(3+N) MHz	—	-45	—	dBm	—
BTX4	TXIB:2MBPS	In-band emission for FTX ± -4 MHz	—	-43	—	dBm	—
		In-band emission for FTX ± -5 MHz	—	-48	—	dBm	—
		In-band emission for FTX ± -(6+N) MHz	—	-51	—	dBm	—
BTRX1	RXSENSE	Receiver sensitivity at 1 Mbps	—	-95.5	—	dBm	(1)
		Receiver sensitivity at 2 Mbps	—	-92.5	—	dBm	(5)
		Receiver sensitivity at 500 kbps	—	-98.5	—	dBm	(5)
		Receiver sensitivity at 125 kbps	—	-102	—	dBm	(5)
BTRX2	MAXINSIG	Maximum input signal level at 1 Mbps	—	0	—	dBm	—
		Maximum input signal level at 2 Mbps	—	0	—	dBm	
		Maximum input signal level at 500 kbps	—	0	—	dBm	
		Maximum input signal level at 125 kbps	—	0	—	dBm	

.....continued

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	Symbol	Characteristics	Min.	Typ(1)	Max.	Units	Conditions
BTRX3(4)	CI1M:COCH	C/I Co channel rejection	—	13	—	dB	—
	CI1M: ± -1 MHz	C/I adjacent channel rejection	—	13	—	dB	—
	CI1M: ± -2 MHz	C/I adjacent channel rejection	—	13	—	dB	—
	CI1M:ADJ(3+n)	C/I alternate channel rejection	—	15	—	dB	—
	CI1M:IMG	C/I image frequency rejection	—	15	—	dB	—
	CI1M:IMG ± -1 MHz	C/I adjacent channel to image freq rejection	—	14	—	dB	—
BTRX4(4)	CIS2:COCH	C/I Co channel rejection	—	11	—	dB	—
	CIS2: ± -1 MHz	C/I adjacent channel rejection	—	17	—	dB	—
	CIS2: ± -2 MHz	C/I adjacent channel rejection	—	18	—	dB	—
	CIS2:ADJ(3+n)	C/I alternate channel rejection	—	17	—	dB	—
	CIS2:IMG	C/I image frequency rejection	—	14	—	dB	—
	CIS2:IMG ± -1 MHz	C/I adjacent channel to image freq rejection	—	18	—	dB	—
BTRX5(4)	CIS8:COCH	C/I Co channel rejection	—	6	—	dB	—
	CIS8: ± -1 MHz	C/I adjacent channel rejection	—	13	—	dB	—
	CIS8: ± -2 MHz	C/I adjacent channel rejection	—	13	—	dB	—
	CIS8:ADJ(3+n)	C/I alternate channel rejection	—	14	—	dB	—
	CIS8:IMG	C/I image frequency rejection	—	8	—	dB	—
	CIS2:IMG ± -1 MHz	C/I adjacent channel to image freq rejection	—	16	—	dB	—

.....continued

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ(1)**	**Max.**	**Units**	**Conditions**
BTRX6(4)	CI2M:COCH	C/I Co channel rejection	—	13	—	dB	—
	CI2M: ± -2 MHz	C/I adjacent channel rejection	—	16	—	dB	—
	CI2M: ± -4 MHz	C/I adjacent channel rejection	—	19	—	dB	—
	CI2M:ADJ(6+2n)	C/I alternate channel rejection	—	16	—	dB	—
	CI2M:IMG	C/I image frequency rejection	—	13	—	dB	—
	CI2M:IMG ± -2 MHz	C/I adjacent channel to image freq rejection	—	19	—	dB	—
BTRX7(4)	BLOCK1M:<2 GHZ	Blocking performance from 30-2 GHz	—	20	—	dB	—
	BLOCK1M:2 GHZ<SIG<2399 MHz	Blocking performance from 2003-2399 MHz	—	14	—	dB	—
	BLOCK1M:2484 MHZ<SIG<2977 MHz	Blocking performance between 2484-2997 MHz	—	20	—	dB	—
	BLOCK1M:3 GHZ<SIG<12.75 GHz	Blocking performance between 3-12.5 GHz	—	20	—	dB	—
BTRX8(4)	BLE1M:INTERMOD	Inter modulation performance for BLEM	—	13.5	—	dB	—
	BLE2M:INTERMOD	Inter modulation performance for BLEM	—	19.5	—	dB	—

Notes:

1. Measured at 25°C, averaged across all voltages and channels.
2. Measured on a board with the reference schematic.
3. All measurement across voltage based on the SIG specifications.
4. The specified value is the limit above the SIG specifications.
5. PDU length = 37, channels = 2402/2426/2440/2480 MHz.

Table 38-27. Bluetooth Low Energy RF Current Characteristics

AC Characteristics					Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**RF Power**	**CPU Frequency**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
IBLETX1	IDDTXMPA	Current consumption with output power in DC-DC mode 1 Mbps	+12 dBm	64 MHz	—	42.8	—	mA	—
IBLETX2			+12 dBm	32 MHz	—	40.5	—	mA	—
IBLETX3			+12 dBm	8 MHz	—	39.0	—	mA	—
IBLETX4		Current consumption at +12 dBm output power in MLDO mode	+12 dBm	64 MHz	—	96.7	—	mA	—
IBLETX5			+12 dBm	32 MHz	—	91.8	—	mA	—
IBLETX6			+12 dBm	8 MHz	—	88.1	—	mA	—
IBLETX7	IDDTXLPA	Current consumption at +4 dBm output power in DC-DC mode 1 Mbps	4 dBm	64 MHz	—	24.9	—	mA	—
IBLETX8			4 dBm	32 MHz	—	22.9	—	mA	—
IBLETX9			4 dBm	8 MHz	—	21.1	—	mA	—
IBLETX10		Current consumption at +4 dBm output power in MLDO mode	4 dBm	64 MHz	—	55.5	—	mA	—
IBLETX11			4 dBm	32 MHz	—	48.7	—	mA	—
IBLETX12			4 dBm	8 MHz	—	45.8	—	mA	—
IBLETX7	IDDTXLPA0	Current consumption at +0 dBm output power in DC-DC mode 1 Mbps	0 dBm	64 MHz	—	22.7	—	mA	—
IBLETX8			0 dBm	32 MHz	—	20.9	—	mA	—
IBLETX9			0 dBm	8 MHz	—	18.2	—	mA	—
IBLETX10		Current consumption at 0 dBm output power in MLDO mode	0 dBm	64 MHz	—	47.6	—	mA	—
IBLETX11			0 dBm	32 MHz	—	43.0	—	mA	—
IBLETX12			0 dBm	8 MHz	—	39.7	—	mA	—

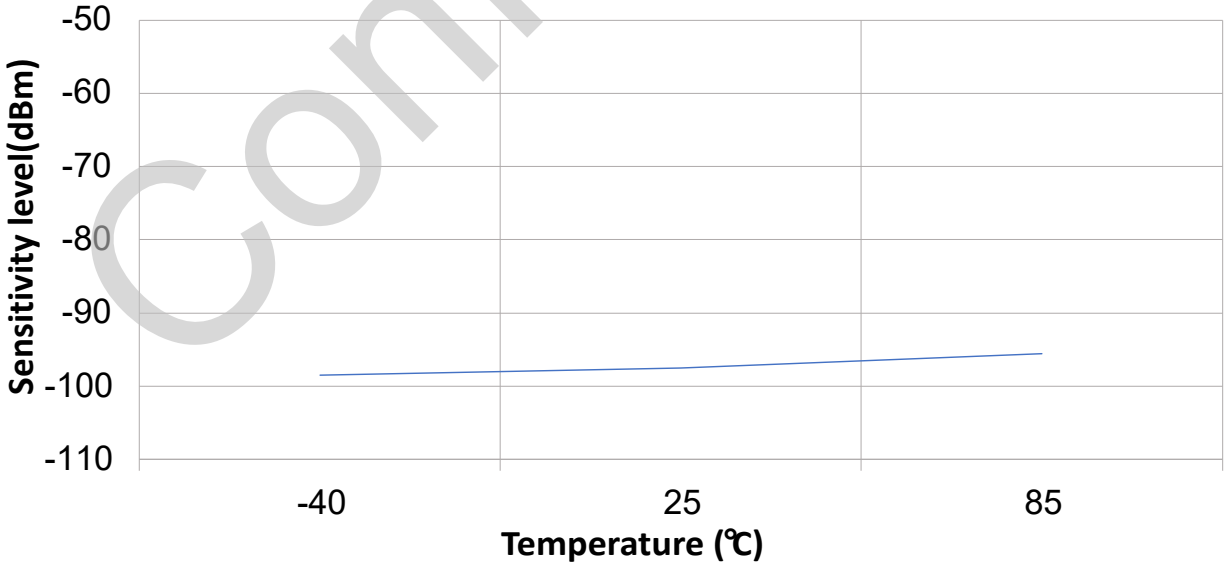
.....continued

AC Characteristics					Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9-3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	Symbol	Characteristics	RF Power	CPU Frequency	Min.	Typ.	Max.	Units	Conditions
IBLERX1	IDDRXBLE1M	Current consumption at RX signal level -80 dBm in DC-DC mode	-80 dBm	64 MHz	—	20.6	—	mA	—
IBLERX2			-80 dBm	32 MHz	—	18.2	—	mA	—
IBLERX3			-80 dBm	8 MHz	—	16.5	—	mA	—
IBLERX4		Current consumption at RX signal level -80 dBm in MLDO mode	-80 dBm	64 MHz	—	40.6	—	mA	—
IBLERX5			-80 dBm	32 MHz	—	35.1	—	mA	—
IBLERX6			-80 dBm	8 MHz	—	30.9	—	mA	—

Notes:

- Current consumption is measured on a board based upon the Microchip Technology Reference Design.
- Current consumption is for the entire SoC (including the MCU), measured at the input power rail.
- Current consumption is measured using HUT code.
- Current reported is the average of the current during the transmit or receive burst (exclude off cycle of the transmit/receive operation).

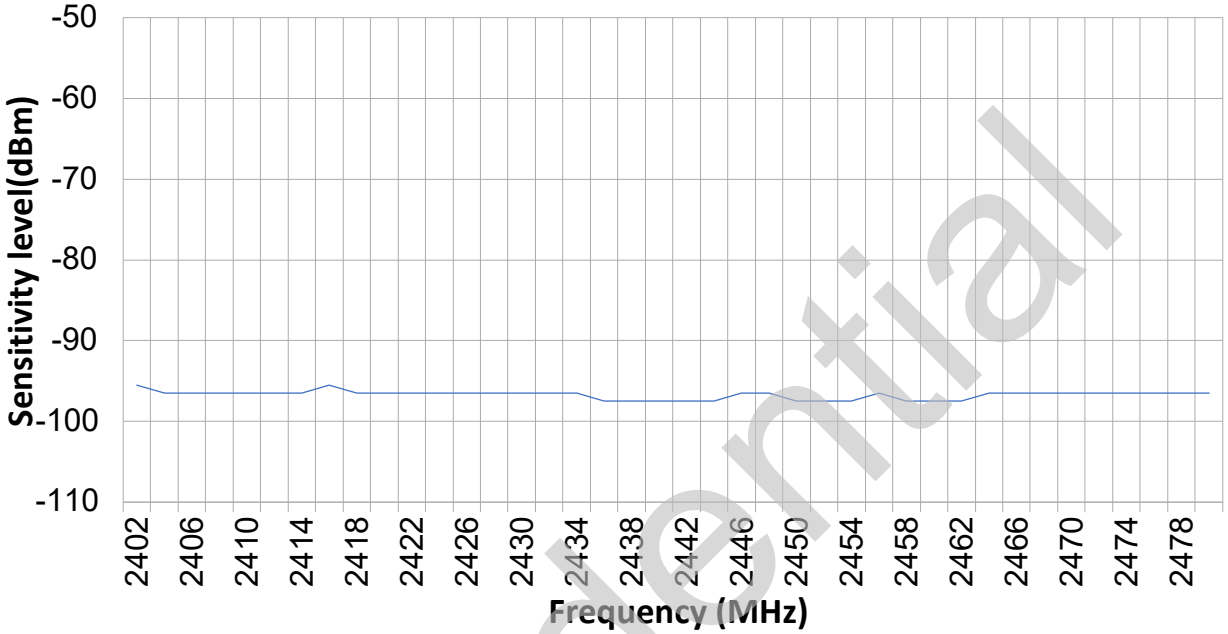
Figure 38-24. Module Bluetooth Low Energy Receive Sensitivity vs Temperature



Notes:

- Bluetooth Low Energy receive sensitivity is measured across temperature at 3.6V, 2440 MHz, uncoded data at 1 Ms/s.
- PDU length = 37.
- Sensitivity is measured according to the SIG specifications.

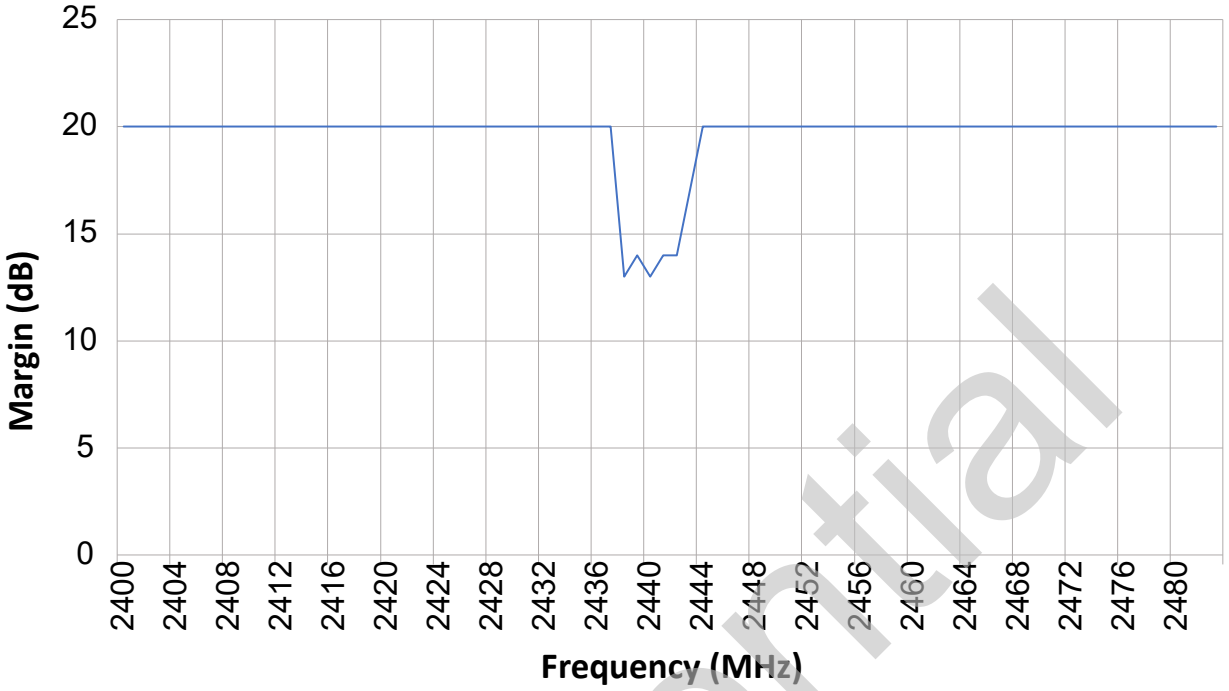
Figure 38-25. Module Bluetooth Low Energy Receive Sensitivity vs Frequency



Notes:

- Bluetooth Low Energy sensitivity is measured across channels at 3.6V at 25℃, uncoded data at 1 Ms/s.
- PDU length = 37.
- Sensitivity is measured according to the SIG specifications.

