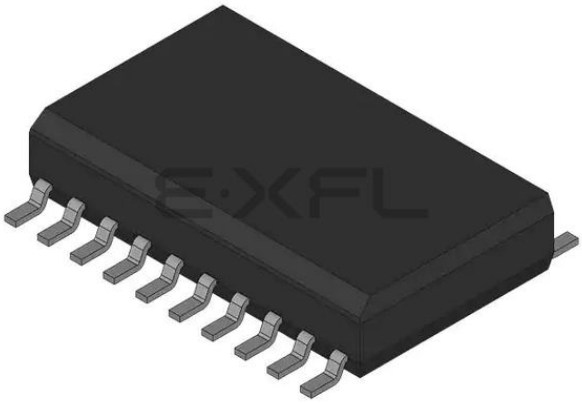


Welcome to E-XFL.COM

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	Obsolete
Core Processor	AVR
Core Size	8-Bit
Speed	16MHz
Connectivity	I²C, LINbus, SPI, UART/USART, USI
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	16
Program Memory Size	16KB (8K x 16)
Program Memory Type	FLASH
EEPROM Size	512 x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 11x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	20-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/atmel/attiny167-15sz

1.3 Automotive Quality Grade

The Atmel® ATtiny87/167 have been developed and manufactured according to the most stringent requirements of the international standard ISO-TS-16949. This data sheet contains limit values extracted from the results of extensive characterization (temperature and voltage). The quality and reliability of the Atmel ATtiny87/167 have been verified during regular product qualification as per AEC-Q100 grade 1.

As indicated in the ordering information paragraph, this document refers only to grade 1 products, for grade 0 products refer to appendix A.

Table 1-2. Temperature Grade Identification for Automotive Products

Temperature	Temperature Identifier	Comments
–40°C/+125°C	Z	Grade 1
–40°C/+150°C	D	Grade 0

1.4 Disclaimer

Typical values contained in this data sheet are based on simulations and characterization of other AVR® microcontrollers manufactured on the same process technology. Min. and Max values will be available after the device is characterized.

- **Bit 1 – EEPE: EEPROM Program Enable**

The EEPROM program enable signal EEPE is the programming enable signal to the EEPROM. When EEPE is written, the EEPROM will be programmed according to the EEPMn bits setting. The EEMPE bit must be written to one before a logical one is written to EEPE, otherwise no EEPROM write takes place. When the write access time has elapsed, the EEPE bit is cleared by hardware. When EEPE has been set, the CPU is halted for two cycles before the next instruction is executed.

- **Bit 0 – EERE: EEPROM Read Enable**

The EEPROM read enable signal – EERE – is the read strobe to the EEPROM. When the correct address is set up in the EEAR register, the EERE bit must be written to one to trigger the EEPROM read. The EEPROM read access takes one instruction, and the requested data is available immediately. When the EEPROM is read, the CPU is halted for four cycles before the next instruction is executed. The user should poll the EEPE bit before starting the read operation. If a write operation is in progress, it is neither possible to read the EEPROM, nor to change the EEAR register.

3.5.4 General Purpose I/O Register 2 – GPIOR2

Bit	7	6	5	4	3	2	1	0	
	GPIOR27	GPIOR26	GPIOR25	GPIOR24	GPIOR23	GPIOR22	GPIOR21	GPIOR20	GPIOR2
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

3.5.5 General Purpose I/O Register 1 – GPIOR1

Bit	7	6	5	4	3	2	1	0	
	GPIOR17	GPIOR16	GPIOR15	GPIOR14	GPIOR13	GPIOR12	GPIOR11	GPIOR10	GPIOR1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

3.5.6 General Purpose I/O Register 0 – GPIOR0

Bit	7	6	5	4	3	2	1	0	
	GPIOR07	GPIOR06	GPIOR05	GPIOR04	GPIOR03	GPIOR02	GPIOR01	GPIOR00	GPIOR0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

4.3.5 Clock Availability

'Request for clock availability' command enables a hardware oscillation cycle counter driven by the selected source clock, CSEL3..0. The count limit value is determined by the settings of CSUT1..0. The clock is declared ready (CLKRDY = 1) when the count limit value is reached. The CLKRDY flag is reset when the count starts. Once set, this flag remains unchanged until a new count is commanded. To perform this checking, the CKSEL and CSUT fields should not be changed while the operation is running.

