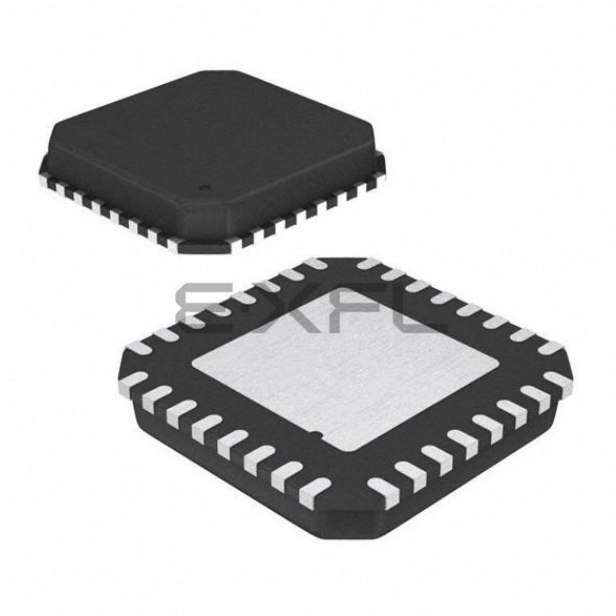


Welcome to E-XFL.COM

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	20MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	28
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	1.7V ~ 5.5V
Data Converters	A/D 28x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	32-VFQFN Exposed Pad
Supplier Device Package	32-VQFN (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/attiny828r-mur

To ensure stable operation of the MCU it is required to avoid sudden changes in the external clock frequency . A variation in frequency of more than 2% from one clock cycle to the next can lead to unpredictable behavior. It is required to ensure that the MCU is kept in Reset during such changes in the clock frequency.

Stable operation for large step changes in system clock frequency is guaranteed when using the system clock prescaler. See “System Clock Prescaler” on page 30.

6.2.2 Calibrated Internal 8MHz Oscillator

The internal 8MHz oscillator operates with no external components and, by default, provides a clock source with an approximate frequency of 8MHz. Though voltage and temperature dependent, this clock can be very accurately calibrated by the user. See Table 104 on page 249 and “Internal Oscillator Speed” on page 293 for more details.

During reset, hardware loads the pre-programmed calibration value into the OSCCAL0 register and thereby automatically calibrates the oscillator. The accuracy of this calibration is referred to as “Factory Calibration” in Table 104 on page 249. For more information on automatic loading of pre-programmed calibration value, see section “Calibration Bytes” on page 229.

It is possible to reach higher accuracies than factory defaults, especially when the application allows temperature and voltage ranges to be narrowed. The firmware can reprogram the calibration data in OSCCAL0 either at start-up or during run-time. The continuous, run-time calibration method allows firmware to monitor voltage and temperature and compensate for any detected variations. See “OSCCAL0 – Oscillator Calibration Register” on page 32, “Temperature Measurement” on page 148, and Table 52 on page 150. The accuracy of this calibration is referred to as “User Calibration” in Table 104 on page 249.

The oscillator temperature calibration registers, OSCTCAL0A and OSCTCAL0B, can be used for one-time temperature calibration of oscillator frequency. See “OSCTCAL0A – Oscillator Temperature Calibration Register A” on page 33 and “OSCTCAL0B – Oscillator Temperature Calibration Register B” on page 33.

When this oscillator is used as the chip clock, it will still be used for the Watchdog Timer and for the Reset Time-out.

Start-up time for this clock source is determined by the SUT fuse bit, as shown in Table 7 on page 30.

6.2.3 Internal 32kHz Ultra Low Power (ULP) Oscillator

The internal 32kHz oscillator is a low power oscillator that operates with no external components. It provides a clock source with an approximate frequency of 32kHz. The frequency depends on supply voltage, temperature and batch variations. See Table 105 on page 250 for accuracy details.

During reset, hardware loads the pre-programmed calibration value into the OSCCAL1 register and thereby automatically calibrates the oscillator. The accuracy of this calibration is referred to as “Factory Calibration” in Table 105 on page 250. For more information on automatic loading of pre-programmed calibration value, see section “Calibration Bytes” on page 229.

Start-up time for this clock source is determined by the SUT fuse bit, as shown in Table 7 on page 30.

6.2.4 Default Clock Settings

The device is shipped with following fuse settings:

- Calibrated Internal 8MHz Oscillator (see CKSEL fuse bits in Table 6 on page 28)
- Longest possible start-up time (see SUT fuse bits in Table 7 on page 30)
- System clock prescaler set to 8 (see CKDIV8 fuse bit on page 32)

The default setting gives a 1MHz system clock and ensures all users can make their desired clock source setting using an in-system or high-voltage programmer.

The pin change interrupts trigger as follows:

- Pin Change Interrupt 0 (PCI0): triggers if any enabled PCINT[7:0] pin toggles
- Pin Change Interrupt 1 (PCI1): triggers if any enabled PCINT[15:8] pin toggles
- Pin Change Interrupt 2 (PCI2): triggers if any enabled PCINT[23:16] pin toggles
- Pin Change Interrupt 3(PCI3): triggers if any enabled PCINT[27:24] pin toggles

Registers PCMSK0, PCMSK1, PCMSK2, and PCMSK3 control which pins contribute to the pin change interrupts.

Pin change interrupts on PCINT[27:0] are detected asynchronously, which means that these interrupts can be used for waking the part also from sleep modes other than Idle mode.

In order for a pin change interrupt (PCINT) to be generated, the device must have an active I/O clock. As shown in Table 9 on page 34, the I/O clock domain is active in Idle Mode, but not in deeper sleep modes. In sleep modes deeper than Idle Mode, a toggled pin must remain in its toggled state until the device has fully woken up. See Table 7 on page 30 for wake up times. If the pin toggles back to its initial state during wake up the device will still complete the procedure but will not generate an interrupt once awake.