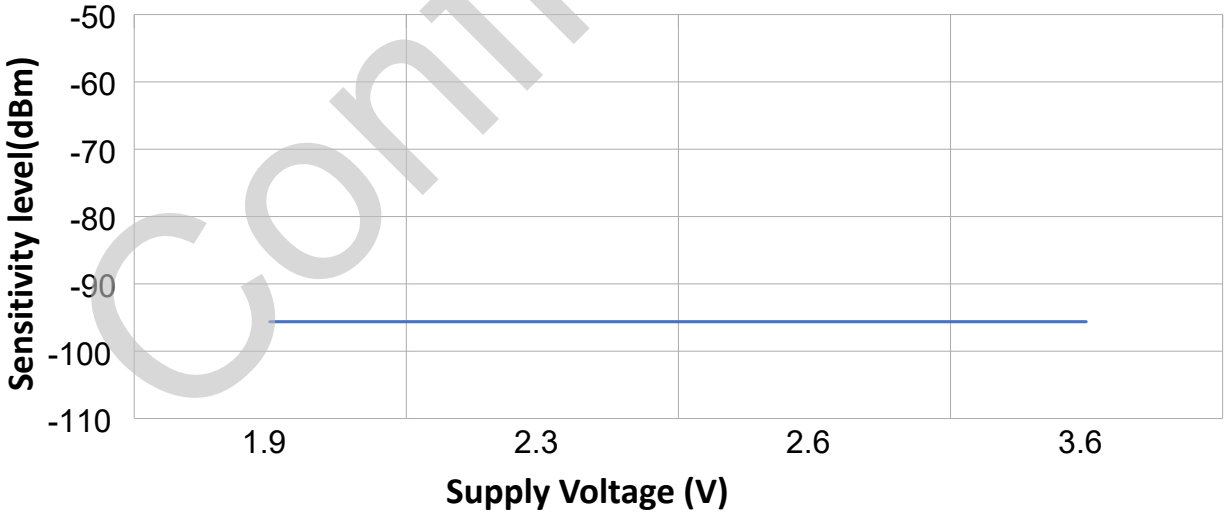
Figure 38-26. Bluetooth Low Energy 1M CI Margin



Notes:

- Bluetooth Low Energy 1M C/I Margin is measured at 2440 MHz at 25°C, 3.6V, uncoded data at 1 Ms/s.
- C/I test is done with HUT code based on the SIG specifications.
- Reported C/I margin is the margin above the C/I specifications from SIG.

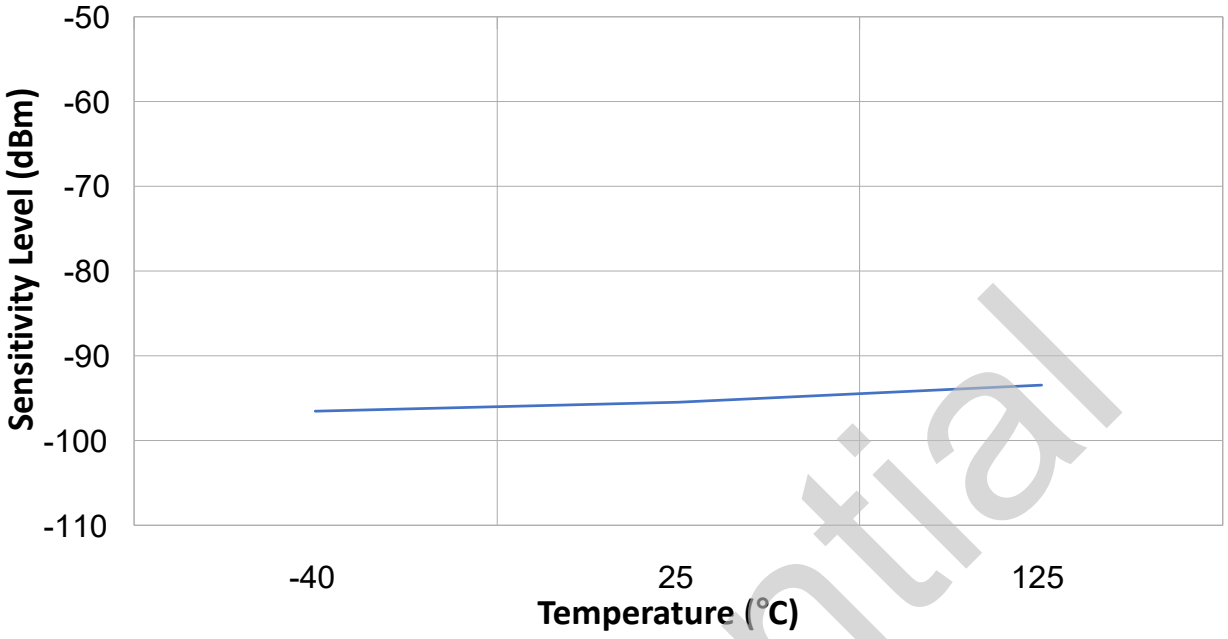
Figure 38-27. Bluetooth Low Energy Receive Sensitivity vs Voltage



Notes:

- Bluetooth Low Energy receive sensitivity is measured at 2440 MHz at 25°C, uncoded data at 1 Ms/s.
- PDU length = 37.
- Sensitivity is measured according to the SIG specifications.

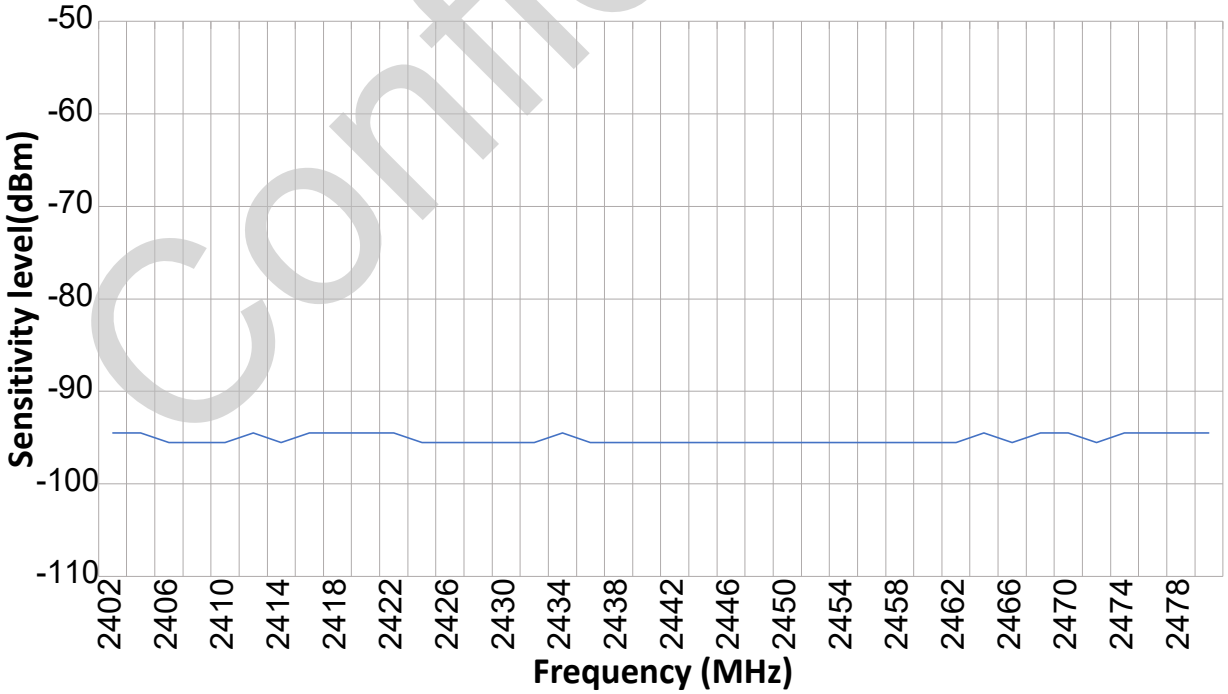
Figure 38-28. Bluetooth Low Energy Receive Sensitivity vs Temperature



Notes:

- Bluetooth Low Energy receive sensitivity is measured across channels at 3.6V, 2440 MHz, uncoded data at 1 Ms/s.
- PDU length = 37.
- Sensitivity is measured according to the SIG specifications.

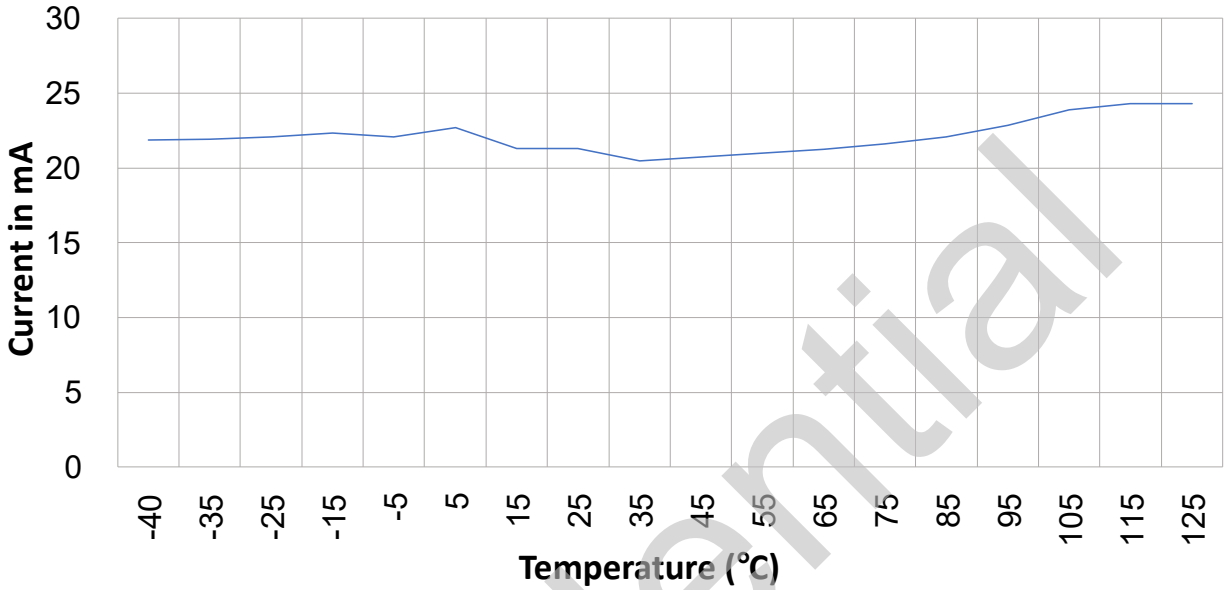
Figure 38-29. Bluetooth Low Energy Receive Sensitivity vs Frequency



Notes:

- Bluetooth Low Energy receiver sensitivity is measured across channels at 3.6V at 25℃, uncoded data at 1 Ms/s.
- PDU length = 37.
- Sensitivity is measured according to the SIG specifications.

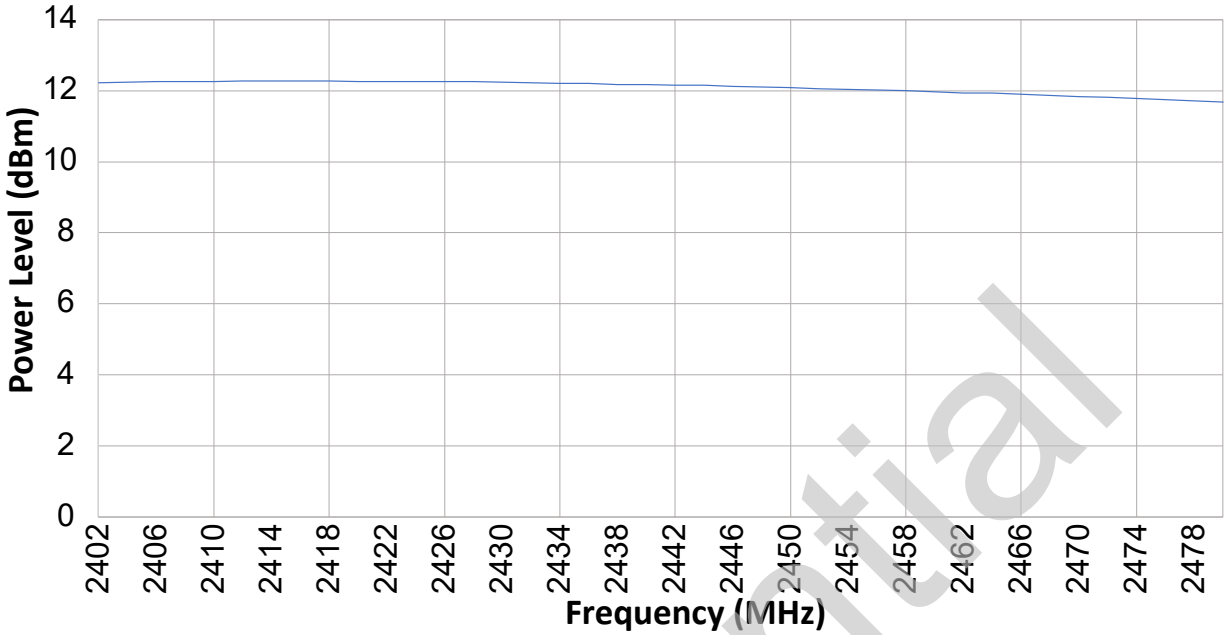
Figure 38-30. Bluetooth Low Energy Receive Current vs Temperature



Notes:

- Bluetooth Low Energy receive current is measured at 3.3V (Buck mode), uncoded data at 1 Ms/s with LNA configured at maximum gain.
- PDU length = 37.
- Current is measured on input power rail to SoC (includes processor current as well).
- Current is measured with HUT code.

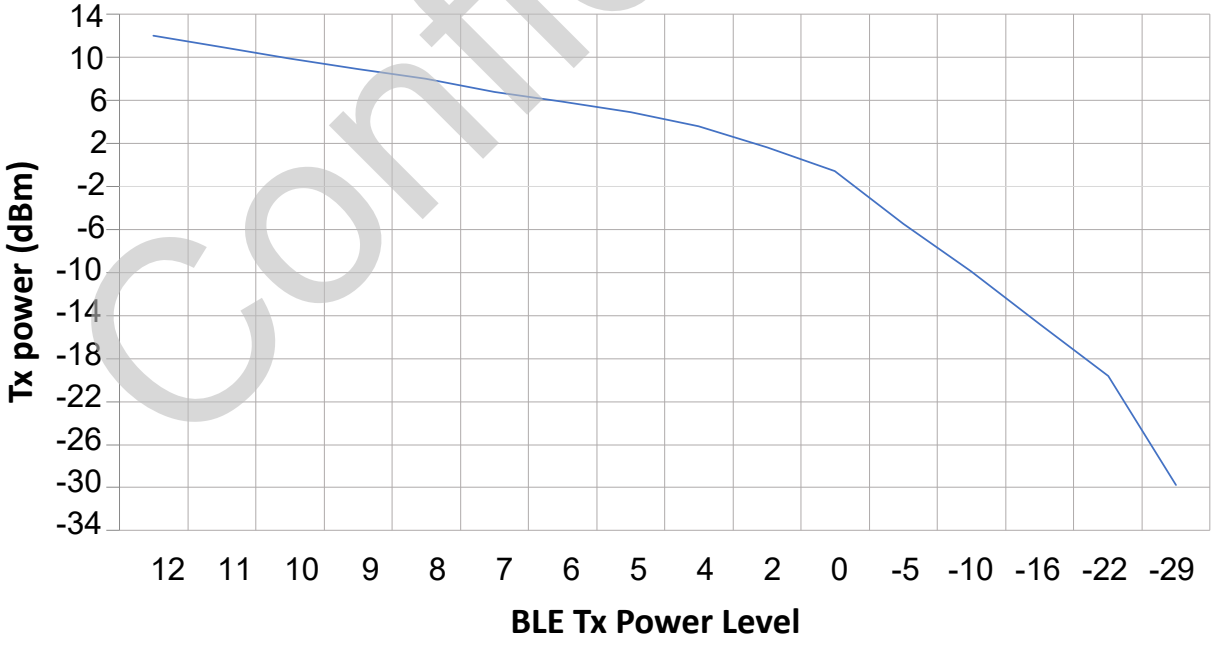
Figure 38-31. Bluetooth Low Energy Transmit Power vs Frequency



Notes:

- Bluetooth Low Energy transmit power is measured across frequency after transmit power calibration at 3.3V (Buck mode).
- Transmit power is measured with HUT code.
- Transmit power is measured after the PA matching and LPF.

