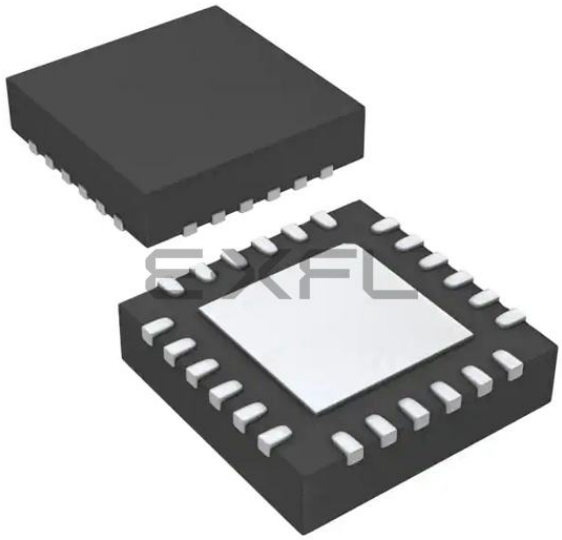


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	AVR
Core Size	8-Bit
Speed	16MHz
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, WDT
Number of I/O	22
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 12x10b; D/A 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	24-VFQFN Exposed Pad
Supplier Device Package	24-QFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/attiny417-mfr

5. I/O Multiplexing and Considerations

5.1 Multiplexed Signals

Table 5-1. PORT Function Multiplexing

VQFN 24-pin	Pin Name (1,2)	Other/Special	ADC0	PTC(3)	AC0	DAC	USART0	SPI0	TWI0	TCA0	TCB0	TCD0	CCL
23	PA0	$\overline{\text{RESET}}$ UPDI	AIN0										LUT0-IN0
24	PA1	BREAK	AIN1				TXD	MOSI	SDA				LUT0-IN1
1	PA2	EVOUT0	AIN2				RxD	MISO	SCL				LUT0-IN2
2	PA3	EXTCLK	AIN3				XCK	SCK		WO3			
3	GND												
4	VDD												
5	PA4		AIN4	X0/Y0			XDIR	$\overline{\text{SS}}$		WO4		WOA	LUT0-OUT
6	PA5		AIN5	X1/Y1	OUT					WO5	WO	WOB	
7	PA6		AIN6	X2/Y2	AINN0	OUT							
8	PA7		AIN7	X3/Y3	AINP0								LUT1-OUT
9	PB7												
10	PB6												
11	PB5	CLKOUT	AIN8		AINP1					WO2			
12	PB4		AIN9		AINN1					WO1			LUT0-OUT
13	PB3	TOSC1					RxD			WO0			
14	PB2	TOSC2, EVOUT1					TxD			WO2			
15	PB1		AIN10	X4/Y4			XCK		SDA	WO1			
16	PB0		AIN11	X5/Y5			XDIR		SCL	WO0			
17	PC0							SCK			WO	WOC	
18	PC1							MISO				WOD	LUT1-OUT
19	PC2	EVOUT2						MOSI					
20	PC3							$\overline{\text{SS}}$		WO3			LUT1-IN0
21	PC4	BREAK								WO4			LUT1-IN1
22	PC5									WO5			LUT1-IN2

Note:

1. Pins names are of type P*xn*, with *x* being the PORT instance (A,B) and *n* the pin number. Notation for signals is PORT*x*_PIN*n*. All pins can be used as event input.
2. All pins can be used for external interrupt, where pins P*x*2 and P*x*6 of each port have full asynchronous detection.
3. PTC is only available in devices with 8KB Flash (ATtiny817). Every PTC line can be configured as X-line or Y-line.

Tip: Signals on alternative pin locations are in `typewriter` font.

Bit	7	6	5	4	3	2	1	0
	CMPDEN	CMPCEN	CMPBEN	CMPAEN	CMPD	CMPC	CMPB	CMPA
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 4, 5, 6, 7 – CMPAEN, CMPBEN, CMPCEN, CMPDEN: Compare x Enable

Value	Description
0	Compare x output on Pin is disabled
1	Compare x output on Pin is enabled

Bits 0, 1, 2, 3 – CMPA, CMPB, CMPC, CMPD: Compare x

This bit selects the default state of Compare x after Reset, or when entering debug if FAULTDET is '1'.

Value	Description
0	Compare x default state is 0
1	Compare x default state is 1

Related Links

Register Summary - TCD

RSTCTRL - Reset Controller

6.9.4.5 System Configuration 0

Name: SYSCFG0
Offset: 0x05
Reset: 0xC4
Property: -

Bit	7	6	5	4	3	2	1	0
	CRCSRC[1:0]				RSTPINCFG[1:0]			EESAVE
Access	R	R			R	R		R
Reset	1	1			0	1		0

Bits 7:6 – CRCSRC[1:0]: CRC Source

See CRC description for more information about the functionality.

Value	Name	Description
00	FLASH	CRC of full Flash (boot, application code and application data)
01	BOOT	CRC of boot section
10	BOOTAPP	CRC of application code and boot sections
11	NOCRC	No CRC

Bits 3:2 – RSTPINCFG[1:0]: Reset Pin Configuration

These bits select the Reset/UPDI pin configuration.

Value	Description
0x0	GPIO
0x1	UPDI
0x2	RESET

2. Within four instructions, the software must execute the appropriate instruction. The protected change is immediately disabled if the CPU performs accesses to the Flash, NVMCTRL, or EEPROM, or if the SLEEP instruction is executed.