External interrupts INT0 and INT1 can be triggered by a falling or rising edge, or a low level. When INT0 or INT1 is enabled and configured as level triggered, the interrupt will trigger as long as the pin is held low.

Note that recognition of falling or rising edge interrupts on INT0 and INT1 requires the presence of an I/O clock, as described in “Clock System” on page 27.

9.2.1 Low Level Interrupt

A low level interrupt on INT0 or INT1 is detected asynchronously. This means that the interrupt source can be used for waking the part also from sleep modes other than Idle (the I/O clock is halted in all sleep modes except Idle).

Note that if a level triggered interrupt is used for wake-up from Power-down, the required level must be held long enough for the MCU to complete the wake-up to trigger the level interrupt. If the level disappears before the end of the Start-up Time, the MCU will still wake up, but no interrupt will be generated. The start-up time is defined as described in “Clock System” on page 27.

If the low level on the interrupt pin is removed before the device has woken up then program execution will not be diverted to the interrupt service routine but continue from the instruction following the SLEEP command.

9.2.2 Pin Change Interrupt Timing

A timing example of a pin change interrupt is shown in Figure 20.

10.2.6 Program Example

The following code example shows how to set port B pin 0 high, pin 1 low, and define the port pins from 2 to 3 as input with a pull-up assigned to port pin 2. The resulting pin values are read back again, but as previously discussed, a *nop* instruction is included to be able to read back the value recently assigned to some of the pins.

Assembly Code Example

```
; Define pull-ups and set outputs high
; Define directions for port pins
ldi                 r16,(1<<PUEB2)
ldi                 r17,(1<<PB0)
ldi                 r18,(1<<DDB1)|(1<<DDB0)
out                 PUEB,r16
out                 PORTB,r17
out                 DDRB,r18

; Insert nop for synchronization
nop

; Read port pins
in                  r16,PINB
```

Note: See “Code Examples” on page 7.

10.3 Alternative Port Functions

Most port pins have alternative functions in addition to being general digital I/Os. In Figure 25 below is shown how the port pin control signals from the simplified Figure 22 on page 60 can be overridden by alternative functions.

Pin	Signal	Composition
PB6	PUOE	0
	PUOV	0
	DDOE	0
	DDOV	0
	PVOE	0
	PVOV	0
	PTOE	0
	DIEOE	(PCINT14 • PCIE1) + ADC14D
	DIEOV	PCINT14 • PCIE1
	DI	PCINT14 Input
	AIO	ADC14 Input
PB7	PUOE	0
	PUOV	0
	DDOE	0
	DDOV	0
	PVOE	0
	PVOV	0
	PTOE	0
	DIEOE	(PCINT15 • PCIE1) + ADC15D
	DIEOV	PCINT15 • PCIE1
	DI	PCINT15 Input
	AIO	ADC15 Input

10.3.3 Alternative Functions of Port C

The alternative functions of port C are shown in Table 25.

Table 25. Alternative Functions of Port C

Pin	Function	Description of Alternative Function
PC0	PCINT16	Pin change interrupt source
	ADC16	Input channel for analog to digital converter (ADC)
	TOCC0	Timer/counter output compare, channel 0 (1)
	SS	SPI Slave Select input (2)
	XCK	USART transfer clock

For actual placement of I/O pins, refer to Figure 1 on page 2 (MLF), and Figure 2 on page 2 (TQFP). Also, see "TOCPMSA1 and TOCPMSA0 – Timer/Counter Output Compare Pin Mux Selection Registers" on page 127, and "TOCPMCOE – Timer/Counter Output Compare Pin Mux Channel Output Enable" on page 128.

11.2.1 Registers

The Timer/Counter (TCNT0) and Output Compare Registers (OCR0A and OCR0B) are 8-bit registers. Interrupt request (abbreviated to Int.Req. in Figure 26) signals are all visible in the Timer Interrupt Flag Register (TIFR). All interrupts are individually masked with the Timer Interrupt Mask Register (TIMSK). TIFR and TIMSK are not shown in the figure.

The Timer/Counter can be clocked internally, via the prescaler, or by an external clock source on the T0 pin. The Clock Select logic block controls which clock source and edge the Timer/Counter uses to increment (or decrement) its value. The Timer/Counter is inactive when no clock source is selected. The output from the Clock Select logic is referred to as the timer clock (clk_{T0}).

The double buffered Output Compare Registers (OCR0A and OCR0B) is compared with the Timer/Counter value at all times. The result of the compare can be used by the Waveform Generator to generate a PWM or variable frequency output on the Output Compare pins (OC0A and OC0B). See "Output Compare Unit" on page 89 for details. The Compare Match event will also set the Compare Flag (OCF0A or OCF0B) which can be used to generate an Output Compare interrupt request.

11.2.2 Definitions

Many register and bit references in this section are written in general form. A lower case "n" replaces the Timer/Counter number, in this case 0. A lower case "x" replaces the Output Compare Unit, in this case Compare Unit A or Compare Unit B. However, when using the register or bit defines in a program, the precise form must be used, i.e., TCNT0 for accessing Timer/Counter0 counter value and so on.

The definitions in Table 29 are also used extensively throughout the document.

Table 29. Definitions

Constant	Description
BOTTOM	The counter reaches BOTTOM when it becomes 0x00
MAX	The counter reaches its MAXimum when it becomes 0xFF (decimal 255)
TOP	The counter reaches the TOP when it becomes equal to the highest value in the count sequence. The TOP value can be assigned to be the fixed value 0xFF (MAX) or the value stored in the OCR0A Register. The assignment depends on the mode of operation

11.3 Clock Sources

The Timer/Counter can be clocked by an internal or an external clock source. The clock source is selected by the Clock Select logic which is controlled by the Clock Select (CS0[2:0]) bits located in the Timer/Counter Control Register (TCCR0B). For details on clock sources and prescaler, see "Timer/Counter Prescaler" on page 131.