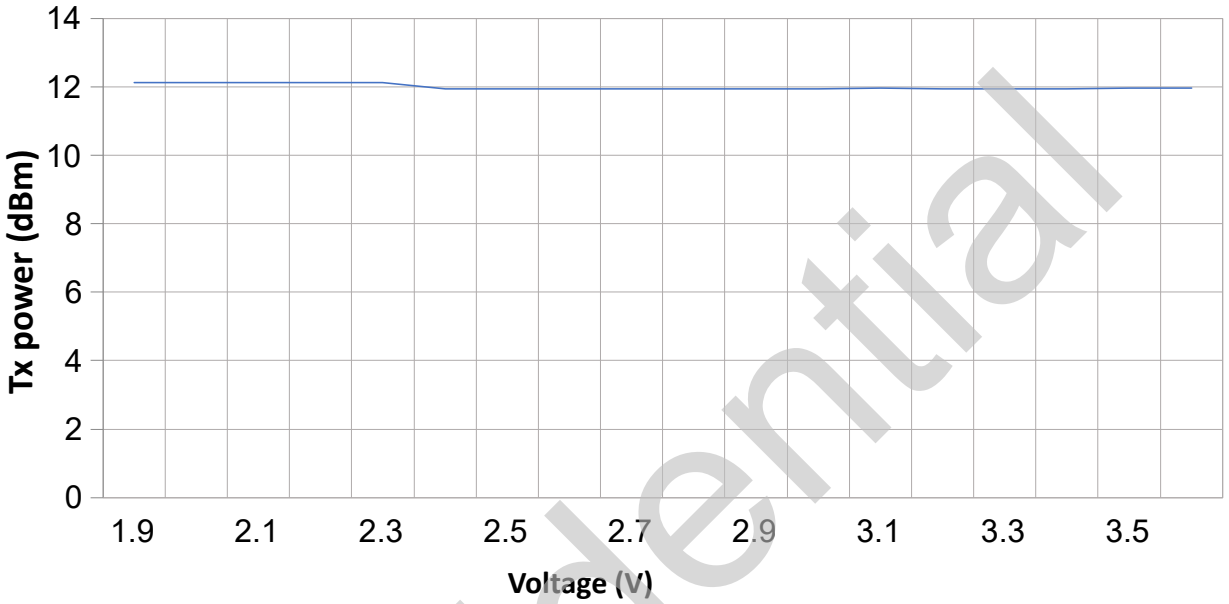
Figure 38-32. Bluetooth Low Energy Transmit Power vs Transmit Power Level



Notes:

- Bluetooth Low Energy transmit power is measured at 2440 MHz after transmit power calibration.
- Transmit power is measured on board based on Microchip Technology Reference Design.
- Transmit power is measured after PA match and LPF.
- Transmit power is measured with HUT code.
- Transmit power is controlled by transmit power settings on HUT code for measurement.

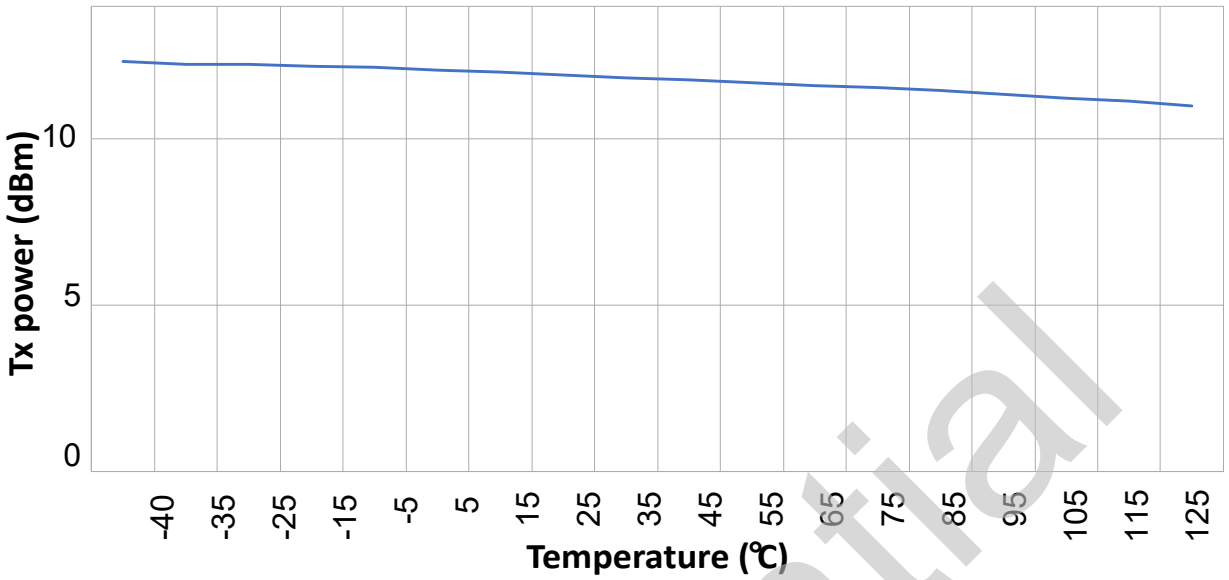
Figure 38-33. Bluetooth Low Energy Transmit Power vs VDD Supply Voltage



Notes:

- Bluetooth Low Energy transmit power is measured across voltage after transmit power calibration.
- Transmit power is measured after calibration at +12 dBm (± 0.5 dBm).
- Transmit power is measured on board based on the Microchip Reference Design.
- Transmit power is measured after the LPA and PA match section.
- Transmit power is measured with HUT code.

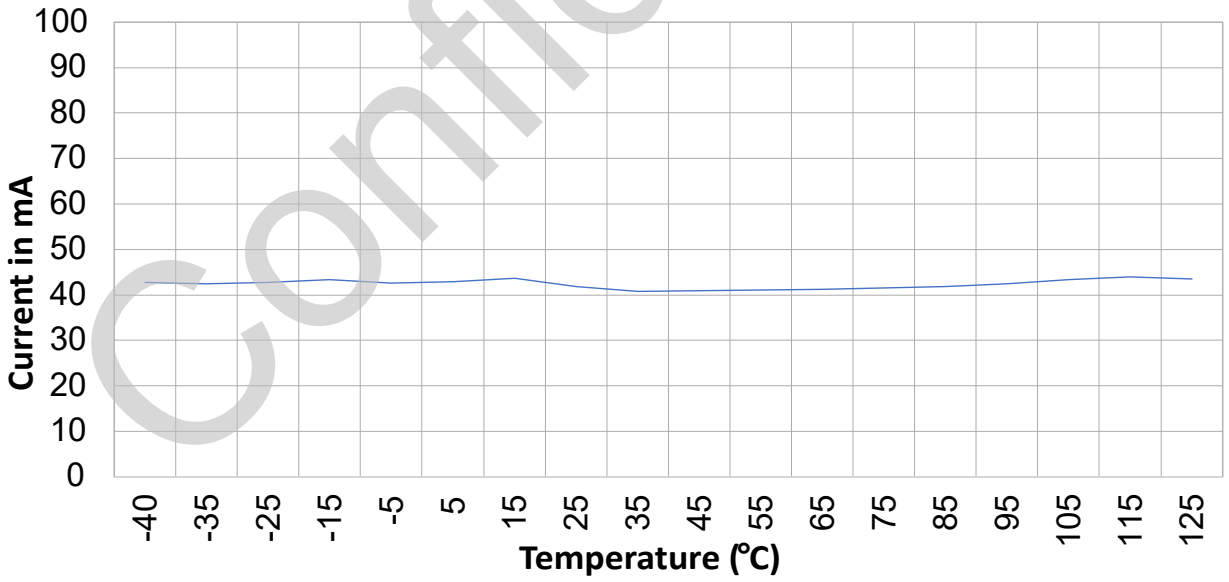
Figure 38-34. Bluetooth Low Energy Transmit Power vs. Temperature



Notes:

- Bluetooth Low Energy transmit power is measured across temperature after transmit power calibration at 3.6V and 2440 MHz.
- Transmit power is measured with HUT code.
- Temperature power compensation is triggered before power measurement.
- Transmit power is measured after the PA matching and LPF.

Figure 38-35. Bluetooth Low Energy Transmit Current vs Temperature



Notes:

- Bluetooth Low Energy transmit current is measured at 3.3V (Buck mode) at 2440 MHz across temperature.
- Transmit current is measured after calibration at +12 dBm (± 0.5 dBm).
- Current is measured on input power rail to SoC.
- Current is measured with HUT code.

38.21 Zigbee RF Characteristics

Table 38-28. Zigbee RF Characteristics

AC Characteristics			Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.(1,2)**	**Max.**	**Units**	**Conditions**
ZBG1	FREQ	Frequency range	2405	—	2480	MHz	—
ZBG2	FCH	Channel spacing	—	5	—	MHz	—
ZBG3	PSDU	Bit rate	—	250, 500, 1000, 2000	—	kbps	—
ZBT1	TXOPMPA	Transmit output power MPA	—	11.5	—	dBm	(4)
ZBT2	TXOPLPA	Transmit output power LPA	—	4.0	—	dBm	(4)
ZBT3	POWERRANGE	Output power range	—	26	—	dB	TX power on ZB power levels from (-14 to 12 dBm)
ZBT4	EVM	Error vector magnitude	—	10	—	%RMS	—
ZBRX1	SENS250	Receiver sensitivity in 250 kbps	—	-99	—	dBm	—
	SENS500	Receiver sensitivity in 500 kbps	—	-96	—	dBm	—
	SENS1M	Receiver sensitivity in 1 Mbps	—	-94	—	dBm	—
	SENS2M	Receiver sensitivity in 2 Mbps	—	-88	—	dBm	—
ZBRX2	PMAX	Maximum input level	—	0	—	dBm	—
ZBRX3(3)	PACRP	Adjacent channel rejection +5 MHz	—	35	—	dB	(6)
ZBRX4(3)	PACRN	Adjacent channel rejection -5 MHz	—	31	—	dB	
ZBRX5(3)	PALRP	Alternate channel rejection +10 MHz	—	17	—	dB	
ZBRX6(3)	PALRN	Alternate channel rejection -10 MHz	—	17	—	dB	
ZBRX7	LOLEAKLPA	LO leakage on LPA mode	—	-34	—	dB	(5)
ZBRX7A	LOLEAKMPA	LO leakage on MPA mode	—	-28	—	dB	(5)

.....continued							
AC Characteristics			**Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp**				
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.(1,2)**	**Max.**	**Units**	**Conditions**
ZBRX8	RSSIRANGE	Dynamic range of RSSI	—	40	—	dB	—

Notes:

1. Measured on a board based on the Microchip Technology reference design.
2. Measured across channels at 3.3V and according to the 802.15.4 standard specifications.
3. Specified value is the margin above the 802.15.4 standard limits.
4. Measured across channels and voltages.
5. LO leakage on LPA mode, measured across voltage.
6. All results are based on measurement conditions as per the 802.15.4 standards.

Table 38-29. Zigbee RF Current Characteristics

AC Characteristics				**Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp**				
Param. No.	**Symbol**	**Characteristics**	**CPU Frequency**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
IZBTX1	IDDTXMPA	Current consumption at +12 dBm output power in DC-DC mode 250 kbps	64 MHz	—	43.3	—	mA	—
IZBTX2			32 MHz	—	41.3	—	mA	—
IBLETX3			8 MHz	—	39.9	—	mA	—
IZBTX4		Current consumption at +12 dBm output power in MLDO mode	64 MHz	—	96.4	—	mA	—
IZBTX5			32 MHz	—	92.1	—	mA	—
IZBTX6			8 MHz	—	89.0	—	mA	—
IZBTX7	IDDTXLPA	Current consumption at +4 dBm output power in DC-DC mode 250 kbps	64 MHz	—	27.3	—	mA	—
IZBTX8			32 MHz	—	24.9	—	mA	—
IZBTX9			8 MHz	—	22.7	—	mA	—
IZBTX10		Current consumption at +4 dBm output power in MLDO mode	64 MHz	—	51.7	—	mA	—
IZBTX11			32 MHz	—	47.6	—	mA	—
IZBTX12			8 MHz	—	45.0	—	mA	—
IZBRX1	IDDRXZB	Current consumption at RX signal level -95 dBm in DC-DC mode	64 MHz	—	19.4	—	mA	—
IZBRX2			32 MHz	—	17.4	—	mA	—
IZBRX3			8 MHz	—	15	—	mA	—
IZBRX4		Current consumption at RX signal level -95 dBm in MLDO mode	64 MHz	—	38.5	—	mA	—
IZBRX5			32 MHz	—	33.5	—	mA	—
IZBRX6			8 MHz	—	30.3	—	mA	—

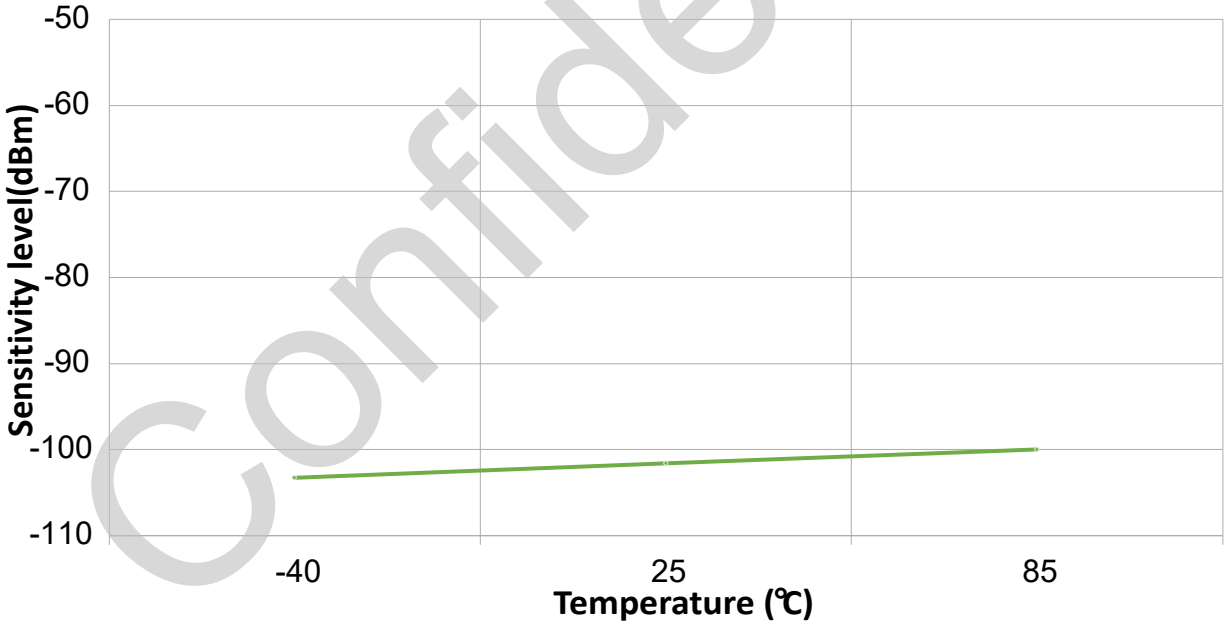
.....continued

AC Characteristics				Standard Operating Conditions: V_{DDIO} = V_{DDANA} 1.9V to 3.6V (unless otherwise stated) Operating Temperature: -40°C ≤ T_A ≤ +85°C for Industrial Temp				
Param. No.	Symbol	Characteristics	CPU Frequency	Min.	Typ.	Max.	Units	Conditions
IZBRX1	IDDRXZBRPC	Current consumption at RX signal level -95 dBm in DC-DC mode	64 MHz	—	13.6	—	mA	—
IZBRX2			32 MHz	—	11.3	—	mA	—
IZBRX3			8 MHz	—	9.8	—	mA	—
IZBRX4		Current consumption at RX signal level -95 dBm in MLDO mode	64 MHz	—	29	—	mA	—
IZBRX5			32 MHz	—	24.2	—	mA	—
IZBRX6			8 MHz	—	20.7	—	mA	—

Notes:

- Current consumption is measured on a board based upon the Microchip Technology Reference Design.
- Current consumption is for the entire SoC (including the MCU).
- Current consumption is measured using the HUT code.
- Current reported is the average of the current during the transmit burst (exclude off cycle of the transmission).