Once the correct signature is written by the CPU, interrupts will be ignored for the duration of the configuration change enable period. Any interrupt request (including non-maskable interrupts) during the CCP period will set the corresponding interrupt flag as normal, and the request is kept pending. After the CCP period is completed, any pending interrupts are executed according to their level and priority.

© 2017 Microchip Technology Inc.

Table 11-1. SLPCTRL System Dependencies

Dependency	Applicable	Peripheral
Clocks	Yes	CLKCTRL
I/O Lines and Connections	No	-
Interrupts	no	-
Events	No	-
Debug	Yes	UPDI

11.2.2.1 Clocks

This peripheral depends on the peripheral clock.

Related Links

CLKCTRL - Clock Controller

11.2.2.2 I/O Lines and Connections

Not applicable.

11.2.2.3 Interrupts

Not applicable.

11.2.2.4 Events

Not applicable.

11.2.2.5 Debug Operation

When run-time debugging, this peripheral will continue normal operation. The SLPCTRL is only affected by a break in debug operation: If the SLPCTRL is in a sleep mode when a break occurs, the device will wake up and the SLPCTRL will go to active mode, even if there are no pending interrupt requests.

If the peripheral is configured to require periodical service by the CPU through interrupts or similar, improper operation or data loss may result during halted debugging.

11.3 Functional Description

11.3.1 Initialization

To put the device into a sleep mode, follow these steps:

- Configure and enable the interrupts that should wake up the device from sleep. Also enable global interrupts.

  Warning: If there are no interrupts enabled when going to sleep, the device cannot wake up again. Only a reset will allow the device to continue operation.

- Select the sleep mode to be entered and enable the Sleep Controller by writing to the Sleep Mode bits (SMODE) and the Enable bit (SEN) in the Control A register (SLPCTRL.CTRLA). A SLEEP instruction must be run to make the device actually go to sleep.

16.3.2 Operation

16.3.2.1 Basic Functions

Each I/O pin Pxn can be controlled by the registers in PORT x. Each pin group x has its own set of PORT registers, the base address of the register set for pin n is at the byte address $\mathrm{PORT} + 0\mathrm{x}10 + n$. The index within that register set is n.

To use pin number n as an output only, write bit n of the PORT.DIR register to '1'. This can also be done by writing bit n in the PORT.DIRSET register to '1' - this will avoid disturbing the configuration of other pins in that group. The n^{th} bit in the PORT.OUT register must be written to the desired output value.

Similarly, writing a PORT.OUTSET bit to '1' will set the corresponding bit in the PORT.OUT register to '1'. Writing a bit in PORT.OUTCLR to '1' will clear that bit in PORT.OUT to zero. Writing a bit in PORT.OUTTGL or PORT.IN to '1' will toggle that bit in PORT.OUT.

To use pin n as an input, bit n in the PORT.DIR register must be written to '0' to disable the output driver. This can also be done by writing bit n in the PORT.DIRCLR register to '1' - this will avoid disturbing the configuration of other pins in that group. The input value can be read from bit n in register PORT.IN as long as the ISC bit is not set to INPUT_DISABLE.

Writing a bit to '1' in PORT.DIRTGL will toggle that bit in PORT.DIR, and toggle the direction of the corresponding pin.

16.3.2.2 Virtual Ports

The Virtual Port registers map the most frequently used regular Port registers into the bit-accessible I/O space. Writing to the Virtual Port registers has the same effect as writing to the regular registers, but allows for memory-specific instructions, such as bit-manipulation instructions, which are not valid for the extended I/O memory space where the regular Port registers reside.

Table 16-2. Virtual Port Mapping

Regular Port Register	Mapped to Virtual Port Register
PORT.DIR	VPORT.DIR
PORT.OUT	VPORT.OUT
PORT.IN	VPORT.IN
PORT.INTFLAG	VPORT.INTFLAG

Related Links

Register Summary - VPORT

I/O Multiplexing and Considerations

Peripherals and Architecture

16.3.2.3 Pin Configuration

The Pin n Configuration register (PORT.PINnCTRL) is used to configure inverted I/O, pullup, and input sensing of a pin.

All input and output on the respective pin n can be inverted by writing a '1' to the Inverted I/O Enable bit (INVEN) in PORT.PINnCTRL.

Toggling the INVEN bit causes an edge on the pin, which can be detected by all peripherals using this pin, and is seen by interrupts or Events if enabled.

Pullup of pin n is enabled by writing a '1' to the Pullup Enable bit (PULLUPEN) in PORT.PINnCTRL.

© 2017 Microchip Technology Inc.

16.5.2 Data Direction Set

Name: DIRSET
Offset: 0x01
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	DIRSET[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 – DIRSET[7:0]: Data Direction Set
This bit field can be used instead of a read-modify-write to set individual pins as output.
Writing a '1' to DIRSET[n] will set the corresponding PORT.DIR[n] bit.

Reading this bit field will always return the value of PORT.DIR.

16.5.3 Data Direction Clear

Name: DIRCLR
Offset: 0x02
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	DIRCLR[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 – DIRCLR[7:0]: Data Direction Clear
This register can be used instead of a read-modify-write to configure individual pins as input.
Writing a '1' to DIRCLR[n] will clear the corresponding bit in PORT.DIR.

Reading this bit field will always return the value of PORT.DIR.

16.5.4 Data Direction Toggle

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

An interrupt request is generated when the corresponding interrupt source is enabled and the Interrupt Flag is set. The interrupt request remains active until the Interrupt Flag is cleared. See the peripheral's INTFLAGS register for details on how to clear Interrupt Flags.