11.4 Counter Unit

The main part of the 8-bit Timer/Counter is the programmable bi-directional counter unit. Figure 27 on page 89 shows a block diagram of the counter and its surroundings.

12.11.11TIFR1 – Timer/Counter Interrupt Flag Register

Bit	7	6	5	4	3	2	1	0	
0x16 (0x36)	–	–	ICF1	–	–	OCF1B	OCF1A	TOV1	TIFR1
Read/Write	R	R	R/W	R	R	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bits 7, 6, 4, 3 – Res: Reserved Bit**

These bits are reserved for future use. To ensure compatibility with future devices, these bits must be set to zero when the register is written.

- **Bit 5 – ICF1: Timer/Counter1, Input Capture Flag**

This flag is set when a capture event occurs on the ICP1 pin. When the Input Capture Register (ICR1) is set by the WGM1[3:0] to be used as the TOP value, the ICF1 flag is set when the counter reaches the TOP value.

ICF1 is automatically cleared when the Input Capture Interrupt Vector is executed. Alternatively, ICF1 can be cleared by writing a logic one to its bit location.

- **Bit 2 – OCF1B: Timer/Counter1, Output Compare B Match Flag**

This flag is set in the timer clock cycle after the counter (TCNT1) value matches the Output Compare Register B (OCR1B).

Note that a Forced Output Compare (1B) strobe will not set the OCF1B flag.

OCF1B is automatically cleared when the Output Compare Match B Interrupt Vector is executed. Alternatively, OCF1B can be cleared by writing a logic one to its bit location.

- **Bit 1 – OCF1A: Timer/Counter1, Output Compare A Match Flag**

This flag is set in the timer clock cycle after the counter (TCNT1) value matches the Output Compare Register A (OCR1A).

Note that a Forced Output Compare (1A) strobe will not set the OCF1A flag.

OCF1A is automatically cleared when the Output Compare Match A Interrupt Vector is executed. Alternatively, OCF1A can be cleared by writing a logic one to its bit location.

- **Bit 0 – TOV1: Timer/Counter1, Overflow Flag**

The setting of this flag is dependent of the WGM1[3:0] bits setting. In Normal and CTC modes, the TOV1 flag is set when the timer overflows. See Table 42 on page 125 for the TOV1 flag behavior when using another WGM1[3:0] bit setting.

TOV1 is automatically cleared when the Timer/Counter1 Overflow Interrupt Vector is executed. Alternatively, TOV1 can be cleared by writing a logic one to its bit location.

13. Timer/Counter Prescaler

Timer/Counter0 and Timer/Counter1 share the same prescaler module, but the Timer/Counters can have different prescaler settings. The description below applies to both Timer/Counters. Tn is used as a general name, n = 0, 1.

The Timer/Counter can be clocked directly by the system clock (by setting the CSn[2:0] = 1). This provides the fastest operation, with a maximum Timer/Counter clock frequency equal to system clock frequency ($f_{CLK\_I/O}$). Alternatively, one of four taps from the prescaler can be used as a clock source. The prescaled clock has a frequency of either $f_{CLK\_I/O}/8$, $f_{CLK\_I/O}/64$, $f_{CLK\_I/O}/256$, or $f_{CLK\_I/O}/1024$.

13.1 Prescaler Reset

The prescaler is free running, i.e., operates independently of the Clock Select logic of the Timer/CounterCounter, and it is shared by the Timer/Counter Tn. Since the prescaler is not affected by the Timer/Counter's clock select, the state of the prescaler will have implications for situations where a prescaled clock is used. One example of prescaling artifacts occurs when the timer is enabled and clocked by the prescaler (CSn[2:0] = 2, 3, 4, or 5). The number of system clock cycles from when the timer is enabled to the first count occurs can be from 1 to N+1 system clock cycles, where N equals the prescaler divisor (8, 64, 256, or 1024).

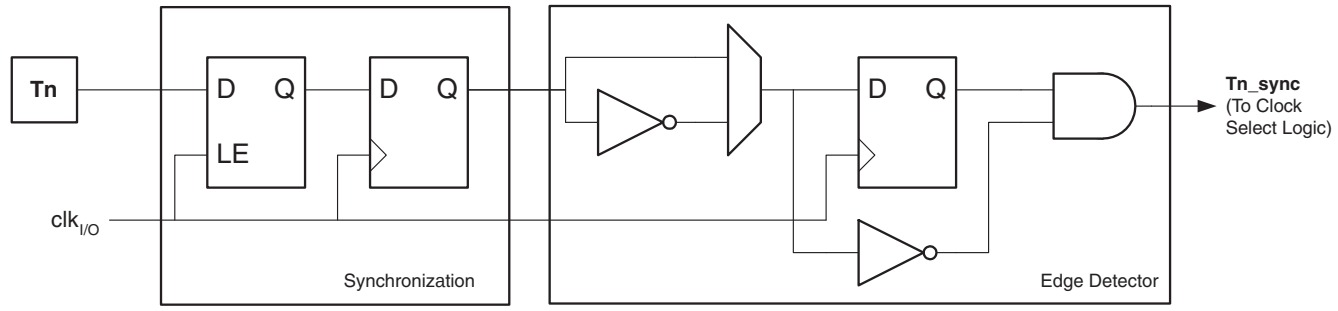
It is possible to use the Prescaler Reset for synchronizing the Timer/Counter to program execution.

13.2 External Clock Source

An external clock source applied to the Tn pin can be used as Timer/Counter clock (clk_{Tn}). The Tn pin is sampled once every system clock cycle by the pin synchronization logic. The synchronized (sampled) signal is then passed through the edge detector. Figure 50 shows a functional equivalent block diagram of the Tn synchronization and edge detector logic. The registers are clocked at the positive edge of the internal system clock ($clk_{I/O}$). The latch is transparent in the high period of the internal system clock.