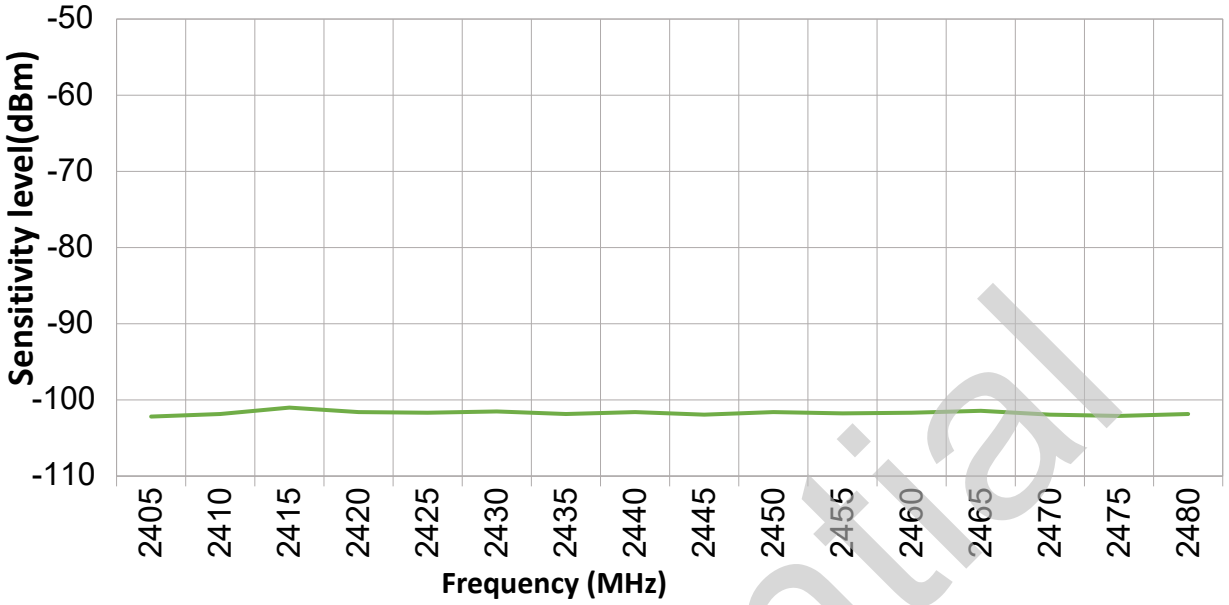
Figure 38-36. Module Zigbee Receive Sensitivity vs. Temperature



Notes:

- Receiver sensitivity is measured based on the 802.15.4 specifications.
- Receiver sensitivity is measured at 2440 MHz at 3.6V, 250 kbps.
- Measured after receiver calibration.

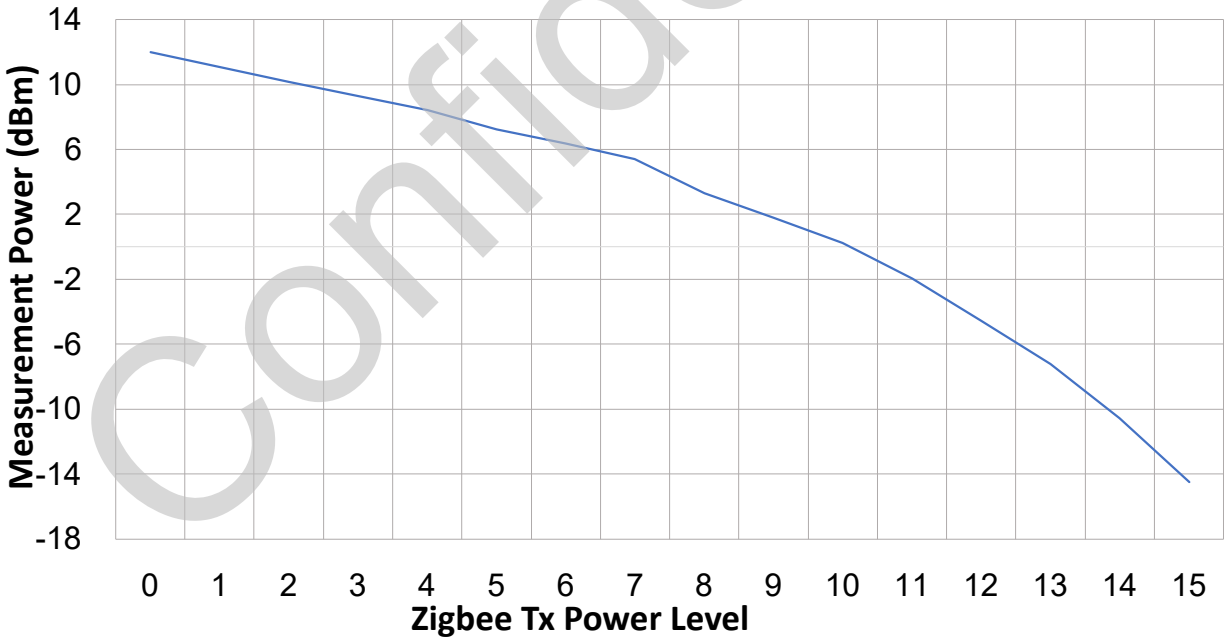
Figure 38-37. Module Zigbee Receive Sensitivity vs. Frequency



Notes:

- RX sensitivity across channels is measured at 3.6V at 25°C, 250 kbps.
- Sensitivity is measured according to the 802.15.4 specifications.

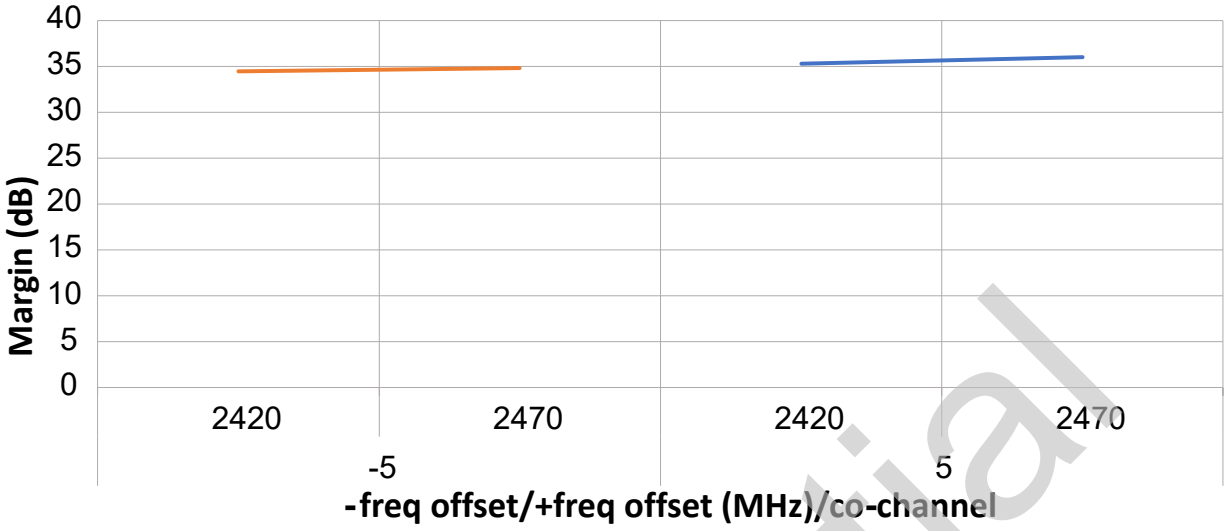
Figure 38-38. Zigbee TX Setting vs. Measurement Power



Notes:

- Transmit power is measured after calibration.
- Transmit power is measured across power levels at 2440 MHz at 3.6V, 25°C.
- Transmit power is measured after the PA matching and LPF.
- Transmit power is configured using the transmit power setting in HUT code.

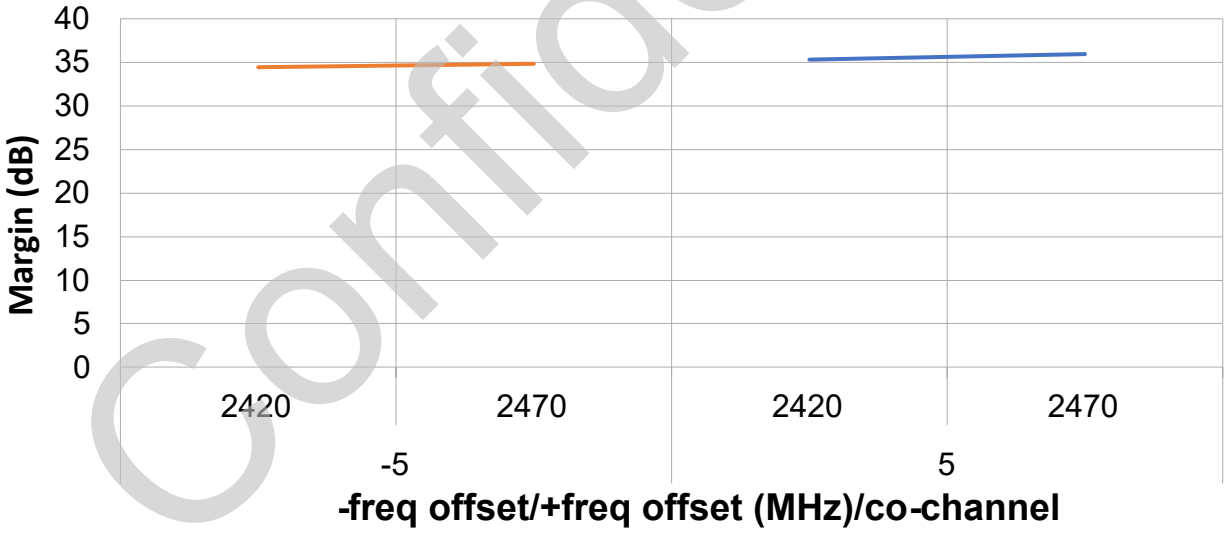
Figure 38-39. Zigbee ACR +-5M Margin



Notes:

- Adjacent channel rejection is measured at 3.6V at 25℃, 250 kbps.
- Measured based on the 802.15.4 adjacent channel relative jamming specification.
- Margin specified is the margin above the 802.15.4 specifications.
- Measured after receiver calibration.

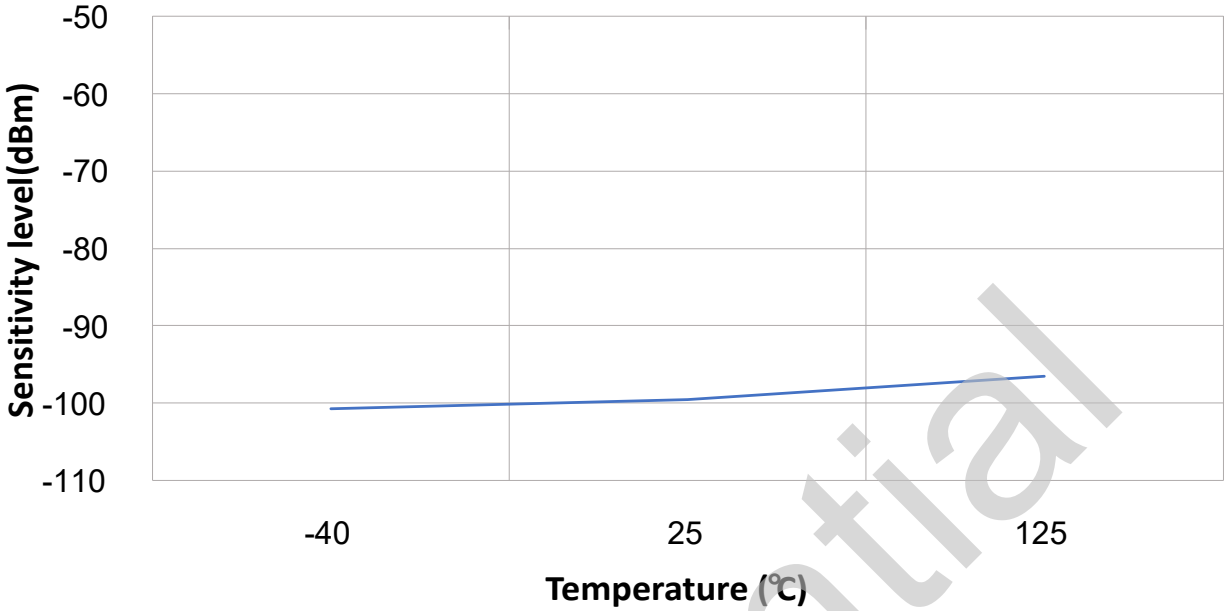
Figure 38-40. Zigbee ACR +-10M Margin



Notes:

- Adjacent channel rejection is measured at 3.6V at 25℃, 250 kbps.
- Measured based on the 802.15.4 adjacent channel relative jamming specification.
- Margin specified is the margin above the 802.15.4 specifications.
- Measured after receiver calibration.