Related Links

AVR CPU

SREG

17.3.3 Sleep Mode Operation

There are two separate fuses defining the BOD configuration in different sleep modes: One fuse defines the mode used in Active mode and Idle sleep mode (ACTIVE in FUSE.BODCFG), and is written to the ACTIVE bits in the Control A register (BOD.CTRLA). The second fuse (SLEEP in FUSE.BODCFG) selects the mode used in Standby sleep mode and Power Down sleep mode, and is loaded into the SLEEP bits in the Control A register (BOD.CTRLA).

The operating mode in Active mode and Idle sleep mode (i.e. ACTIVE in BOD.CTRLA) cannot be altered by software. The operating mode in Standby sleep mode and Power Down sleep mode can be altered by writing to the SLEEP bits in the Control A register (BOD.CTRLA).

When the device is going into Standby sleep mode or Power Down sleep mode, the BOD will change operation mode as defined by SLEEP in BOD.CTRLA. When the device is waking up from Standby or Power Down sleep mode, the BOD will operate in the mode defined by the ACTIVE bit field in BOD.CTRLA.

17.3.4 Synchronization

Not applicable.

17.3.5 Configuration Change Protection

This peripheral has registers that are under Configuration Change Protection (CCP). In order to write to these, a certain key must be written to the CPU.CCP register first, followed by a write access to the protected bits within four CPU instructions.

It is possible to try writing to these registers any time, but the values are not altered.

The following registers are under CCP:

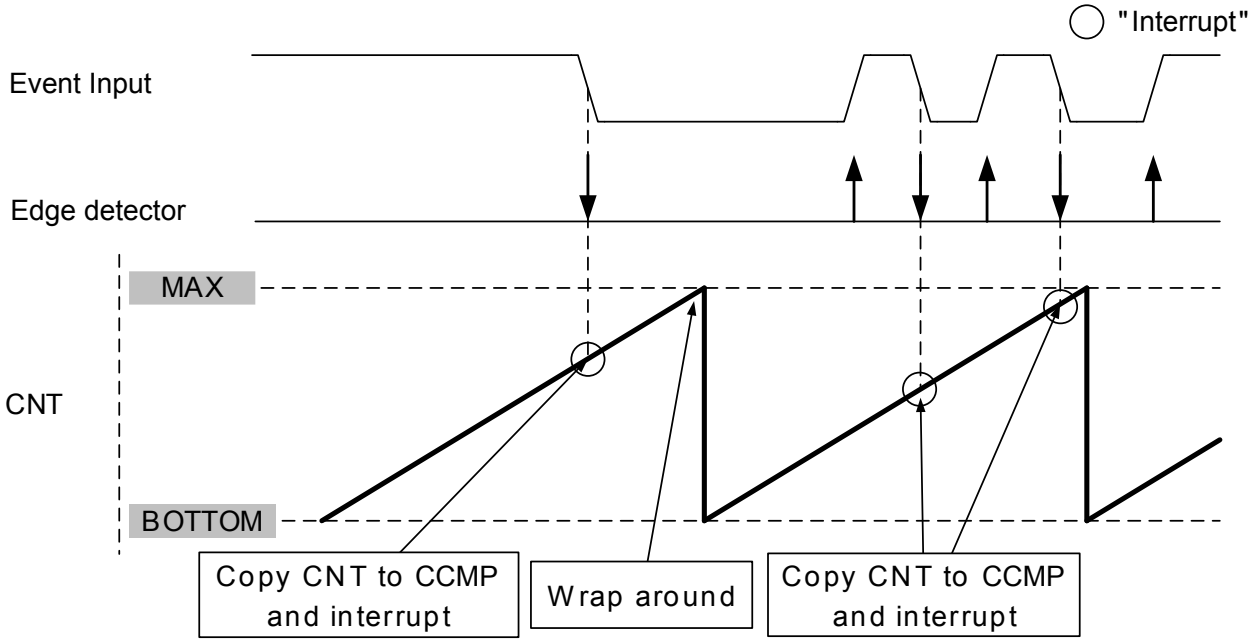
Table 17-3. Registers under Configuration Change Protection

Register	Key
SLEEP in BOD.CTRLA	IOREG

Related Links

Sequence for Write Operation to Configuration Change Protected I/O Registers

Figure 21-4. Input Capture on Event



It is recommended to write zero to the TCB.CNT register when entering this mode from any other mode.

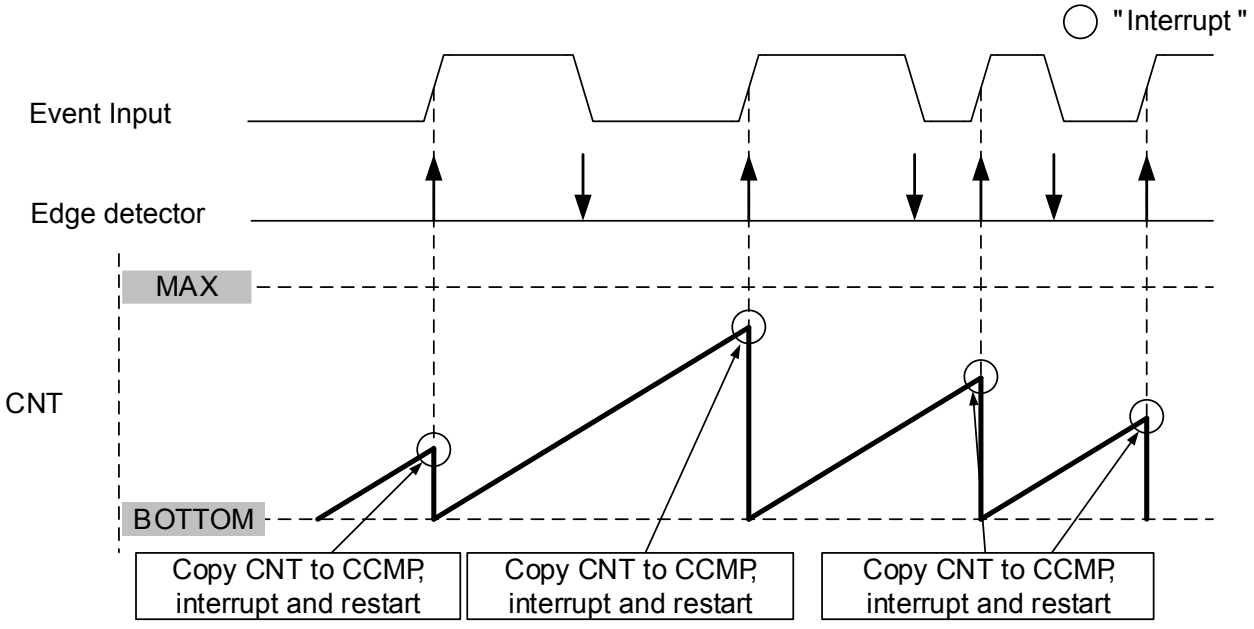
Input Capture Frequency Measurement Mode