The edge detector generates one clk_{T0} pulse for each positive (CSn[2:0] = 7) or negative (CSn[2:0] = 6) edge it detects.

Figure 50. T0 Pin Sampling



The synchronization and edge detector logic introduces a delay of 2.5 to 3.5 system clock cycles from an edge has been applied to the Tn pin to the counter is updated.

Enabling and disabling of the clock input must be done when Tn has been stable for at least one system clock cycle, otherwise it is a risk that a false Timer/Counter clock pulse is generated.

Each half period of the external clock applied must be longer than one system clock cycle to ensure correct sampling. The external clock must be guaranteed to have less than half the system clock frequency ($f_{ExtClk} < f_{clk\_I/O}/2$) given a 50/50% duty cycle. Since the edge detector uses sampling, the maximum frequency of an external clock it can detect is half the sampling frequency (Nyquist sampling theorem). However, due to variation of the system clock frequency and duty cycle caused by oscillator source tolerances, it is recommended that maximum frequency of an external clock source is less than $f_{clk\_I/O}/2.5$.

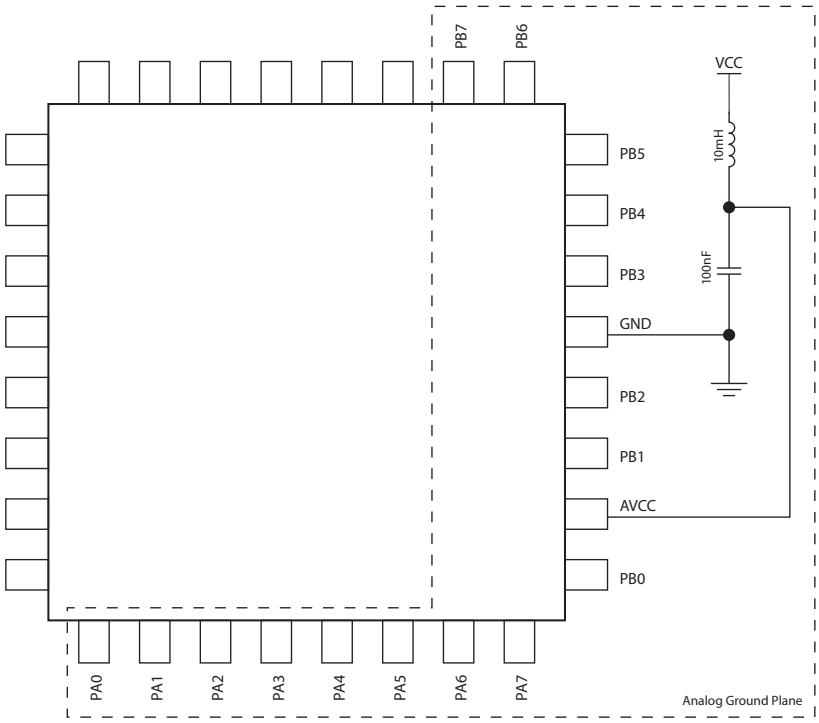
An external clock source can not be prescaled.

15.9 Noise Canceling Techniques

Digital circuitry inside and outside the device generates EMI which might affect the accuracy of analog measurements. When conversion accuracy is critical, the noise level can be reduced by applying the following techniques:

- Keep analog signal paths as short as possible.
- Make sure analog tracks run over the analog ground plane.
- Keep analog tracks well away from high-speed switching digital tracks.
- If any port pin is used as a digital output, it mustn't switch while a conversion is in progress.
- Place bypass capacitors as close to V_{CC} and GND pins as possible.
- The analog supply voltage pin (AV_{CC}) should be connected to the digital supply voltage pin (V_{CC}) via an LC network as shown in Figure 61.

Figure 61. ADC Power Connections



Where high ADC accuracy is required it is recommended to use ADC Noise Reduction Mode, as described in Section 15.7 on page 144. This is especially the case when system clock frequency is above 1 MHz, or when the ADC is used for reading the internal temperature sensor, as described in Section 15.12 on page 148. A good system design with properly placed, external bypass capacitors does reduce the need for using ADC Noise Reduction Mode

15.10 ADC Accuracy Definitions

An n-bit single-ended ADC converts a voltage linearly between GND and V_{REF} in 2^n steps (LSBs). The lowest code is read as 0, and the highest code is read as 2^n-1.

Several parameters describe the deviation from the ideal behavior, as follows:

- Normal asynchronous mode
- Double speed asynchronous mode
- Master synchronous mode
- Slave synchronous mode

The UMSEL bit (see “UCSRC – USART Control and Status Register C” on page 186) selects between asynchronous and synchronous operation. In asynchronous mode, the speed is controlled by the U2X bit (see “UCSRA – USART Control and Status Register A” on page 184).

In synchronous mode (UMSEL = 1), the direction bit of the XCK pin (DDR_XCK) in the Data Direction Register where the XCK pin is located (DDRx) controls whether the clock source is internal (master mode), or external (slave mode). The XCK pin is active in synchronous mode, only.

17.3.1 Internal Clock Generation – The Baud Rate Generator

Internal clock generation is used in asynchronous and synchronous master modes of operation. The description in this section refers to Figure 71 on page 165.

The USART Baud Rate Register (UBRR) and the down-counter connected to it function as a programmable prescaler, or baud rate generator. The down-counter, running at system clock (f_{osc}) is loaded with the UBRR value each time the counter has counted down to zero, or when UBRR0L is written.

A clock is generated each time the counter reaches zero. This is the baud rate generator clock output and has a frequency of f_{osc}/(UBRR+1). Depending on the mode of operation the transmitter divides the baud rate generator clock output by 2, 8 or 16. The baud rate generator output is used directly by the receiver’s clock and data recovery units. However, the recovery units use a state machine that uses 2, 8 or 16 states, depending on mode set by UMSEL, U2X and DDR_XCK bits.