Note that once the new clock source is selected ('*enable clock source*' command), the count procedure is automatically started. The user (code) should wait for the setting of the CLKRDY flag in CLKSCR register before using a newly selected clock.

At any time, the user (code) can ask for the availability of a clock source. The user (code) can request it by writing the *'request for clock availability '* command in the CLKSCR register. A full polling of the status of clock sources can thus be done.

4.3.6 System Clock Source Recovering

The '*recover system clock source*' command returns the current clock source used to drive the system clock as per Table 4-1 on page 26. The CKSEL field of CLKSELR register is then updated with this returned value. There is no information on the SUT used or status on CKOUT.

4.3.7 Clock Switching

To drive the system clock, the user can switch from the current clock source to any other of the following ones (one of them being the current clock source):

1. Calibrated internal RC oscillator 8.0MHz,
2. Internal watchdog oscillator 128kHz,
3. External clock,
4. External low-frequency oscillator,
5. External Crystal/Ceramic Resonator.

The clock switching is performed by a sequence of commands. First, the user (code) must make sure that the new clock source is operating. Then the '*clock source switching*' command can be issued. Once this command has been successfully completed using the '*recover system clock source*' command, the user (code) may stop the previous clock source.

It is strongly recommended to run this sequence only once the interrupts have been disabled. The user (code) is responsible for the correct implementation of the clock switching sequence.

- **Bit 7 – CLKPCE: Clock Prescaler Change Enable**

The CLKPCE bit must be written to logic one to enable change of the CLKPS bits. The CLKPCE bit is only updated when the other bits in CLKPR are simultaneously written to zero. CLKPCE is cleared by hardware four cycles after it is written or when the CLKPS bits are written. Rewriting the CLKPCE bit within this time-out period does neither extend the time-out period, nor clear the CLKPCE bit.

- **Bits 6:4 – Res: Reserved Bits**

These bits are reserved bits in the Atmel® ATtiny87/167 and will always read as zero.

- **Bits 3:0 – CLKPS3:0: Clock Prescaler Select Bits 3 - 0**

These bits define the division factor between the selected clock source and the internal system clock. These bits can be written run-time to vary the clock frequency to suit the application requirements. As the divider divides the master clock input to the MCU, the speed of all synchronous peripherals is reduced when a division factor is used. The division factors are given in Table 4-10.

To avoid unintentional changes of clock frequency, a special write procedure must be followed to change the CLKPS bits:

1. Write the Clock Prescaler Change Enable (CLKPCE) bit to one and all other bits in CLKPR to zero.
2. Within four cycles, write the desired value to CLKPS while writing a zero to CLKPCE.

Interrupts must be disabled when changing prescaler setting in order not to disturb the procedure.

The CKDIV8 fuse determines the initial value of the CLKPS bits. If CKDIV8 is unprogrammed, the CLKPS bits will be reset to “0000”. If CKDIV8 is programmed, CLKPS bits are reset to “0011”, giving a division factor of eight at start up. This feature should be used if the selected clock source has a higher frequency than the maximum frequency of the device at the present operating conditions. Note that any value can be written to the CLKPS bits regardless of the CKDIV8 Fuse setting. The application software must ensure that a sufficient division factor is chosen if the selected clock source has a higher frequency than the maximum frequency of the device at the present operating conditions. The device is shipped with the CKDIV8 fuse programmed.