In this mode, the TCB captures the counter value and restarts on either a positive or negative edge of the event input signal.

The interrupt flag is automatically cleared after the high byte of the Compare/Capture register (TCB.CCMP) has been read, and an interrupt request is generated.

The figure below illustrates this mode when configured to act on rising edge.

Figure 21-5. Input Capture Frequency Measurement



Value	Description
0	The peripheral is halted in break debug mode and ignores events.
1	The peripheral will continue to run in break debug mode when the CPU is halted.

21.5.8 Temporary Value

The Temporary register is used by the CPU for single-cycle, 16-bit access to the 16-bit registers of this peripheral. It can also be read and written by software. See also Accessing 16-bit Registers. There is one common Temporary register for all the 16-bit registers of this peripheral.

Name: TEMP
Offset: 0x09
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	TEMP[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 – TEMP[7:0]: Temporary Value

21.5.9 Count

The TCB.CNTL and TCB.CNTH register pair represents the 16-bit value TCB.CNT. The low byte [7:0] (suffix L) is accessible at the original offset. The high byte [15:8] (suffix H) can be accessed at offset + 0x01. For more details on reading and writing 16-bit registers, refer to Accessing 16-bit Registers.

CPU and UPDI write access has priority over internal updates of the register.

Name: CNT
Offset: 0x0A
Reset: 0x00
Property: -

Bit	15	14	13	12	11	10	9	8
	CNT[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
	CNT[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 – CNT[15:8]: Count Value high
These bits hold the MSB of the 16-bit counter register.

Bits 7:0 – CNT[7:0]: Count Value low
These bits hold the LSB of the 16-bit counter register.

Name	Description
Four ramp	Counter is reset to zero four times during a TCD cycle.
Dual ramp	Counter count both up and down between zero and selected top value.

22.3 Functional Description

22.3.1 Initialization and Disabling

To initialize the TCD:

1. Configure the static registers to the desired functionality.
2. Write desired initial values to the double-buffered registers.
3. Ensure that the Enable Ready bit (ENRDY) in the Status register (TCD.STATUS) is set to '1'.
4. Enable the TCD by writing a '1' to the ENABLE bit in the Control A register (TCD.CTRLA).

It is possible to disable the TCD in two different ways:

1. By writing a '0' to ENABLE in TCD.CTRLA. This disables the TCD instantly when synchronized to the TCD core domain.
2. By writing a '1' to the Disable at End of Cycle Strobe bit (DISEOC) in the Control E register (TCD.CTRLE). This disables the TCD at the end of the TCD cycle.

The bit fields in the TCD.CTRLA register are enable-protected, with exception of the ENABLE bit. They can only be written when ENABLE is written to '0' first.

Related Links

Register Synchronization Categories

22.3.2 Operation

22.3.2.1 Register Synchronization Categories

Most of the IO registers need to be synchronized to the asynchronous TCD core clock domain. This is done in different ways for different register categories:

- Command and Enable Control registers
- Doubled-buffered registers
- Static registers
- Normal IO and STATUS registers

See Table 22-3 for categorized registers.