Table 60 contains equations for calculating the baud rate (in bits per second) and for calculating the UBRR value for each mode of operation using an internally generated clock source.

Table 60. Equations for Calculating Baud Rate Register Setting

Operating Mode	Baud Rate[1]	UBRR Value
Asynchronous Normal mode (U2Xn = 0)	$BAUD = \frac{f_{OSC}}{16 \times (\boldsymbol{UBRR} + 1)}$	$\boldsymbol{UBRR} = \frac{f_{OSC}}{16 \times BAUD} - 1$
Asynchronous Double Speed mode (U2Xn = 1)	$BAUD = \frac{f_{OSC}}{8 \times (\boldsymbol{UBRR} + 1)}$	$\boldsymbol{UBRR} = \frac{f_{OSC}}{8 \times BAUD} - 1$
Synchronous Master mode	$BAUD = \frac{f_{OSC}}{2 \times (\boldsymbol{UBRR} + 1)}$	$\boldsymbol{UBRR} = \frac{f_{OSC}}{2 \times BAUD} - 1$

Note: 1. Baud rate is defined as the transfer rate in bits per second (bps)

Signal description for Table 60:

BAUD	Baud rate (in bits per second, bps)
f_{osc}	System Oscillator clock frequency
UBRR	Contents of the UBRRH and UBRRL Registers, (0-4095)

Some examples of UBRR values for selected system clock frequencies are shown in Table 63 on page 181.

- **Bit 4 – RXEN: Receiver Enable**

Writing this bit to one enables the USART Receiver. When enabled, the receiver will override normal port operation for the RxD pin. Disabling the receiver will flush the receive buffer, invalidating FE, DOR, and UPE Flags.

- **Bit 3 – TXEN: Transmitter Enable**

Writing this bit to one enables the USART Transmitter. When enabled, the transmitter will override normal port operation for the TxD pin. Disabling the transmitter (writing TXENn to zero) will not become effective until ongoing and pending transmissions are completed, i.e., when the transmit shift register and transmit buffer register do not contain data to be transmitted. When disabled, the transmitter will no longer override the TxD port.

- **Bit 2 – UCSZ2: Character Size**

The UCSZ2 bit combined with the UCSZ[1:0] bits sets the number of data bits (Character SiZe) in a frame the receiver and transmitter use.

- **Bit 1 – RXB8: Receive Data Bit 8**

RXB8 is the ninth data bit of the received character when operating with serial frames with nine data bits. It must be read before reading the low bits from UDR.

- **Bit 0 – TXB8: Transmit Data Bit 8**

TXB8 is the ninth data bit in the character to be transmitted when operating with serial frames with nine data bits. It must be written before writing the low bits to UDR.

17.11.4 UCSRC – USART Control and Status Register C

Bit	7	6	5	4	3	2	1	0	
(0xC2)	**UMSEL1**	**UMSEL0**	**UPM1**	**UPM0**	**USBS**	**UCSZ1**	**UCSZ0**	**UCPOL**	**UCSRC**
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	1	1	0	

- **Bits 7:6 – UMSEL[1:0]: USART Mode Select**

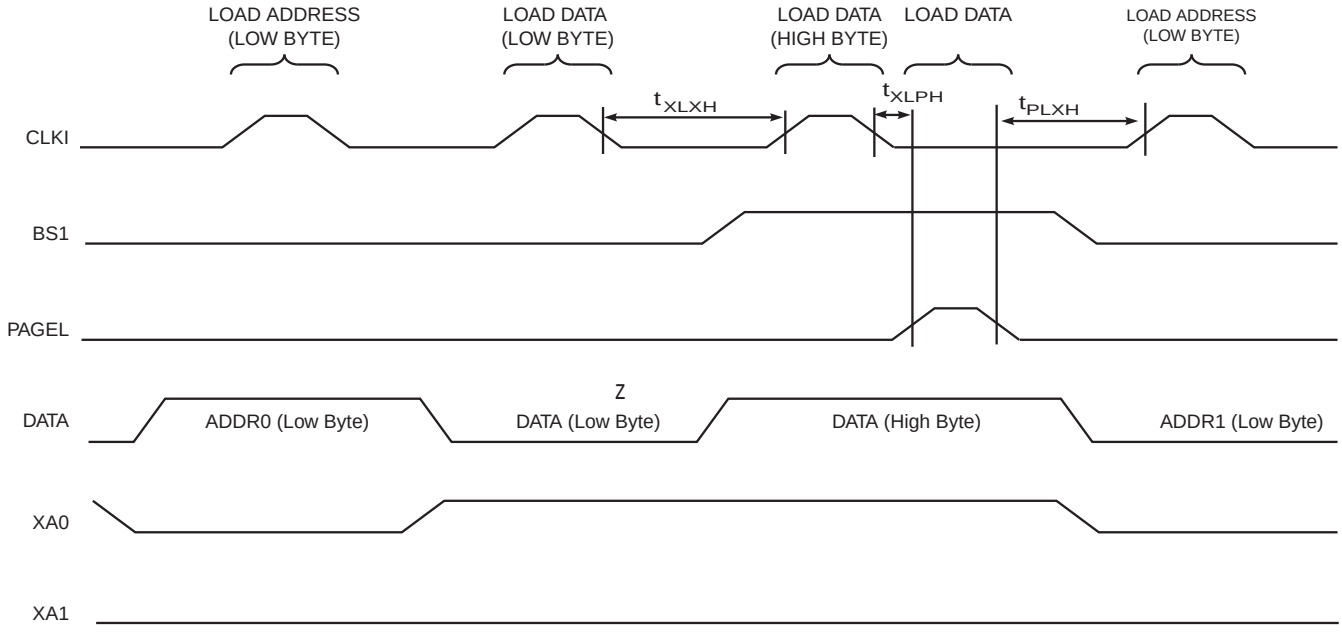
These bits select the mode of operation of the USART, as shown in Table 67.