Table 4-10. Clock Prescaler Select

CLKPS3	CLKPS2	CLKPS1	CLKPS0	Clock Division Factor
0	0	0	0	1
0	0	0	1	2
0	0	1	0	4
0	0	1	1	8
0	1	0	0	16
0	1	0	1	32
0	1	1	0	64
0	1	1	1	128
1	0	0	0	256
1	0	0	1	Reserved
1	0	1	0	Reserved
1	0	1	1	Reserved
1	1	0	0	Reserved
1	1	0	1	Reserved
1	1	1	0	Reserved
1	1	1	1	Reserved

4.5.4 CLKSELR - Clock Selection Register

Bit	7	6	5	4	3	2	1	0	
	-	COUT	CSUT1	CSUT0	CSEL3	CSEL2	CSEL1	CSEL0	CLKSELR
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	~ (CKOUT) fuse	SUT1..0 fuses			CKSEL3..0 fuses			

- **Bit 7– Res: Reserved Bit**

This bit is reserved bit in the Atmel® ATtiny87/167 and will always read as zero.

- **Bit 6 – COUT: Clock Out**

The COUT bit is initialized with ~(CKOUT) fuse bit.
The COUT bit is only used in case of '*CKOUT*' command. Refer to Section 4.2.7 "Clock Output Buffer" on page 32 for using.
In case of '*recover system clock Source*' command, COUT it is not affected (no recovering of this setting).

- **Bits 5:4 – CSUT1:0: Clock Start-up Time**

CSUT bits are initialized with the values of SUT fuse bits.
In case of '*enable/disable clock source*' command, CSUT field provides the code of the clock start-up time. Refer to subdivisions of Section 4.2 "Clock Sources" on page 26 for code of clock start-up times.
In case of '*recover system clock source*' command, CSUT field is not affected (no recovering of SUT code).

- **Bits 3:0 – CSEL3:0: Clock Source Select**

CSEL bits are initialized with the values of CKSEL fuse bits.
In case of '*enable/disable clock source*', '*request for clock availability*' or '*clock source switch*' command, CSEL field provides the code of the clock source. Refer to Table 4-1 on page 26 and subdivisions of Section 4.2 "Clock Sources" on page 26 for clock source codes.
In case of '*recover system clock source*' command, CSEL field contains the code of the clock source used to drive the clock control unit as described in Figure 4-1 on page 25.

In fast PWM mode, the compare unit allows generation of PWM waveforms on the OC0A pin. Setting the COM0A1:0 bits to two will produce a non-inverted PWM and an inverted PWM output can be generated by setting the COM0A1:0 to three (See Table 10-2 on page 96). The actual OC0A value will only be visible on the port pin if the data direction for the port pin is set as output. The PWM waveform is generated by setting (or clearing) the OC0A register at the compare match between OCR0A and TCNT0, and clearing (or setting) the OC0A register at the timer clock cycle the counter is cleared (changes from MAX to BOTTOM).

The PWM frequency for the output can be calculated by the following equation:

$$f_{OCnxPWM} = \frac{f_{\text{clk\_I/O}}}{N \cdot 256}$$

The N variable represents the prescale factor (1, 8, 32, 64, 128, 256, or 1024).

The extreme values for the OCR0A register represent special cases when generating a PWM waveform output in the fast PWM mode. If the OCR0A is set equal to BOTTOM, the output will be a narrow spike for each MAX+1 timer clock cycle. Setting the OCR0A equal to MAX will result in a constantly high or low output (depending on the polarity of the output set by the COM0A1:0 bits.)

A frequency (with 50% duty cycle) waveform output in fast PWM mode can be achieved by setting OC0A to toggle its logical level on each compare match (COM0A1:0 = 1). The waveform generated will have a maximum frequency of $f_{oc}0_A = f_{clk\_I/O}/2$ when OCR0A is set to zero. This feature is similar to the OC0A toggle in CTC mode, except the double buffer feature of the output compare unit is enabled in the fast PWM mode.