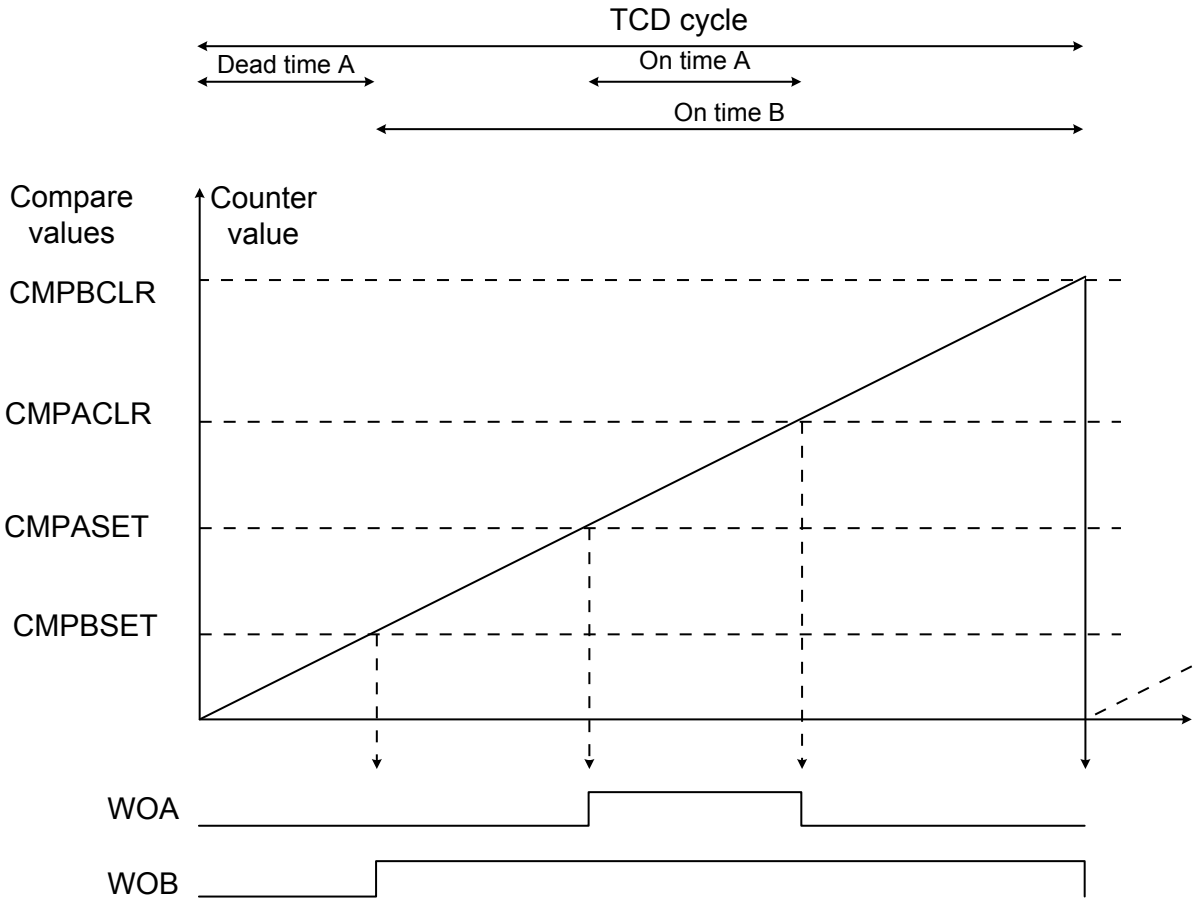
Command and Enable Registers

Because of synchronization between the clock domains it is only possible to change the Enable bits while the Enable Ready bit (ENRDY) in the Status register (TCD.STATUS) is '1'.

The Control E register commands (TCD.CTRLE) are automatically synchronized to the TCD core domain when the TCD is enabled and as long as there not a synchronization ongoing already. Check in the Status register if the Command Ready bit (CCMDRDY) is '1' (TCD.STATUS) to ensure that it is possible to write a new command. TCD.CTRLE is a strobe register that will clear itself when the command is done.

The Control E register commands are:

Figure 22-4. One Ramp Mode with *CMPBSET < CMPASET*



If any of the other compare values are bigger than CMPBCLR it will never be triggered when running in One ramp mode. And if The CMPACLR is smaller than the CMPASET value, the clear value will not have any effect.

Two Ramp Mode

In Two Ramp Mode the TCD counter counts up until it reaches the CMPACLR value, then it resets and counts up until it reaches the CMPBCLR value. Then, the TCD cycle is done and the counter restarts from 0x000, beginning a new TCD cycle. The TCD cycle period is give by:

$$T_{\mathrm{TCD\_cycle}} = \frac{(\mathrm{CMPACLR} + 1 + \mathrm{CMPBCLR} + 1)}{f_{\mathrm{CLK\_TCD\_CNT}}}$$

The TCD counter value is synchronized to CAPTUREx by either software or an event.

The capture register is blocked for update of new capture data until TCDn.CAPTURExH is read.

Name: CAPTUREA, CAPTUREB
Offset: 0x22 + n*0x02 [n=0..1]
Reset: 0x00
Property: -

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

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

Bits 11:0 – CAPTURE[11:0]: Capture Byte

22.5.18 Compare Set x

For compare operation, these registers are continuously compared to the counter value. Normally, the outputs form the comparators are then used for generating waveforms.

Name: CMPASET, CMPBSET
Offset: 0x28 + n*0x04 [n=0..1]
Reset: 0x00
Property: -

Bit	15	14	13	12	11	10	9	8
					CMPSET[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
	CMPSET[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 – CMPSET[11:0]: Compare Set
These bits hold value of the compare register.

22.5.19 Compare Clear x

For compare operation, these registers are continuously compared to the counter value. Normally, the outputs form the comparators are then used for generating waveforms.

The RTC.PERL and RTC.PERH register pair represents the 16-bit value, RTC.PER. The low byte [7:0] (suffix L) is accessible at the original offset. The high byte [15:8] (suffix H) can be accessed at offset + 0x01. For more details on reading and writing 16-bit registers, refer to Accessing 16-bit Registers.

Due to synchronization between the RTC clock and system clock domains, there is a latency of two RTC clock cycles from updating the register until this has an effect. Application software needs to check that the STATUS.PERBUSY flag is cleared before writing to this register.

Name: PER
Offset: 0x0A
Reset: 0xFF
Property: -

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	1	1	1	1	1	1	1	1

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	1	1	1	1	1	1	1	1

Bits 15:8 – PER[15:8]: Period high byte
These bits hold the MSB of the 16-bit period register.

Bits 7:0 – PER[7:0]: Period low byte
These bits hold the LSB of the 16-bit period register.