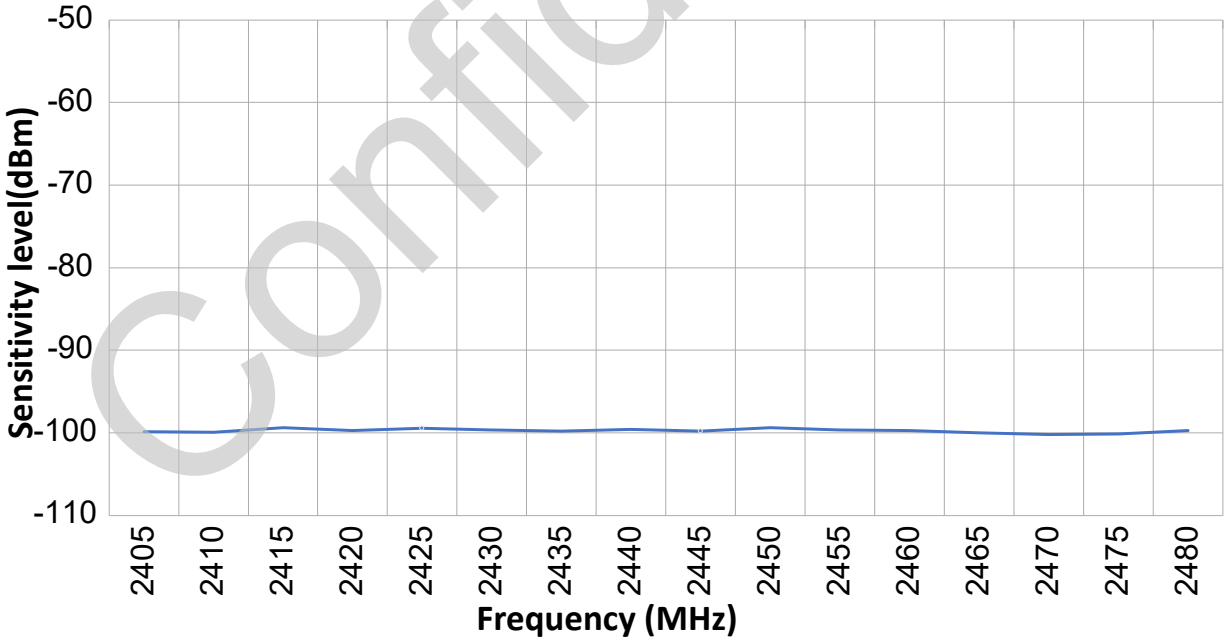
Figure 38-41. Zigbee Receive Sensitivity vs. Temperature



Notes:

- Receiver sensitivity is measured based on the 802.15.4 specifications.
- Sensitivity measured at 3.6V, 25℃ on 2440 MHz across temperature, 250 kbps.
- Measured after receiver calibration.

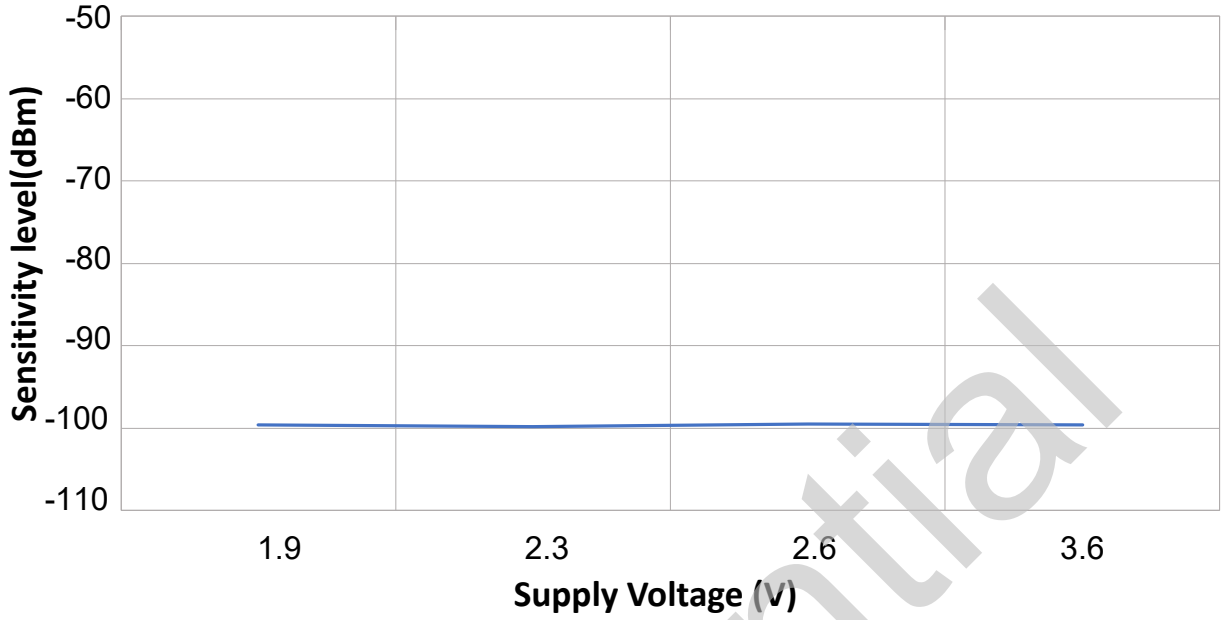
Figure 38-42. Zigbee Receive Sensitivity vs. Frequency



Notes:

- RX sensitivity is measured across channels at 3.6V at 25℃, 250 kbps.
- Sensitivity is measured according to the 802.15.4 specifications.
- Sensitivity is measured after RX calibration.

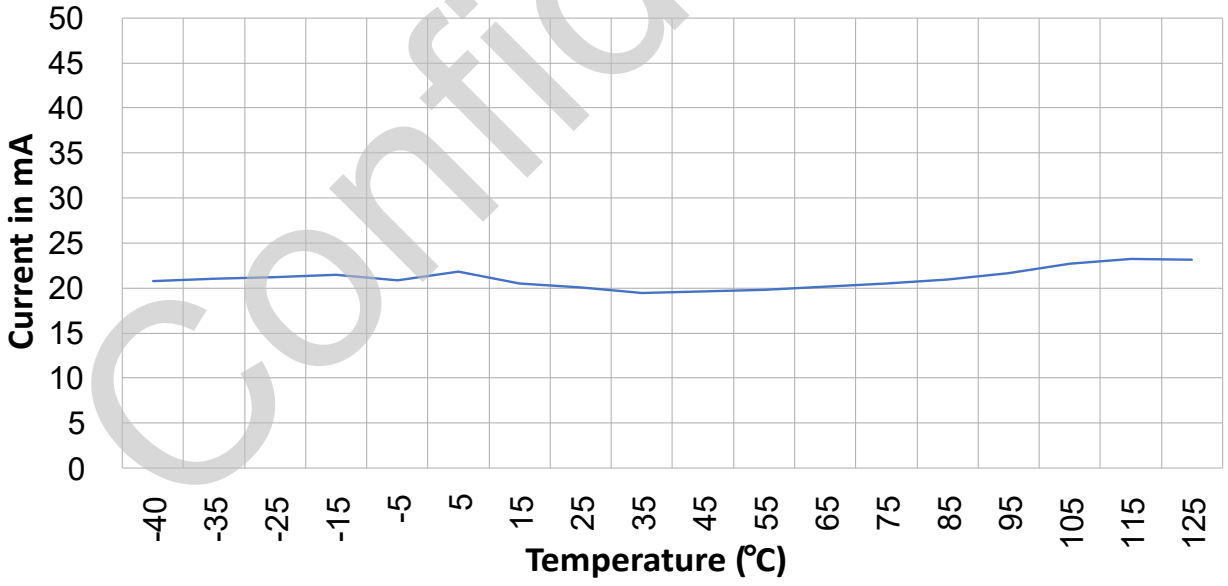
Figure 38-43. Zigbee Receive Sensitivity vs. VDD Supply Voltage



Notes:

- RX sensitivity is measured at 2440 MHz at 25℃ across voltage, 250 kbps.
- Sensitivity is measured according to the 802.15.4 specifications.
- Sensitivity is measured after RX calibration.

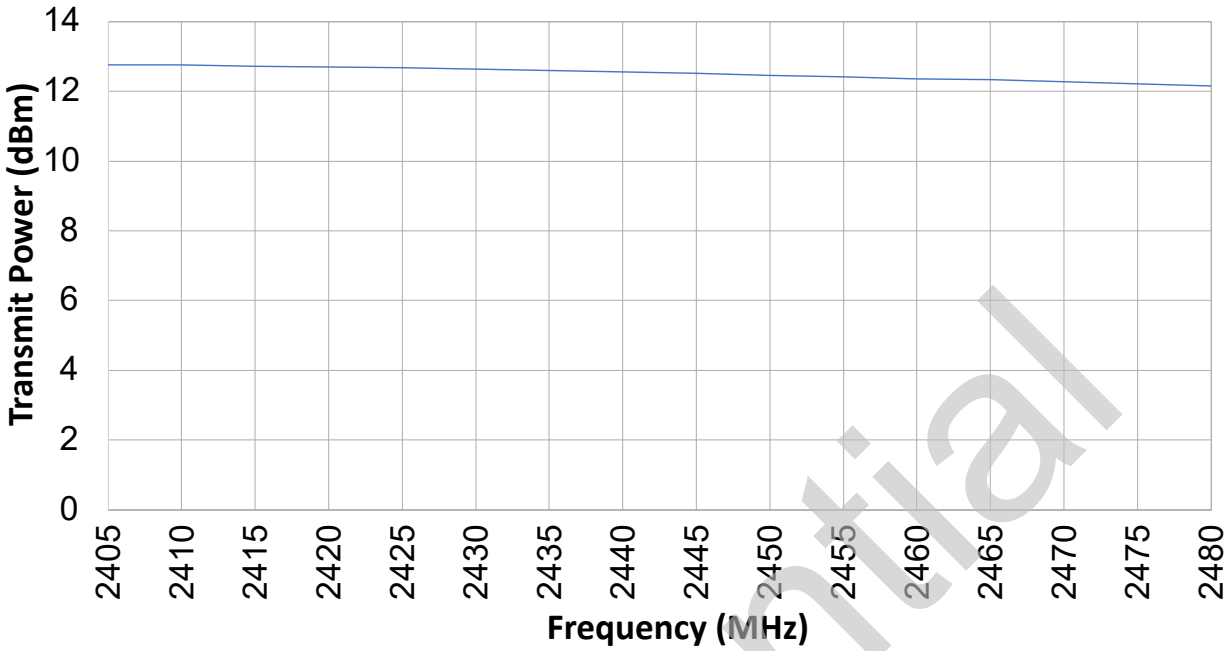
Figure 38-44. Zigbee Receive Current vs. Temperature



Notes:

- Receiver operating at 2440 MHz, 3.3V, 25℃ at maximum LNA gain.
- Measured after receiver calibration.

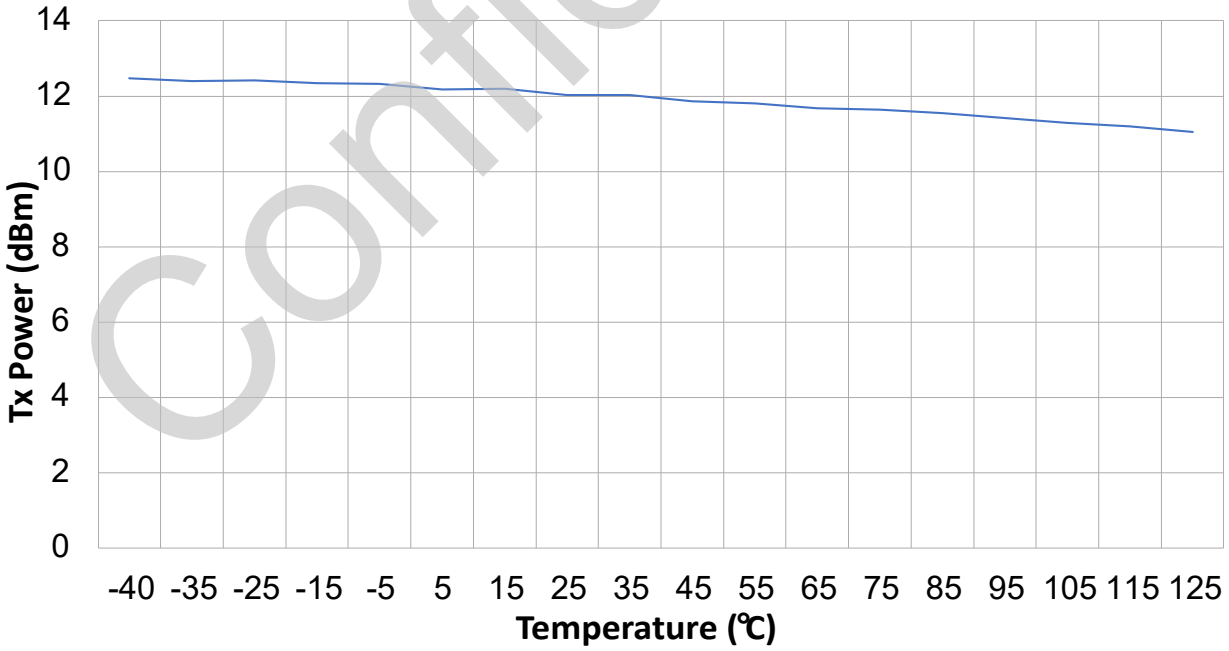
Figure 38-45. Zigbee Transmit Power vs Frequency



Notes:

- Transmit power is measured after calibration.
- Transmit power is measured across the channels at 3.6V at 25℃.
- Transmit power is measured after the PA matching and LPF.

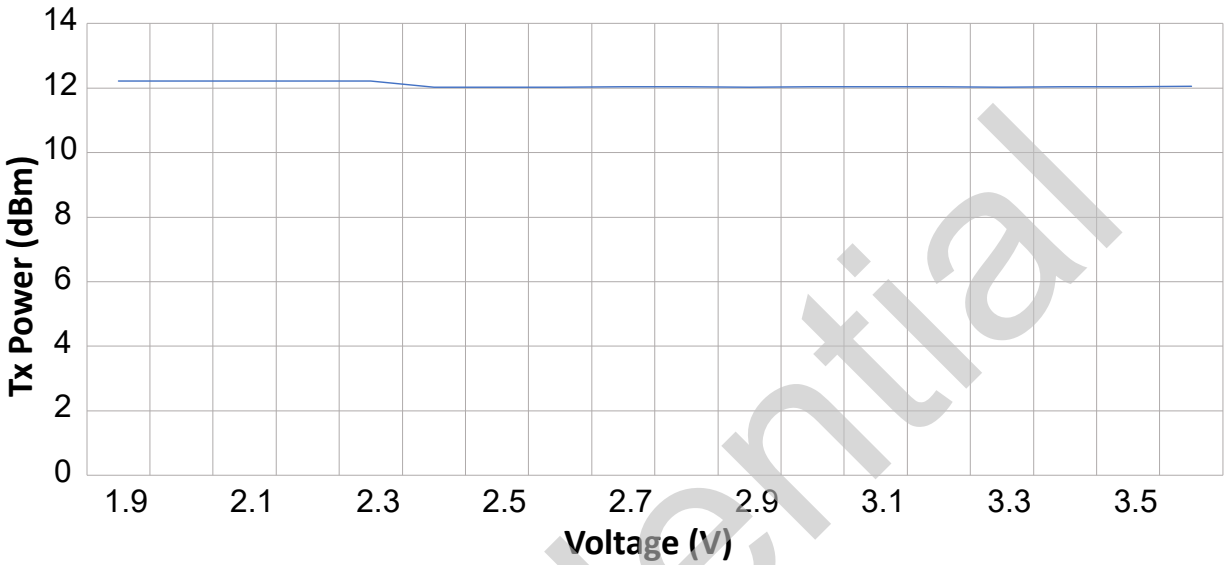
Figure 38-46. Zigbee Transmit Power vs Temperature



Notes:

- Transmit power is measured after calibration.
- Transmit power is measured at 2440 MHz at 3.6V across temperature.
- Transmit power is measured after the PA matching and LPF.
- Transmit power compensation is triggered before measurement across temperature.