10.7.4 Phase Correct PWM Mode

The phase correct PWM mode (WGM01:0 = 1) provides a high resolution phase correct PWM waveform generation option. The phase correct PWM mode is based on a dual-slope operation. The counter counts repeatedly from BOTTOM to MAX and then from MAX to BOTTOM. In non-inverting compare output mode, the output compare (OC0A) is cleared on the compare match between TCNT0 and OCR0A while up counting, and set on the compare match while down counting. In inverting output compare mode, the operation is inverted. The dual-slope operation has lower maximum operation frequency than single slope operation. However, due to the symmetric feature of the dual-slope PWM modes, these modes are preferred for motor control applications.

The PWM resolution for the phase correct PWM mode is fixed to eight bits. In phase correct PWM mode the counter is incremented until the counter value matches MAX. When the counter reaches MAX, it changes the count direction. The TCNT0 value will be equal to MAX for one timer clock cycle. The timing diagram for the phase correct PWM mode is shown on Figure 10-7 on page 91. The TCNT0 value is in the timing diagram shown as a histogram for illustrating the dual-slope operation. The diagram includes non-inverted and inverted PWM outputs. The small horizontal line marks on the TCNT0 slopes represent compare matches between OCR0A and TCNT0.

10.8 Timer/Counter Timing Diagrams

The following figures show the timer/counter in synchronous mode, and the timer clock (clk_T0) is therefore shown as a clock enable signal. In asynchronous mode, $clk_{I/O}$ should be replaced by the timer/counter oscillator clock. The figures include information on when interrupt flags are set. Figure 10-8 contains timing data for basic timer/counter operation. The figure shows the count sequence close to the MAX value in all modes other than phase correct PWM mode.

Figure 10-8. Timer/Counter Timing Diagram, no Prescaling

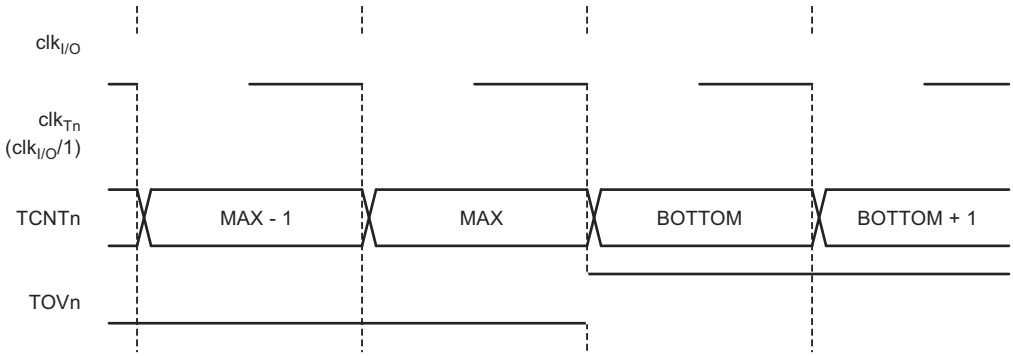


Figure 10-9 shows the same timing data, but with the prescaler enabled.

Figure 10-9. Timer/Counter Timing Diagram, with Prescaler ($f_{clk\_I/O}$/8)