23.11.10 Compare

The RTC.CMPL and RTC.CMPH register pair represents the 16-bit value, RTC.CMP. The low byte [7:0] (suffix L) is accessible at the original offset. The high byte [15:8] (suffix H) can be accessed at offset + 0x01. For more details on reading and writing 16-bit registers, refer to Accessing 16-bit Registers.

Name: CMP
Offset: 0x0C
Reset: 0x00
Property: -

- UCPHA bit functionality is identical to that of the SPI CPHA bit
- UDORD bit functionality is identical to that of the SPI DORD bit

When the USART is set in master SPI mode, configuration and use are in some cases different from those of the standalone SPI module. In addition, the following difference exists:

- The USART in master SPI mode does not include the SPI (Write Collision) feature

The USART in master SPI mode does not include the SPI double speed mode feature, but this can be achieved by configuring the baud rate generator accordingly:

- Interrupt timing is not compatible
- Pin control differs due to the master-only operation of the USART in SPI master mode

A comparison of the USART in master SPI mode and the SPI pins is shown in Table 24-6.

Table 24-6. Comparison of USART in master SPI mode and SPI pins.

USART	SPI	Comment
TxD	MOSI	Master out only
RxD	MISO	Master in only
XCK	SCK	Functionally identical
-	SS	Not supported by USART in master SPI mode

Related Links

CTRLC

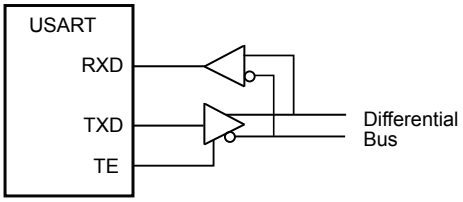
24.3.2.6 RS485 Mode of Operation

The RS485 feature enables the support of external components to comply with the RS-485 standard.

Either an external line driver is supported as shown in the figure below (RS485=0x1 in USART.CTRLA), or control of the transmitter driving the TxD pin is provided (RS485=0x2).

While operating in RS485 mode, the Transmit Enable pin (TE) is driven high when the transmitter is active.

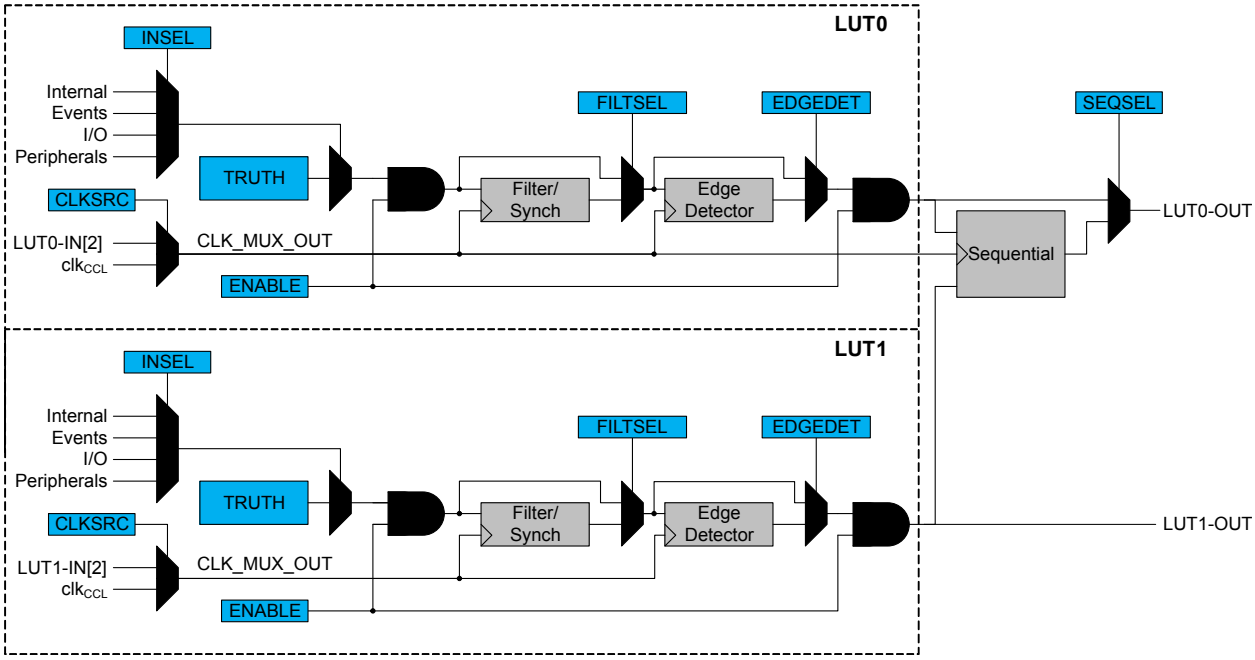
Figure 24-10. RS485 Bus Connection



The TE pin goes high one baud clock cycle in advance of data being shifted out, to allow some guard time to enable the external line driver. The TE pin will remain high for the complete frame including stop bit(s).

28.2.1 Block Diagram

Figure 28-1. Configurable Custom Logic



28.2.2 Signal Description

Pin Name	Type	Description
LUTn-OUT	Digital output	Output from lookup table
LUTn-IN[2:0]	Digital input	Input to lookup table

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

Related Links

I/O Multiplexing and Considerations

28.2.3 System Dependencies

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

Table 28-1. CCL System Dependencies

Dependency	Applicable	Peripheral
Clocks	Yes	CLKCTRL
I/O Lines and Connections	Yes	PORT
Interrupts	Yes	CPUINT
Events	Yes	EVSYS
Debug	Yes	UPDI

28.2.3.1 Clocks

By default, the CCL is using the peripheral clock of the device (CLK_PER).