Table 67. USART Mode of Operation

UMSEL1	UMSEL0	Mode
0	0	Asynchronous USART
0	1	Synchronous USART
1	0	Reserved
1	1	Master SPI (MSPIM)[1]

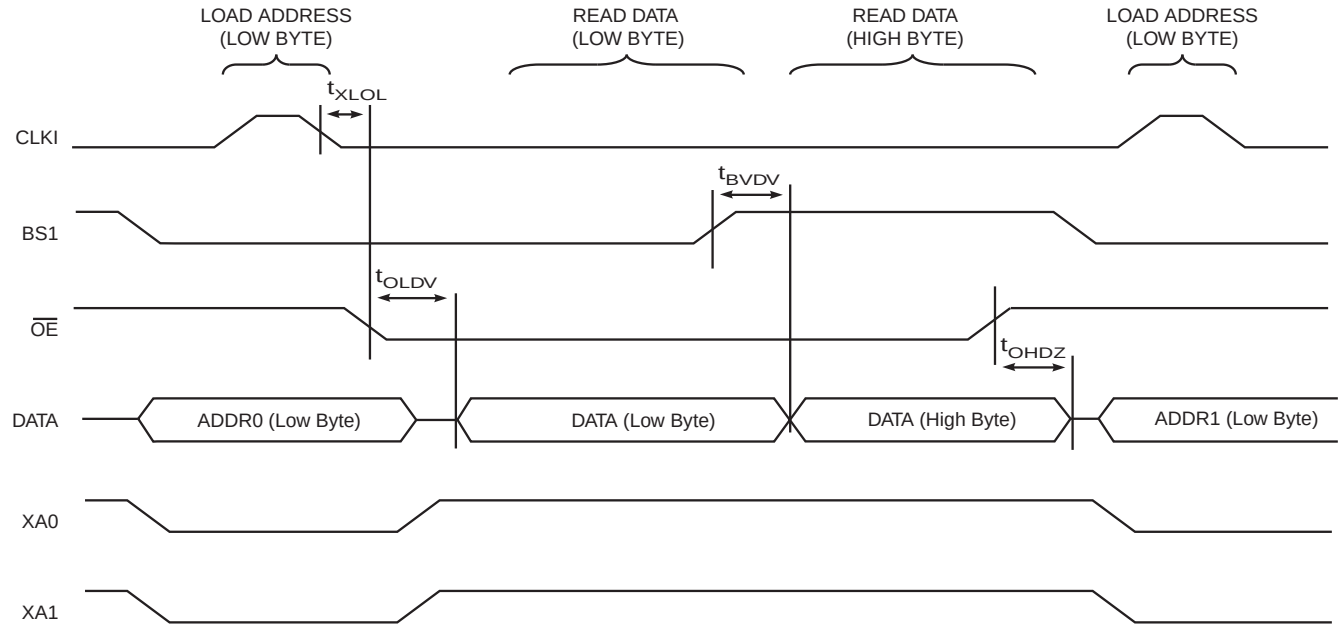
Note: 1. For full description of the Master SPI Mode (MSPIM) Operation, see “USART in SPI Mode” on page 190.

Figure 105. Parallel Programming Timing, Loading Sequence with Timing Requirements[1]



Note: 1. The timing requirements shown in Figure 104 (i.e., t_{DVXH}, t_{XHXL}, and t_{XLDX}) also apply to loading operation.

Figure 106. Parallel Programming Timing, Reading Sequence (within the Same Page) with Timing Requirements[1]



Note: 1. The timing requirements shown in Figure 104 (i.e., t_{DVXH}, t_{XHXL}, and t_{XLDX}) also apply to reading operation.

Figure 154. V_{OL}: Output Voltage vs. Sink Current (Standard I/O Pin, V_{CC} = 3V)

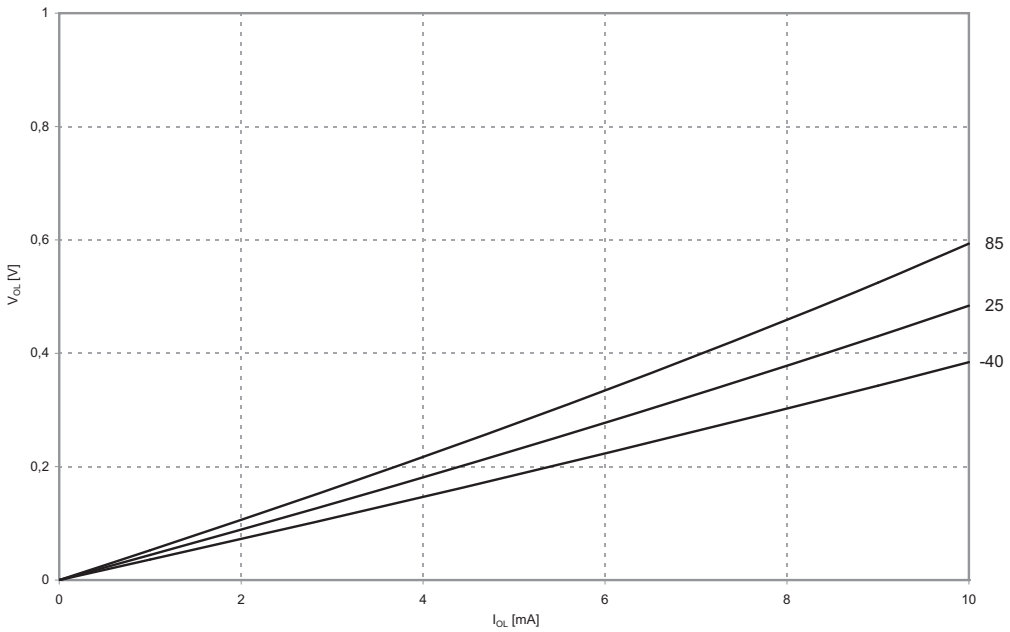


Figure 155. V_{OL}: Output Voltage vs. Sink Current (Standard I/O Pin, V_{CC} = 5V)