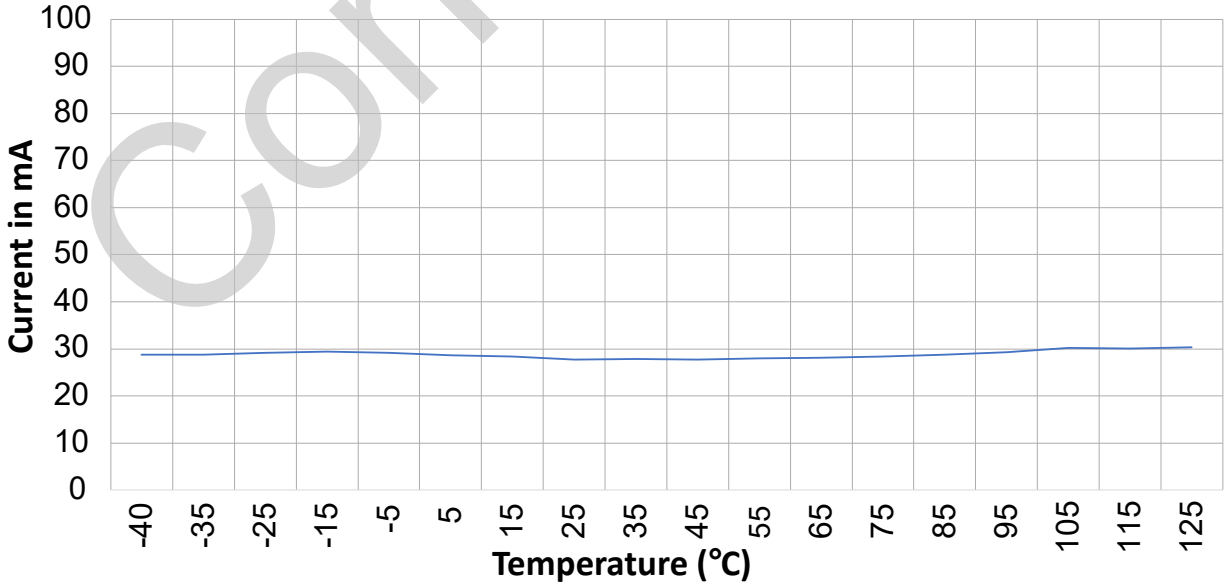
Figure 38-47. Zigbee Transmit Power vs. VDD Supply Voltage



Notes:

- Transmit power is measured after calibration.
- Transmit power is measured across voltage at 2440 MHz and 25°C.
- Transmit power is measured on reference board after the PA matching and LPF.
- Transmit power is configured using transmit power setting in HUT code.

Figure 38-48. Zigbee Transmit Current vs. Temperature



Notes:

- Transmit current is measured at input to SoC (includes SoC power consumption).
- Transmit current is measured at 2440 MHz at 3.3V (Buck mode).
- Transmit power is calibrated to +12 dBm (± 0.5 dBm on MPA mode).

39. Packaging Information

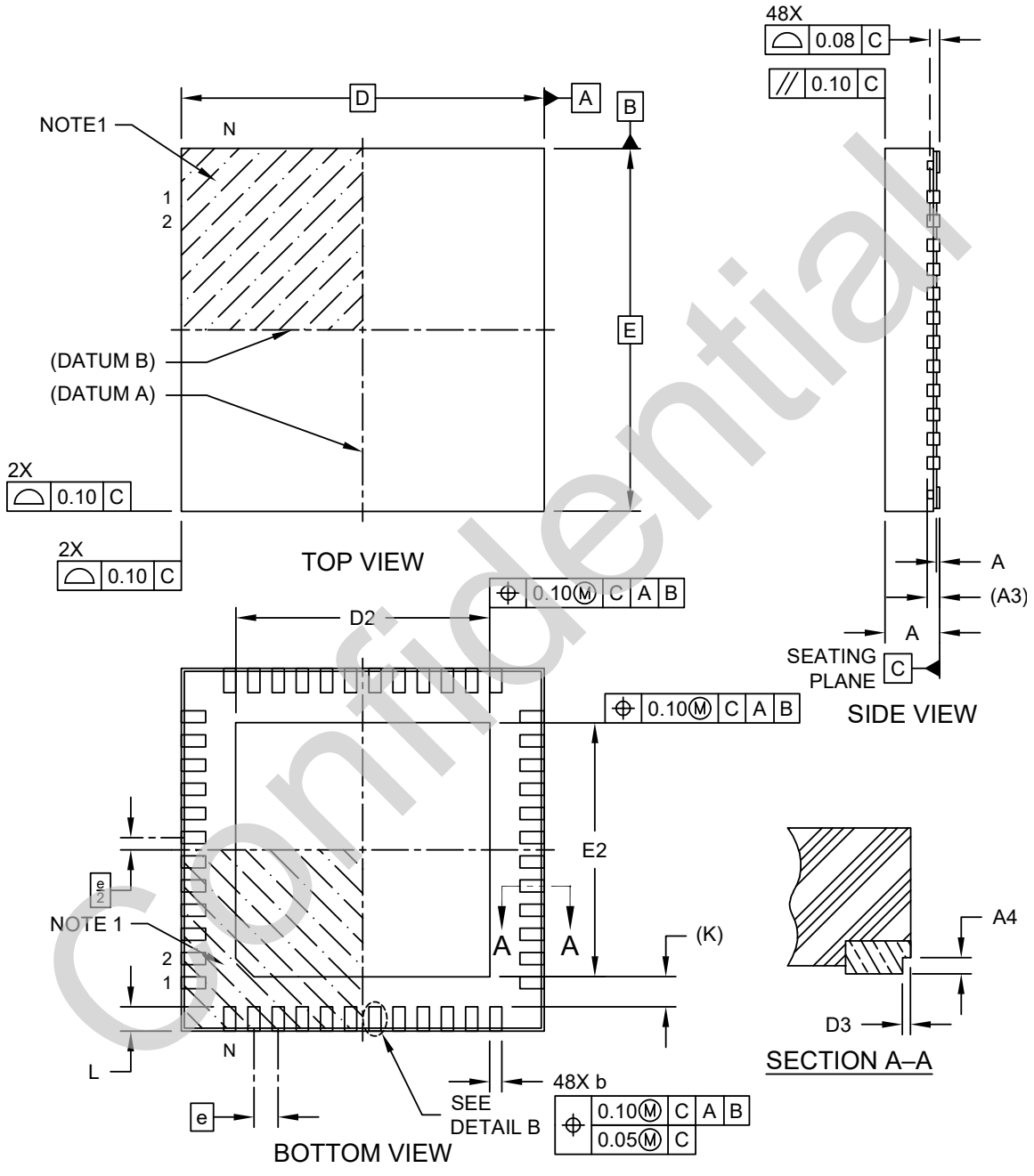
This chapter provides information on package markings, dimension and footprint of the PIC32CX-BZ3 and the WBZ35x family.

39.1 PIC32CX-BZ3 SoC Packaging Information

For the most current package drawings, see the Microchip Packaging Specification located at www.microchip.com/en-us/support/package-drawings.

48-Lead Very Thin Plastic Quad Flat, No Lead Package (ZWX) - 6x6x0.9 mm Body [VQFN] With 4.20 mm Exposed Pad and Stepped Wettable Flanks

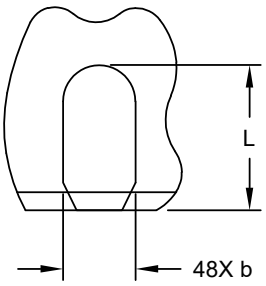
Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



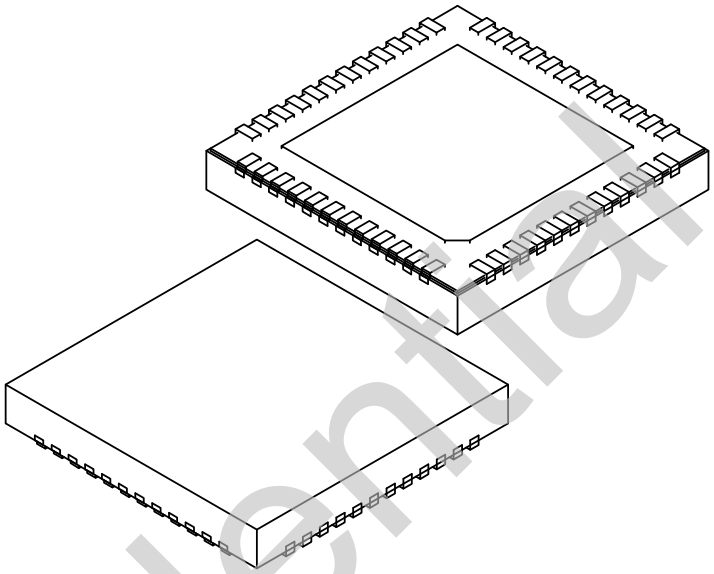
Microchip Technology Drawing C04-531 Rev B Sheet 1 of 2

48-Lead Very Thin Plastic Quad Flat, No Lead Package (ZWX) - 6x6x0.9 mm Body [VQFN] With 4.20 mm Exposed Pad and Stepped Wettable Flanks

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



DETAIL B
ALTERNATE TERMINAL CONFIGURATION



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Terminals	N		48	
Pitch	e		0.40 BSC	
Overall Height	A	0.80	0.85	0.90
Standoff	A1	0.00	0.02	0.05
Terminal Thickness	A3		0.203 REF	
Overall Length	D		6.00 BSC	
Exposed Pad Length	D2	4.10	4.20	4.30
Overall Width	E		6.00 BSC	
Exposed Pad Width	E2	4.10	4.20	4.30
Terminal Width	b	0.15	0.20	0.25
Terminal Length	L	0.35	0.40	0.45
Terminal-to-Exposed-Pad	K		0.50 REF	
Wettable Flank Step Cut Length	D3	-	-	0.085
Wettable Flank Step Cut Height	A4	0.10	-	0.19

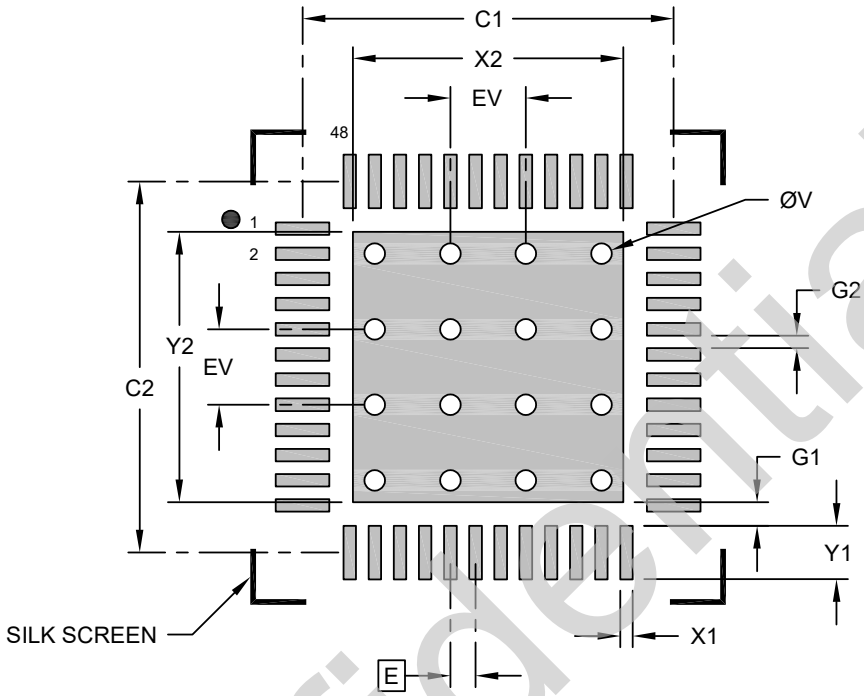
Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-531 Rev B Sheet 2 of 2

48-Lead Very Thin Plastic Quad Flat, No Lead Package (ZWX) - 6x6x0.9 mm Body [VQFN] With 4.20 mm Exposed Pad and Stepped Wettable Flanks

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.40 BSC	
Center Pad Width	X2			4.30
Center Pad Length	Y2			4.30
Contact Pad Spacing	C1		5.90	
Contact Pad Spacing	C2		5.90	
Contact Pad Width (Xnn)	X1			0.20
Contact Pad Length (Xnn)	Y1			0.85
Contact Pad to Center Pad (Xnn)	G1	0.38		
Contact Pad to Contact Pad (Xnn)	G2	0.20		
Thermal Via Diameter	V		0.33	
Thermal Via Pitch	EV		1.20	

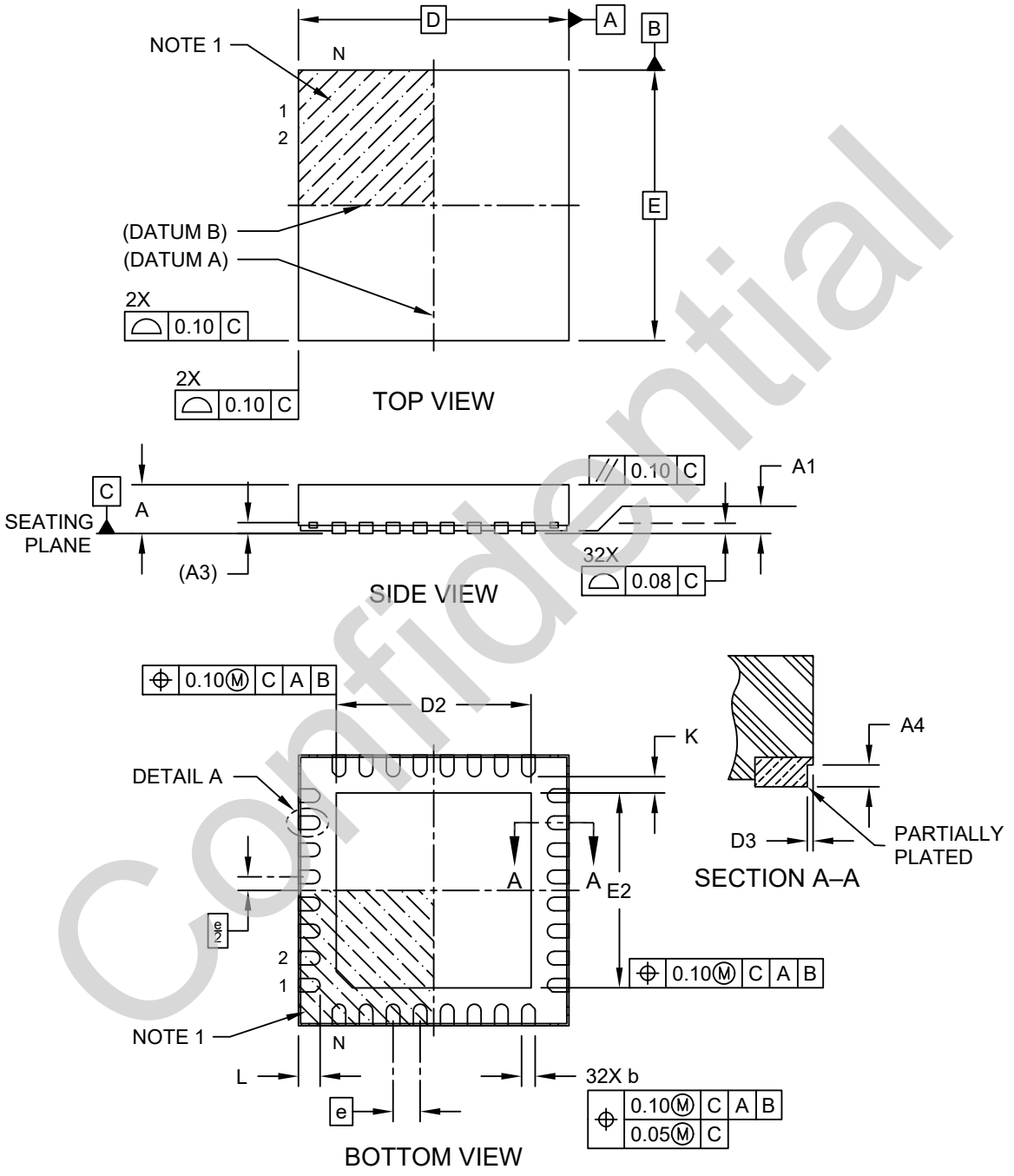
Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-2531 Rev B