29.4 Register Summary - AC

Offset	Name	Bit Pos.								
0x00	CTRLA	7:0	RUNSTDBY	OUTEN	INTMODE[1:0]		LPMODE	HYSMODE[1:0]		ENABLE
0x01	Reserved									
0x02	MUXCTRLA	7:0	INVERT				MUXPOS		MUXNEG[1:0]	
0x03 ... 0x05	Reserved									
0x06	INTCTRL	7:0								CMP
0x07	STATUS	7:0				STATE				CMP

29.5 Register Description

29.5.1 Control A

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

Bit	7	6	5	4	3	2	1	0
	RUNSTDBY	OUTEN	INTMODE[1:0]		LPMODE	HYSMODE[1:0]		ENABLE
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 – RUNSTDBY: Run in Standby Mode
Writing a '1' to this bit allows the AC to continue operation in Standby sleep mode. Since the clock is stopped, interrupts and status flags are not updated.

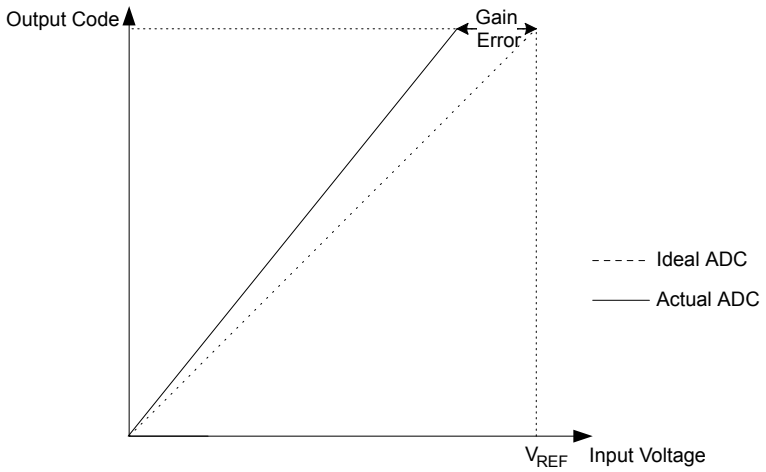
Value	Description
0	In Standby sleep mode, the peripheral is halted
1	In Standby sleep mode, the peripheral continues operation

Bit 6 – OUTEN: Analog Comparator Output Pad Enable
Writing this bit to '1' makes the OUT signal available on the pin.

Bits 5:4 – INTMODE[1:0]: Interrupt Modes
Writing to these bits selects what edge(s) of the AC output an interrupt request is triggered.

Value	Name	Description
0x0	BOTHEDGE	Both negative and positive edge
0x1	-	Reserved
0x2	NEGEDGE	Negative edge
0x3	POSEDGE	Positive edge

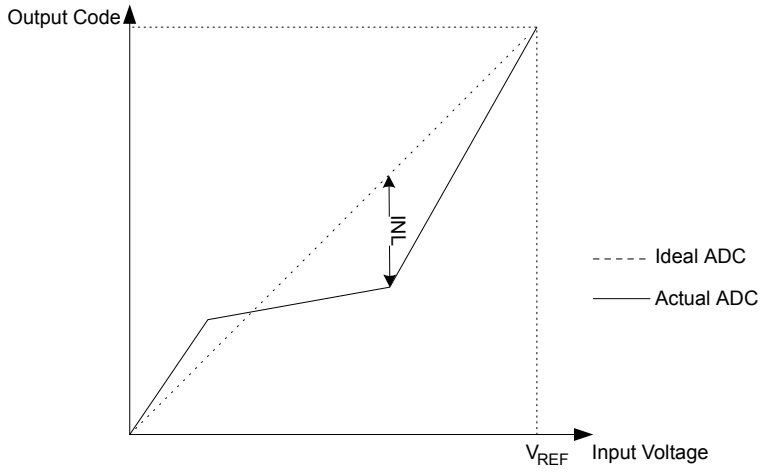
Figure 30-3. Gain Error



Integral Non-Linearity (INL)

After adjusting for offset and gain error, the INL is the maximum deviation of an actual transition compared to an ideal transition for any code. Ideal value: 0 LSB.

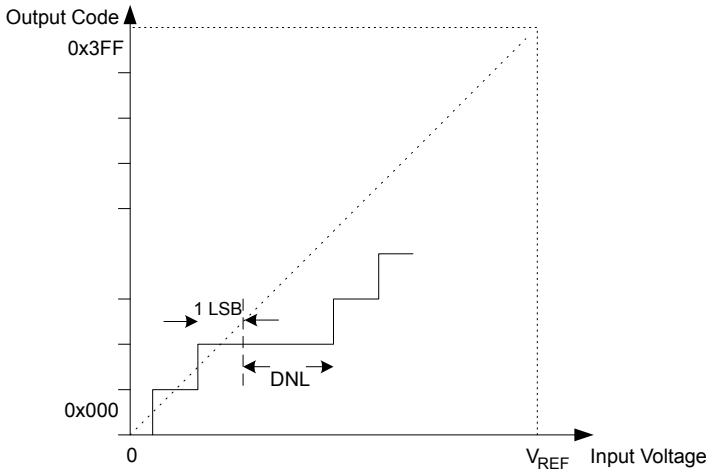
Figure 30-4. Integral Non-Linearity



Differential Non-Linearity (DNL)

The maximum deviation of the actual code width (the interval between two adjacent transitions) from the ideal code width (1 LSB). Ideal value: 0 LSB.