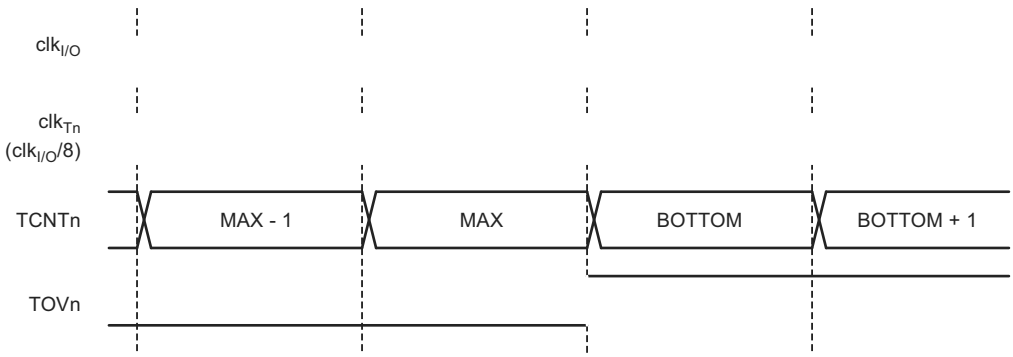
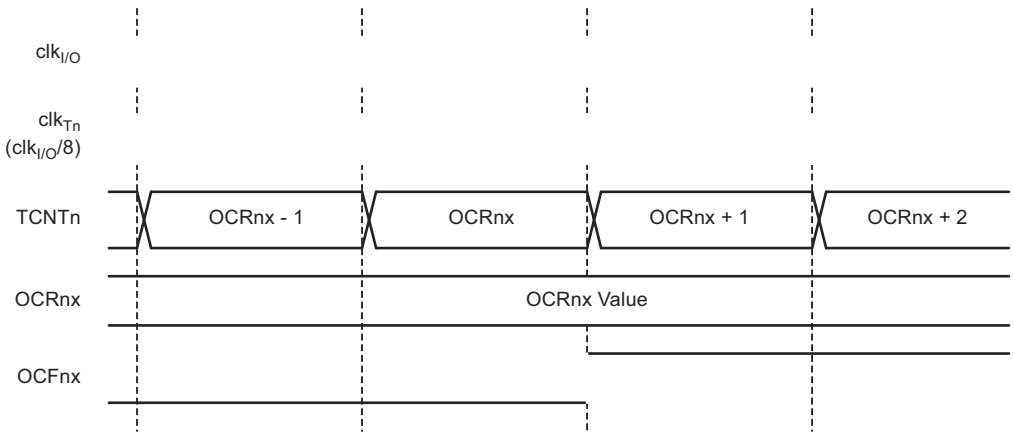


Figure 10-10 shows the setting of OCF0A in all modes except CTC mode.

Figure 10-10.Timer/Counter Timing Diagram, Setting of OCF0A, with Prescaler ($f_{clk\_I/O}$/8)



- **Bit 3 – OCR0AUB: Output Compare 0 Register A Update Busy**

When timer/counter0 operates asynchronously and OCR0A is written, this bit becomes set. When OCR0A has been updated from the temporary storage register, this bit is cleared by hardware. A logical zero in this bit indicates that OCR0A is ready to be updated with a new value.

- **Bit 2 – Res: Reserved Bit**

This bit is reserved in the Atmel® ATtiny87/167 and will always read as zero.

- **Bit 1 – TCR0AUB: Timer/Counter0 Control Register A Update Busy**

When timer/counter0 operates asynchronously and TCCR0A is written, this bit becomes set. When TCCR0A has been updated from the temporary storage register, this bit is cleared by hardware. A logical zero in this bit indicates that TCCR0A is ready to be updated with a new value.

- **Bit 0 – TCR0BUB: Timer/Counter0 Control Register B Update Busy**

When timer/counter0 operates asynchronously and TCCR0B is written, this bit becomes set. When TCCR0B has been updated from the temporary storage register, this bit is cleared by hardware. A logical zero in this bit indicates that TCCR0B is ready to be updated with a new value.

If a write is performed to any of the four timer/counter0 registers while its update busy flag is set, the updated value might get corrupted and cause an unintentional interrupt to occur.

The mechanisms for reading TCNT0, OCR0A, TCCR0A and TCCR0B are different. When reading TCNT0, the actual timer value is read. When reading OCR0A, TCCR0A or TCCR0B the value in the temporary storage register is read.

10.11.5 Timer/Counter0 Interrupt Mask Register – TIMSK0

Bit	7	6	5	4	3	2	1	0	
	–	–	–	–	–	–	OCIE0A	TOIE0	TIMSK0
Read/Write	R	R	R	R	R	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7:2 – Res: Reserved Bits**

These bits are reserved in the Atmel ATtiny87/167 and will always read as zero.