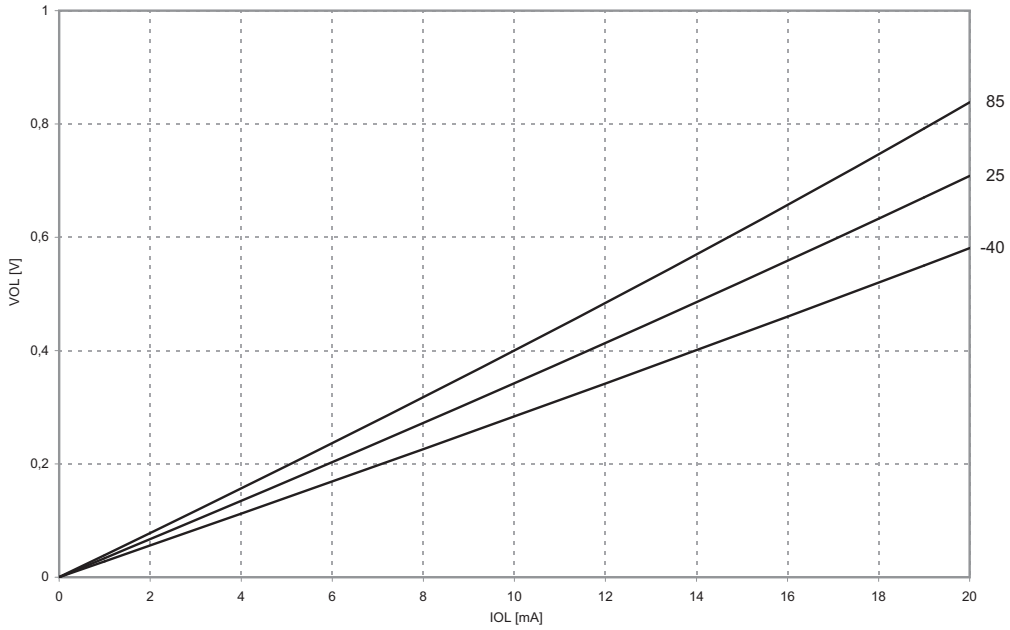


Figure 160. V_{OL}: Output Voltage vs. Sink Current (Extra High Sink I/O Pin, V_{CC} = 3V)

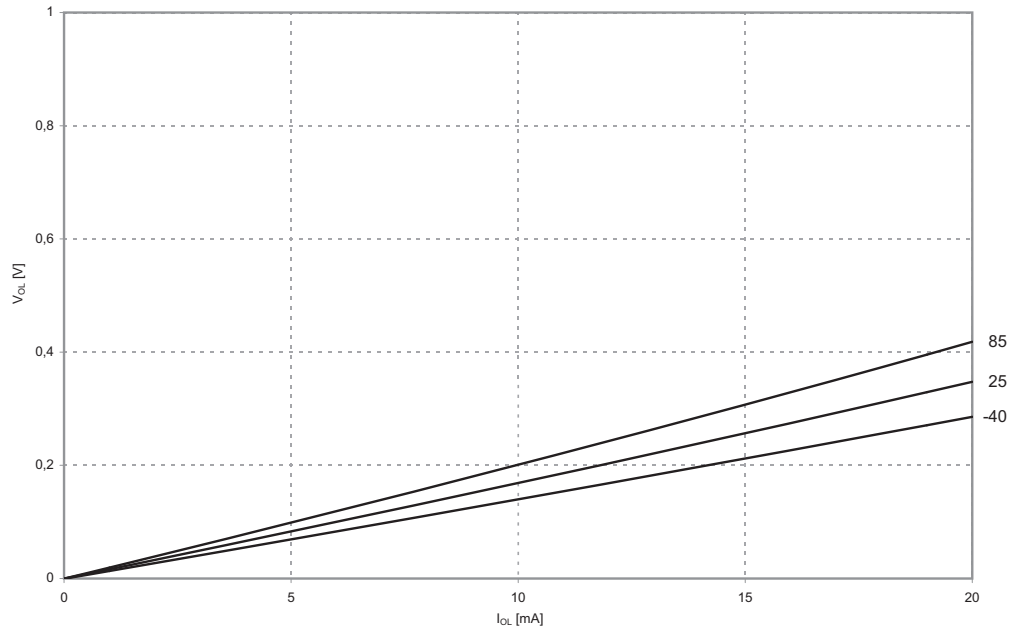


Figure 161. V_{OL}: Output Voltage vs. Sink Current (Extra High Sink I/O Pin, V_{CC} = 5V)