Figure 30-5. Differential Non-Linearity



34.7 BOD and POR Characteristics

Table 34-8. Power Supply Characteristics

Symbol	Description	Condition	Min.	Typ.	Max.	Unit
SRON	Power-on Slope		-	-	100	V/ms

Table 34-9. Power On Reset (POR) Characteristics

Symbol	Description	Condition	Min.	Typ.	Max.	Unit
V_{POR}	POR threshold voltage on V_{DD} falling	V_{DD} falls/rises at 0.5V/ms or slower	0.8	-	1.6	V
	POR threshold voltage on V_{DD} rising		1.4	-	1.8	

Table 34-10. Brownout Detection (BOD) Characteristics

Symbol	Description	Condition	Min.	Typ.	Max.	Unit
V_{BOD}	BOD triggering level (falling/rising)	BODLEVEL7	3.9	4.2	4.5	V
		BODLEVEL2	2.4	2.6	2.9	
		BODLEVEL0	1.7	1.8	2.0	
V_{INT}	Interrupt level 0	Percentage above the selected BOD level	-	4	-	%
	Interrupt level 1		-	13	-	
	Interrupt level 2		-	25	-	
V_{HYS}	Hysteresis	BODLEVEL7	-	80	-	mV
		BODLEVEL2	-	40	-	
		BODLEVEL0	-	25	-	
T_{BOD}	Detection time	Continuous	-	7	-	µs
		Sampled, 1kHz	-	1	-	ms
		Sampled, 125Hz	-	8	-	
T_{start}	Start-up time	Time from enable to ready	-	40	-	µs

34.8 External Reset Characteristics

Table 34-11. External Reset Characteristics

Mode	Description	Condition	Min.	Typ.	Max.	Unit
V_{HVRST}	$\overline{\text{RESET}}$ pin threshold for High-Voltage Reset		11.5	-	12.5	V
V_{RST_VIH}	Input High Voltage for $\overline{\text{RESET}}$		$0.8 \times V_{DD}$	-	$V_{DD}+0.2$	
V_{RST_VIL}	Input Low Voltage for $\overline{\text{RESET}}$		-0.2	-	$0.2 \times V_{DD}$	

40.1.1.3 TCB

1 –

TCB interrupt flag cleared when reading CCMPH

TCBn.INTFLAGS.CAPT is cleared when reading TCBn.CCMPH instead of CCMPL.

Fix/Workaround:

Read both TCBn.CCMPL and TCBn.CCMPH.

2 –

TCB input capture frequency and pulse-width measurement mode not working with prescaled clock

The TCB input capture frequency and pulse-width measurement mode may lock to freeze state if CLKSEL in TCB.CTRLA is set to other value than 0x0.

Fix/Workaround:

Only use CLKSEL equal to 0x0 when using Input capture frequency and pulse-width measurement mode.

40.1.1.4 TWI

1 –

TIMEOUT bits in TWI.MCTRLB register is not accessible

The TIMEOUT bits in TWI.MCTRLB register is not accessible from software.

Fix/Workaround:

When initializing TWI, BUSSTATE in TWI.MSTATUS should be brought into IDLE state by writing 0x1 to it.

2 –

TWI Smart Mode gives extra clock pulse

TWI Master with Smart Mode enabled gives an extra clock pulse on SCL line after sending NACK.

Fix/Workaround:

None.

3 –

TWI Master Mode wrongly detect start bit as a stop bit

If TWI is enabled in master mode followed by an immediate write to the MADDR register, the bus monitor recognize the start bit as a stop bit.

Fix/Workaround:

Wait minimum two clock cycles from TWI.MCTRLA.ENABLE until TWI.MADDR is written.

4 –

TWI Master Enable Quick Command not accessible

TWI.MCTRLA.QCEN is not accessible from software.

Fix/Workaround:

None.

40.1.1.5 USART

1 –

Frame error on previous message may cause false start bit detection

If receiving a frame with RXDATAH.FERR set and reading the RXDATAL before the RxD line goes high, will trigger a false start bit detection.

Fix/Workaround:

41.4 Rev.B - 11/2016

Section	Changes
Document	• Section Conventions added. • Peripheral sections: document structure updated. • Presentation of 16-bit registers updated. • Editorial updates throughout the document.
Pinout	• Updated diagrams, legends.
SYSCFG - System Configuration	• Content on debugging removed; see UPDI. • Moved into section Memories.
I/O Multiplexing and Considerations	• Signal names updated throughout the document.
AVR CPU	• Invalid registers removed.
CLKCTRL - Clock Controller	• Bit field in peripherals for clock selection is CLKSEL. • CLKCTRL.XOSC32KCTRLA is under Configuration Change Protection.
FUSES - Configuration and User Fuses	• OSCLOCK in FUSE.OSCCFG is loaded to LOCK in CLKCTRL.OSC20MCALIBB. • SIGROW.TCD0CFG[7:4] - bit field names updated.
NVMCTRL - Non Volatile Memory Controller	• Flash sections re-organized. • Command value expected within four instructions.
SLPCTRL - Sleep Controller	• Abbreviation SLPCTRL. • Behavior of peripherals in sleep modes updated.
CPUINT - CPU Interrupt Controller	• CPUINT.LVL0PRI has no effect when Round Robin disabled.
EVSYS - Event System	• CLKCTRL is no event user/generator. • Signals EVOUT[2:0] added.
BOD - Brownout Detector	• Register at 0x08 is BOD.VLMCTRLA.
TCA - 16-bit Timer/Counter Type A	• Definitions updated. • TCA clock is CLK_PER. • CMPnOV in TCA.CTRLC.
TCB - 16-bit Timer/Counter Type B	• Definitions updated.