32-Lead Very Thin Plastic Quad Flat, No Lead Package (RTB) - 5x5 mm Body [VQFN] With 3.6x3.6 mm Exposed Pad and Stepped Wettable Flanks; Atmel Legacy ZBS

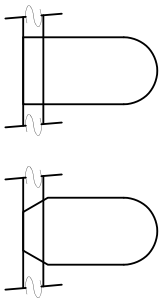
Note:	For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



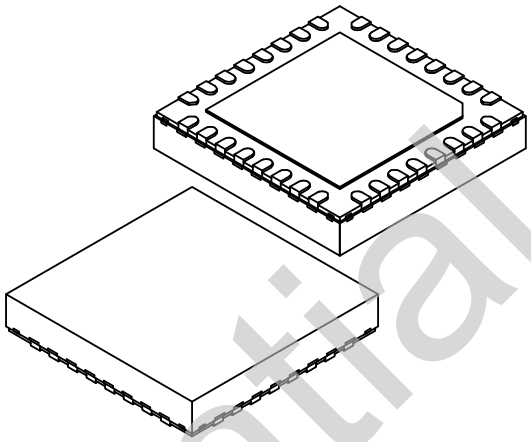
Microchip Technology Drawing C04-21391 Rev G Sheet 1 of 2

32-Lead Very Thin Plastic Quad Flat, No Lead Package (RTB) - 5x5 mm Body [VQFN] With 3.6x3.6 mm Exposed Pad and Stepped Wettable Flanks; Atmel Legacy ZBS

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



DETAIL 1
ALTERNATE TERMINAL CONFIGURATIONS



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Terminals	N		32	
Pitch	e		0.50 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.035	0.05
Terminal Thickness	A3		0.203 REF	
Overall Length	D		5.00 BSC	
Exposed Pad Length	D2	3.50	3.60	3.70
Overall Width	E		5.00 BSC	
Exposed Pad Width	E2	3.50	3.60	3.70
Terminal Width	b	0.20	0.25	0.30
Terminal Length	L	0.35	0.40	0.45
Terminal-to-Exposed-Pad	K	0.20	-	-
Wettable Flank Step Cut Width	D3	-	-	0.085
Wettable Flank Step Cut Depth	A4	0.10	-	0.19

Dimensions D3 and A4 above apply to all new products released after November 1, and all products shipped after January 1, 2019, and supersede dimensions D3 and A4 below.
No physical changes are being made to any package; this update is to align cosmetic and tolerance variations from existing suppliers.

Wettable Flank Step Length	D3	0.035	0.06	0.085
Wettable Flank Step Height	A4	0.10	-	0.19

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M

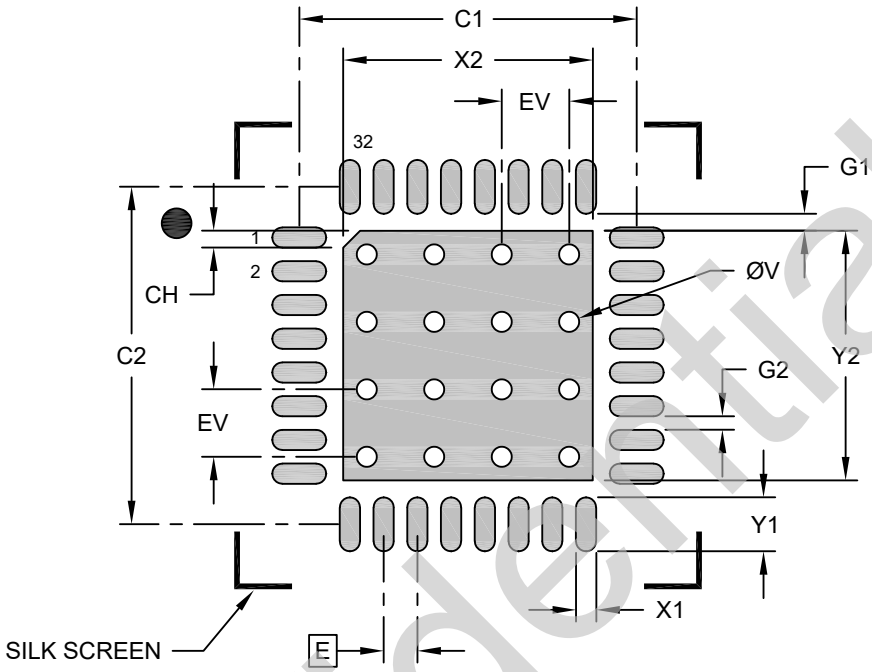
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-21391 Rev G Sheet 2 of 2

32-Lead Very Thin Plastic Quad Flat, No Lead Package (RTB) - 5x5 mm Body [VQFN] With 3.6x3.6 mm Exposed Pad and Stepped Wettable Flanks; Atmel Legacy ZBS

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.50 BSC	
Optional Center Pad Width	X2			3.70
Optional Center Pad Length	Y2			3.70
Exposed Pad 45° Corner Chamfer	CH		0.25	
Contact Pad Spacing	C1		5.00	
Contact Pad Spacing	C2		5.00	
Contact Pad Width (X32)	X1			0.30
Contact Pad Length (X32)	Y1			0.80
Contact Pad to Center Pad (X32)	G1	0.25		
Contact Pad to Contact Pad (X28)	G2	0.20		
Thermal Via Diameter	V		0.30	
Thermal Via Pitch	EV		1.00	

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-23391 Rev G

39.2 WBZ35x Module Packaging Information

The following images illustrate the packaging information of the WBZ351, which is a 39-lead PCB 15.5 mm x 20.7 mm x 2.8 mm with a metal shield and coaxial connector.

Figure 39-1. Dimensions and Top, Bottom and Side Views of WBZ351 Module

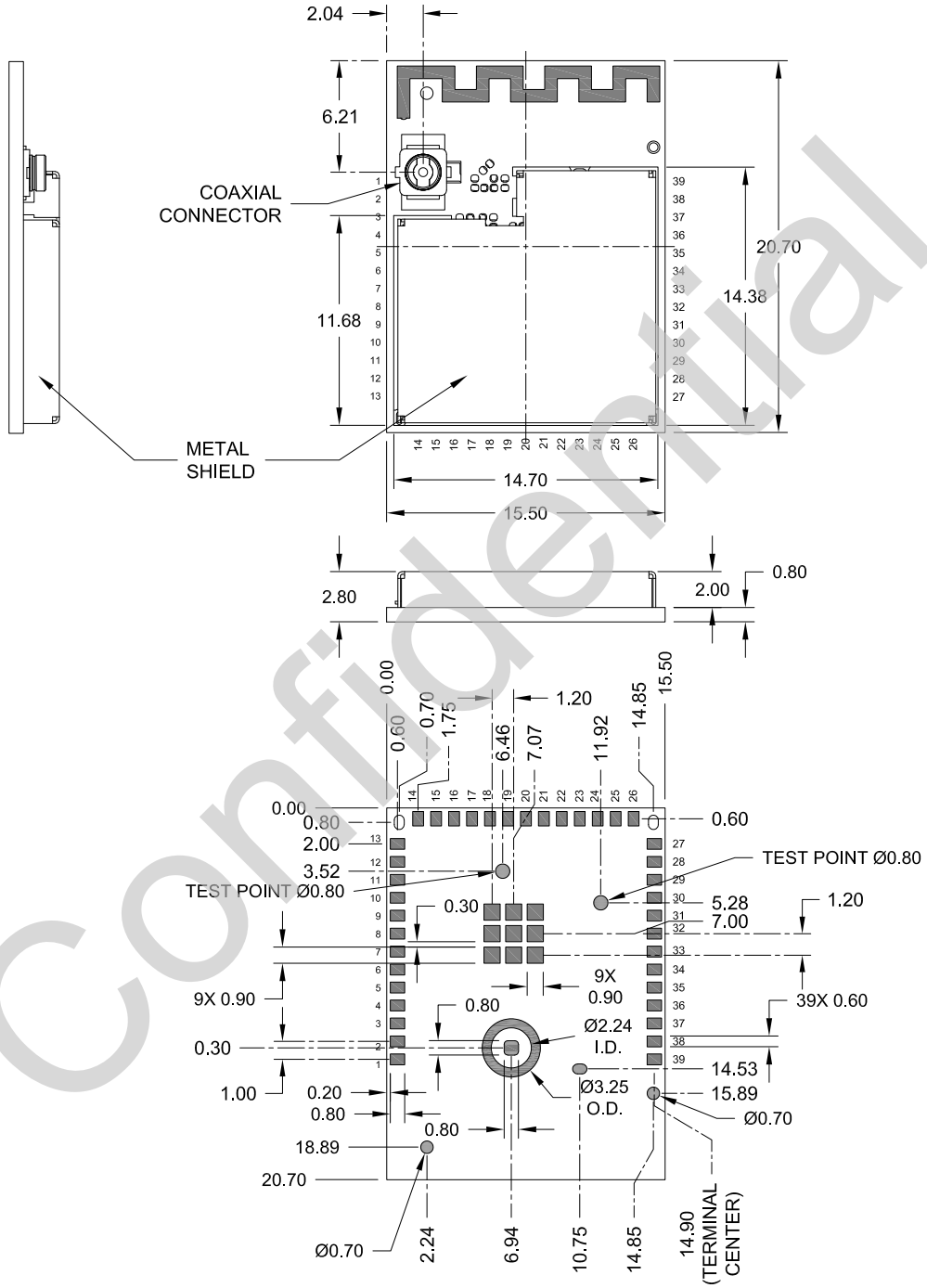


Figure 39-2. WBZ351 Module Package Outline Drawing

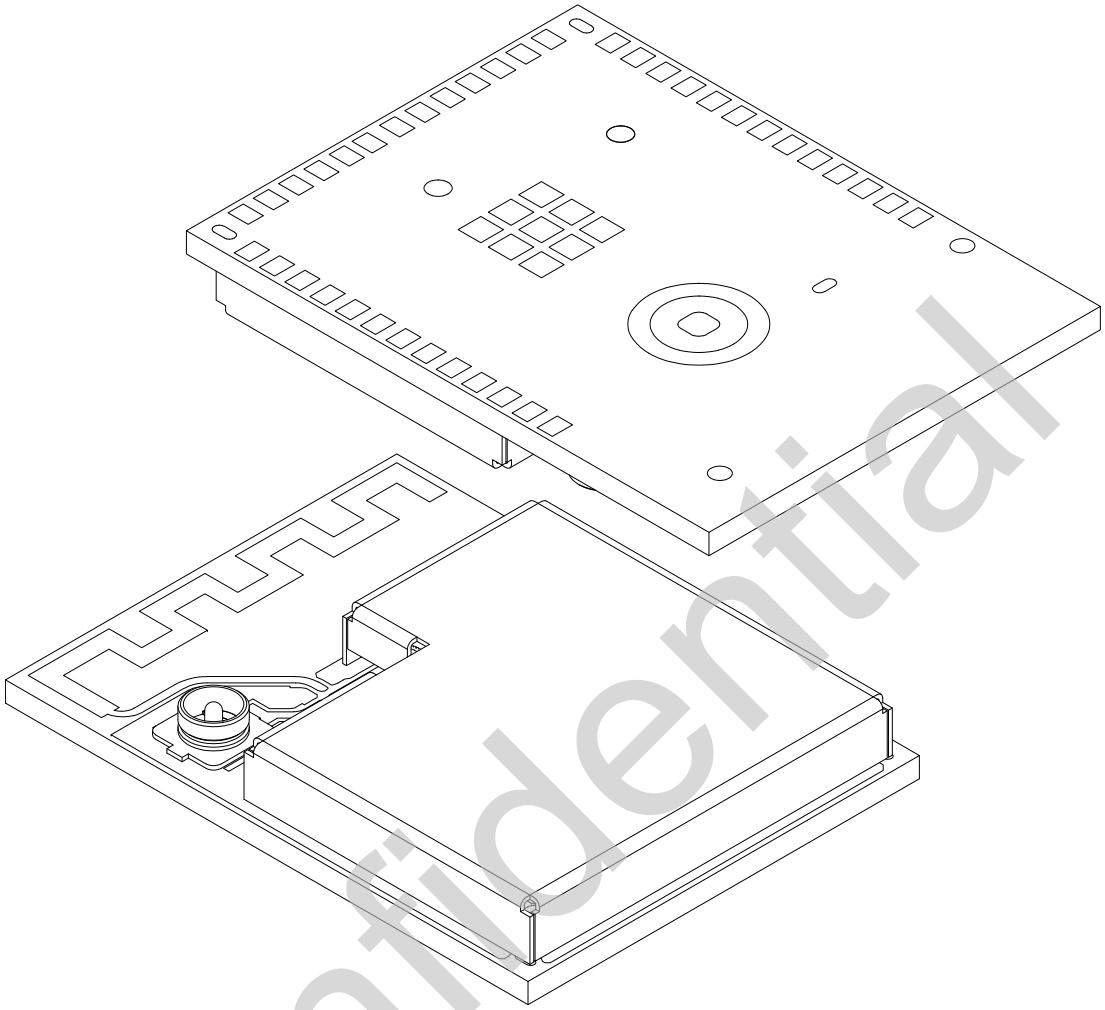
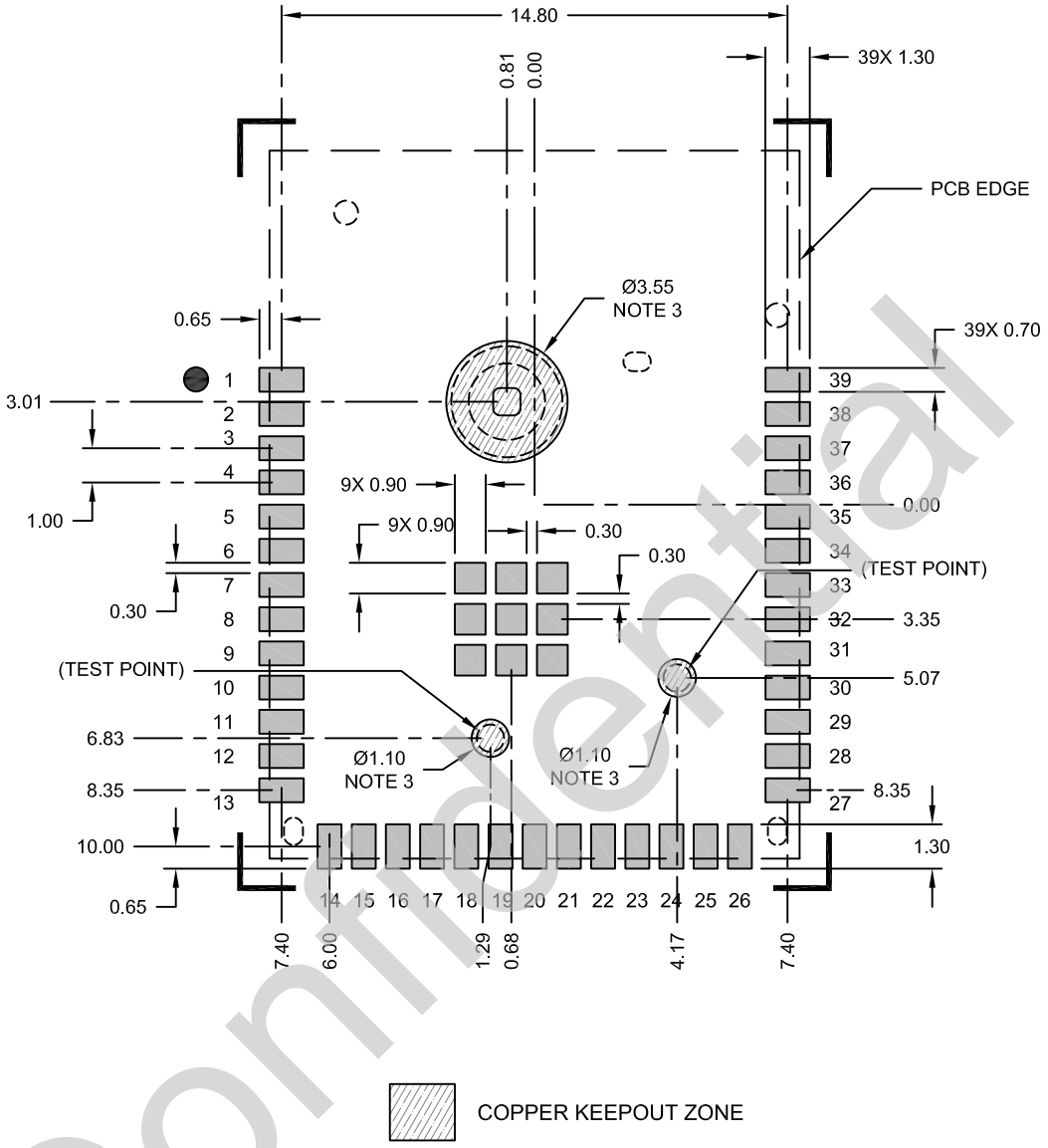


Figure 39-3. Recommended Land Pattern of WBZ351 Module



Notes:

1. All dimensions are in millimeters.
2. Keep this area free from all metal, including ground fill.
3. Keep these areas free from routes and exposed copper. Ground fill with solder mask may be placed here.