- **Bit 1 – OCIE0A: Timer/Counter0 Output Compare Match A Interrupt Enable**

When the OCIE0A bit is written to one and the I-bit in the status register is set (one), the timer/counter0 compare match A interrupt is enabled. The corresponding interrupt is executed if a compare match in timer/counter0 occurs, i.e., when the OCF0A bit is set in the timer/counter0 interrupt flag register – TIFR0.

- **Bit 0 – TOIE0: Timer/Counter0 Overflow Interrupt Enable**

When the TOIE0 bit is written to one and the I-bit in the status register is set (one), the timer/counter0 overflow interrupt is enabled. The corresponding interrupt is executed if an overflow in timer/counter0 occurs, i.e., when the TOV0 bit is set in the timer/counter0 interrupt flag register – TIFR0.

10.11.6 Timer/Counter0 Interrupt Flag Register – TIFR0

Bit	7	6	5	4	3	2	1	0	
	–	–	–	–	–	–	OCF0A	TOV0	TIFR0
Read/Write	R	R	R	R	R	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7:2 – Res: Reserved Bits**

These bits are reserved in the Atmel ATtiny87/167 and will always read as zero.

12.6.1 Input Capture Trigger Source

The main trigger source for the input capture unit is the input capture pin (ICP1). Only timer/counter1 can alternatively use the analog comparator output as trigger source for the input capture unit. The analog comparator is selected as trigger source by setting the Analog Comparator Input Capture (ACIC) bit in the analog comparator control and status register (ACSR). Be aware that changing trigger source can trigger a capture. The input capture flag must therefore be cleared after the change.

Both the input capture pin (ICP1) and the analog comparator output (ACO) inputs are sampled using the same technique as for the T1 pin (Figure 11-1 on page 101). The edge detector is also identical. However, when the noise canceler is enabled, additional logic is inserted before the edge detector, which increases the delay by four system clock cycles. Note that the input of the noise canceler and edge detector is always enabled unless the timer/counter is set in a waveform generation mode that uses ICR1 to define TOP.

An input capture can be triggered by software by controlling the port of the ICP1 pin.

12.6.2 Noise Canceler

The noise canceler improves noise immunity by using a simple digital filtering scheme. The noise canceler input is monitored over four samples, and all four must be equal for changing the output that in turn is used by the edge detector.

The noise canceler is enabled by setting the input capture noise canceler (ICNC1) bit in timer/counter control register B (TCCR1B). When enabled the noise canceler introduces additional four system clock cycles of delay from a change applied to the input, to the update of the ICR1 register. The noise canceler uses the system clock and is therefore not affected by the prescaler.

12.6.3 Using the Input Capture Unit

The main challenge when using the input capture unit is to assign enough processor capacity for handling the incoming events. The time between two events is critical. If the processor has not read the captured value in the ICR1 register before the next event occurs, the ICR1 will be overwritten with a new value. In this case the result of the capture will be incorrect.

When using the input capture interrupt, the ICR1 register should be read as early in the interrupt handler routine as possible. Even though the input capture interrupt has relatively high priority, the maximum interrupt response time is dependent on the maximum number of clock cycles it takes to handle any of the other interrupt requests.

Using the input capture unit in any mode of operation when the TOP value (resolution) is actively changed during operation, is not recommended.

Measurement of an external signal’s duty cycle requires that the trigger edge is changed after each capture. Changing the edge sensing must be done as early as possible after the ICR1 register has been read. After a change of the edge, the input capture flag (ICF1) must be cleared by software (writing a logical one to the I/O bit location). For measuring frequency only, the clearing of the ICF1 flag is not required (if an interrupt handler is used).