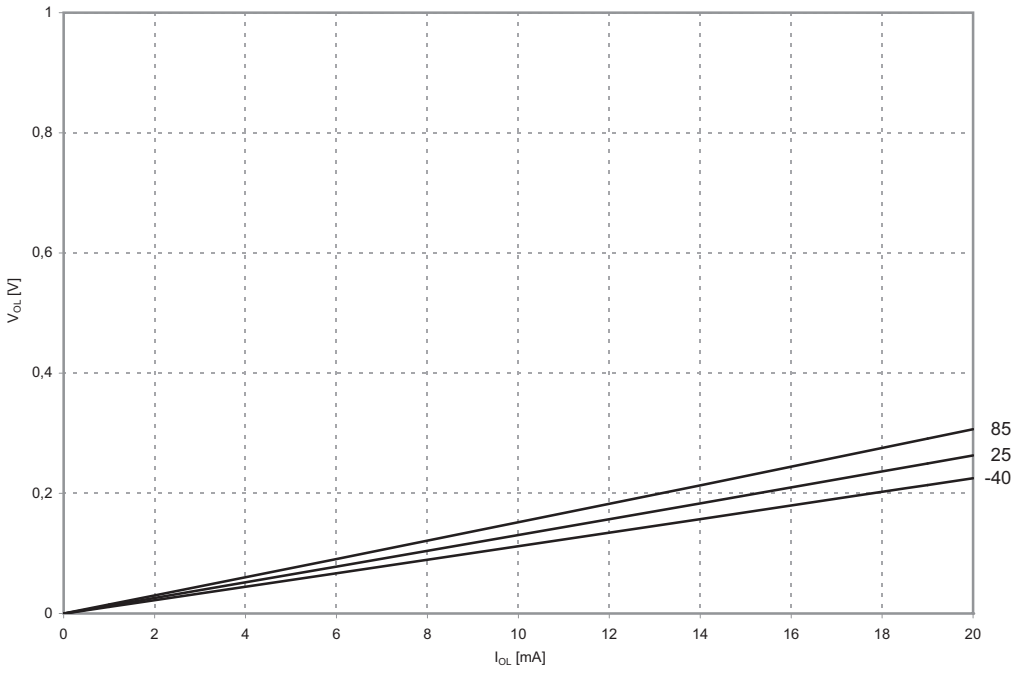
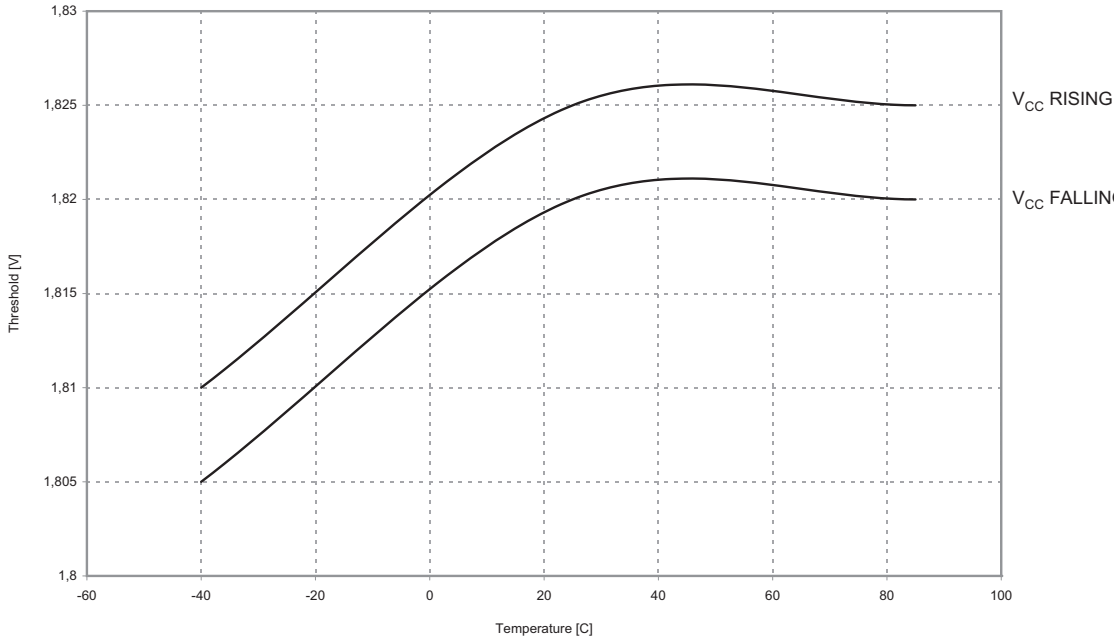
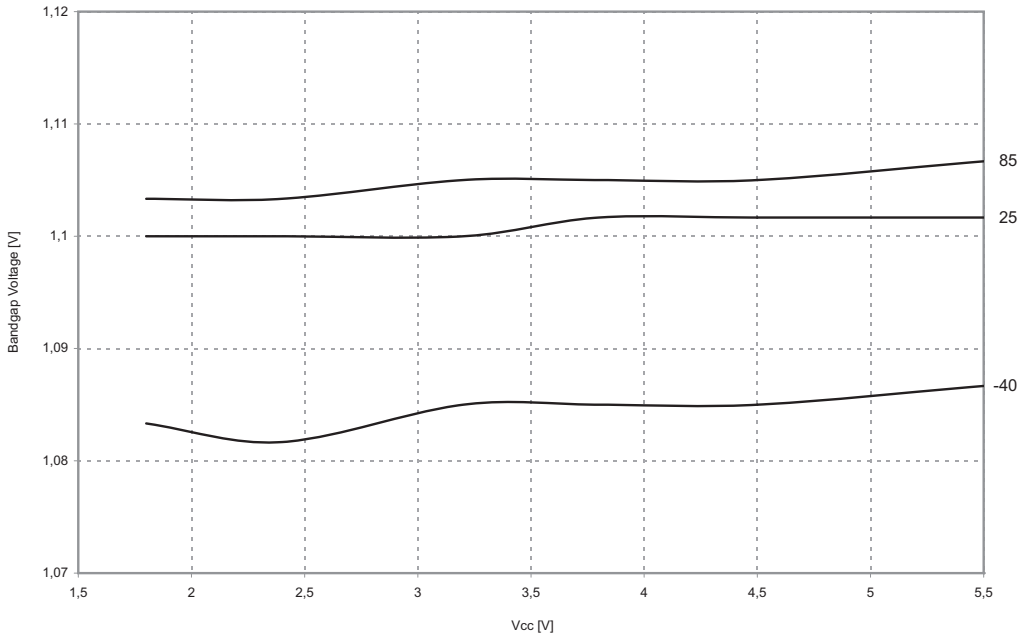


Figure 170. Sampled BOD Threshold vs Temperature (BODLEVEL = 1.8V)



25.11 Bandgap Voltage

Figure 171. Bandgap Voltage vs. Supply Voltage



31. Revision History

Doc. Rev.	Date	Comments
8371A	08/2012	Initial document release.