The following images illustrate the packaging information of the WBZ350, which is 30-lead PCB -13.4 mm x 18.7 mm x 2.8 mm with a metal shield and coaxial connector.

Figure 39-4. Top, Bottom, End and Side Views of WBZ350 Module

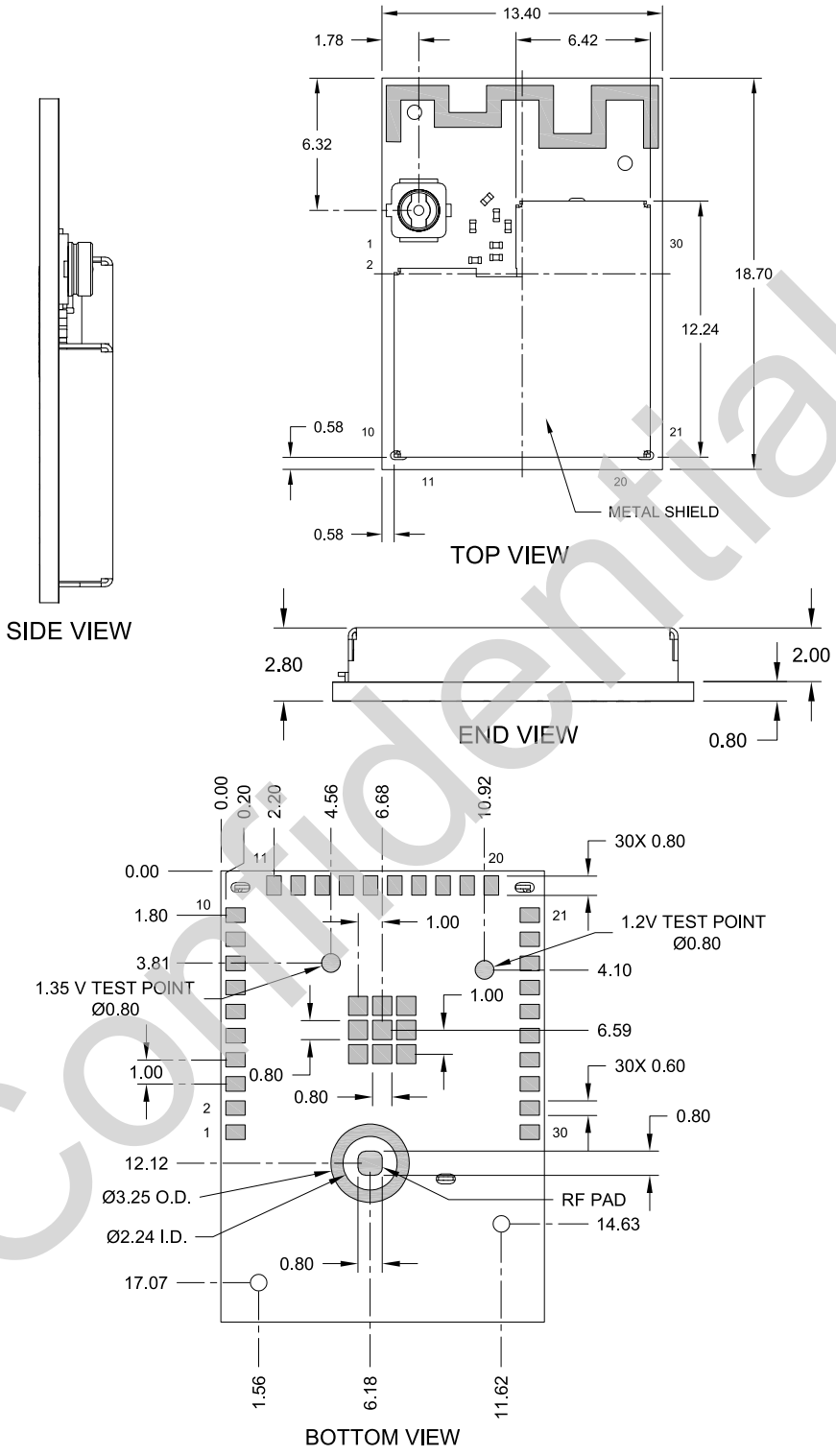
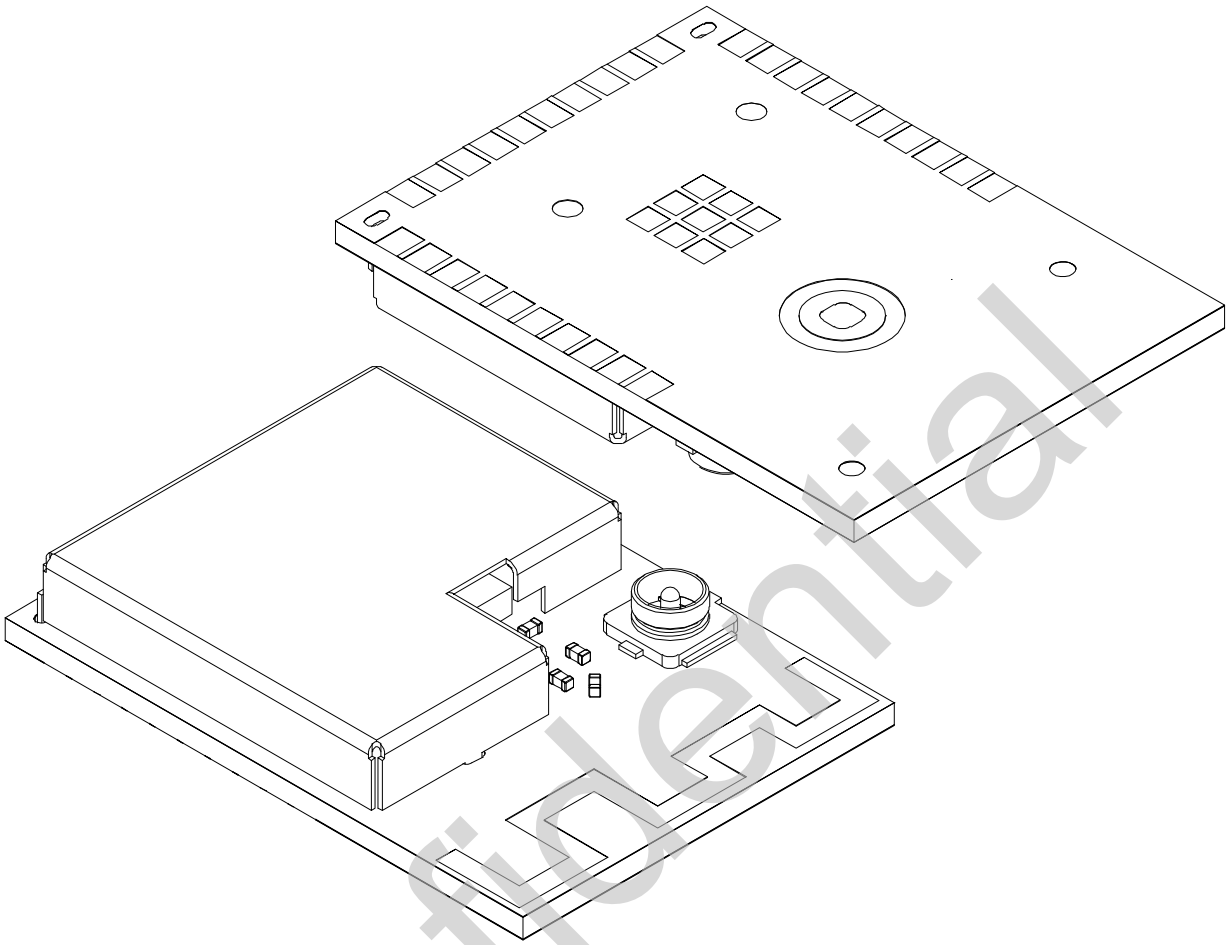


Figure 39-5. WBZ350 Module Package Outline Drawing



40. Document Revision History

Revision	Date	Section	Description
A	11/2022	Document	Initial revision

Microchip Information

The Microchip Website

Microchip provides online support via our website at www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to www.microchip.com/pcn and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip product is strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is "unbreakable". Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded

by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP "AS IS". MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, Adaptec, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, CryptoMemory, CryptoRF, dsPIC, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Kleer, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

AgileSwitch, APT, ClockWorks, The Embedded Control Solutions Company, EtherSynch, Flashtec, Hyper Speed Control, HyperLight Load, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet- Wire, SmartFusion, SyncWorld, Temux, TimeCesium, TimeHub, TimePictra, TimeProvider, TrueTime, and ZL are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, Augmented Switching, BlueSky, BodyCom, Clockstudio, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, Espresso T1S, EtherGREEN, GridTime, IdealBridge, In-Circuit Serial Programming, ICSP, INICnet, Intelligent Paralleling, IntelliMOS, Inter-Chip Connectivity, JitterBlocker, Knob-on-Display, KoD, maxCrypto, maxView, memBrain, Mindi, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PowerSmart, PureSilicon, QMatrix, REAL ICE, Ripple Blocker, RTAX, RTG4, SAM-ICE, Serial Quad I/O, simpleMAP, SimpliPHY, SmartBuffer, SmartHLS, SMART-I.S., storClad, SQI, SuperSwitcher, SuperSwitcher II, Switchtec, SynchroPHY, Total Endurance, Trusted Time, TSHARC, USBCheck, VariSense, VectorBlox, VeriPHY, ViewSpan, WiperLock, XpressConnect, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

The Adaptec logo, Frequency on Demand, Silicon Storage Technology, and Symmcom are registered trademarks of Microchip Technology Inc. in other countries.

GestIC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

ISBN:

Quality Management System

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
Corporate Office 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 Technical Support: www.microchip.com/support Web Address: www.microchip.com **Atlanta** Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455 **Austin, TX** Tel: 512-257-3370 **Boston** Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088 **Chicago** Itasca, IL Tel: 630-285-0071 Fax: 630-285-0075 **Dallas** Addison, TX Tel: 972-818-7423 Fax: 972-818-2924 **Detroit** Novi, MI Tel: 248-848-4000 **Houston, TX** Tel: 281-894-5983 **Indianapolis** Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380 **Los Angeles** Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800 **Raleigh, NC** Tel: 919-844-7510 **New York, NY** Tel: 631-435-6000 **San Jose, CA** Tel: 408-735-9110 Tel: 408-436-4270 **Canada - Toronto** Tel: 905-695-1980 Fax: 905-695-2078	**Australia - Sydney** Tel: 61-2-9868-6733 **China - Beijing** Tel: 86-10-8569-7000 **China - Chengdu** Tel: 86-28-8665-5511 **China - Chongqing** Tel: 86-23-8980-9588 **China - Dongguan** Tel: 86-769-8702-9880 **China - Guangzhou** Tel: 86-20-8755-8029 **China - Hangzhou** Tel: 86-571-8792-8115 **China - Hong Kong SAR** Tel: 852-2943-5100 **China - Nanjing** Tel: 86-25-8473-2460 **China - Qingdao** Tel: 86-532-8502-7355 **China - Shanghai** Tel: 86-21-3326-8000 **China - Shenyang** Tel: 86-24-2334-2829 **China - Shenzhen** Tel: 86-755-8864-2200 **China - Suzhou** Tel: 86-186-6233-1526 **China - Wuhan** Tel: 86-27-5980-5300 **China - Xian** Tel: 86-29-8833-7252 **China - Xiamen** Tel: 86-592-2388138 **China - Zhuhai** Tel: 86-756-3210040	**India - Bangalore** Tel: 91-80-3090-4444 **India - New Delhi** Tel: 91-11-4160-8631 **India - Pune** Tel: 91-20-4121-0141 **Japan - Osaka** Tel: 81-6-6152-7160 **Japan - Tokyo** Tel: 81-3-6880- 3770 **Korea - Daegu** Tel: 82-53-744-4301 **Korea - Seoul** Tel: 82-2-554-7200 **Malaysia - Kuala Lumpur** Tel: 60-3-7651-7906 **Malaysia - Penang** Tel: 60-4-227-8870 **Philippines - Manila** Tel: 63-2-634-9065 **Singapore** Tel: 65-6334-8870 **Taiwan - Hsin Chu** Tel: 886-3-577-8366 **Taiwan - Kaohsiung** Tel: 886-7-213-7830 **Taiwan - Taipei** Tel: 886-2-2508-8600 **Thailand - Bangkok** Tel: 66-2-694-1351 **Vietnam - Ho Chi Minh** Tel: 84-28-5448-2100	**Austria - Wels** Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 **Denmark - Copenhagen** Tel: 45-4485-5910 Fax: 45-4485-2829 **Finland - Espoo** Tel: 358-9-4520-820 **France - Paris** Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 **Germany - Garching** Tel: 49-8931-9700 **Germany - Haan** Tel: 49-2129-3766400 **Germany - Heilbronn** Tel: 49-7131-72400 **Germany - Karlsruhe** Tel: 49-721-625370 **Germany - Munich** Tel: 49-89-627-144-0 Fax: 49-89-627-144-44 **Germany - Rosenheim** Tel: 49-8031-354-560 **Israel - Ra'anana** Tel: 972-9-744-7705 **Italy - Milan** Tel: 39-0331-742611 Fax: 39-0331-466781 **Italy - Padova** Tel: 39-049-7625286 **Netherlands - Drunen** Tel: 31-416-690399 Fax: 31-416-690340 **Norway - Trondheim** Tel: 47-72884388 **Poland - Warsaw** Tel: 48-22-3325737 **Romania - Bucharest** Tel: 40-21-407-87-50 **Spain - Madrid** Tel: 34-91-708-08-90 Fax: 34-91-708-08-91 **Sweden - Gothenberg** Tel: 46-31-704-60-40 **Sweden - Stockholm** Tel: 46-8-5090-4654 **UK - Wokingham** Tel: 44-118-921-5800 Fax: 44-118-921-5820