12.7 Output Compare Units

The 16-bit comparator continuously compares TCNT1 with the output compare register (OCR1A/B). If TCNT equals OCR1A/B the comparator signals a match. A match will set the output compare flag (OCF1A/B) at the next timer clock cycle. If enabled (OCIE1A/B = 1), the output compare flag generates an output compare interrupt. The OCF1A/B flag is automatically cleared when the interrupt is executed. Alternatively the OCF1A/B flag can be cleared by software by writing a logical one to its I/O bit locations. The waveform generator uses the match signal to generate an output according to operating mode set by the waveform generation mode (WGM13:0) bits and compare output mode (COM1A/B1:0) bits. The TOP and BOTTOM signals are used by the waveform generator for handling the special cases of the extreme values in some modes of operation (see Section 12.9 “Modes of Operation” on page 115)

The N variable represents the prescaler factor (1, 8, 64, 256, or 1024).

As for the normal mode of operation, the TOV1 flag is set in the same timer clock cycle that the counter counts from MAX to 0x0000.

12.9.3 Fast PWM Mode

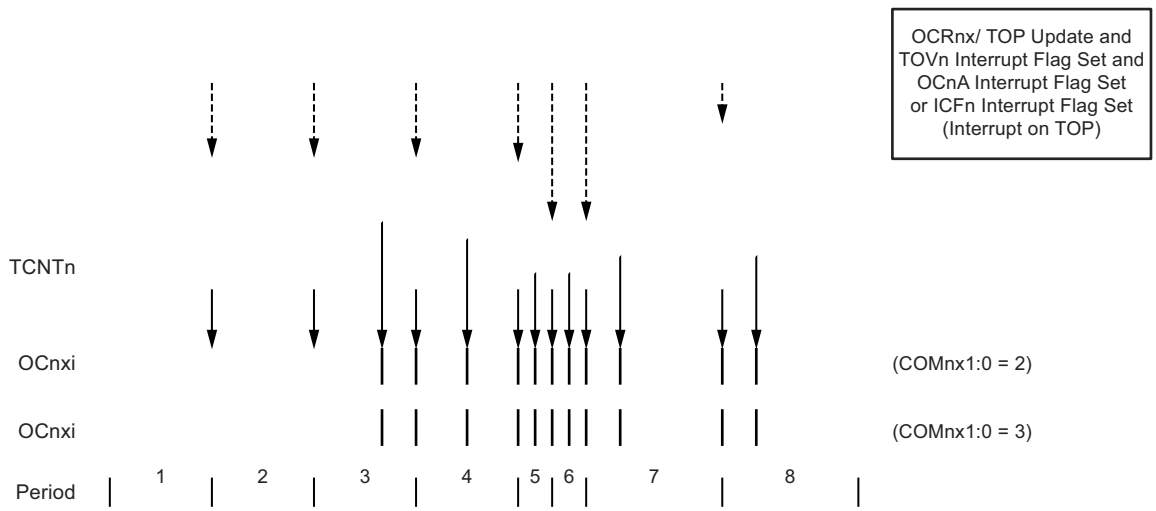
The fast pulse width modulation or fast PWM mode (WGM13:0 = 5, 6, 7, 14, or 15) provides a high frequency PWM waveform generation option. The fast PWM differs from the other PWM options by its single-slope operation. The counter counts from BOTTOM to TOP then restarts from BOTTOM. In non-inverting compare output mode, the output compare (OC1A/B) is set on the compare match between TCNT1 and OCR1A/B, and cleared at TOP. In inverting compare output mode output is cleared on compare match and set at TOP. Due to the single-slope operation, the operating frequency of the fast PWM mode can be twice as high as the phase correct and phase and frequency correct PWM modes that use dual-slope operation. This high frequency makes the fast PWM mode well suited for power regulation, rectification, and DAC applications. High frequency allows physically small sized external components (coils, capacitors), hence reduces total system cost.

The PWM resolution for fast PWM can be fixed to 8-, 9-, or 10-bit, or defined by either ICR1 or OCR1A. The minimum resolution allowed is 2-bit (ICR1 or OCR1A set to 0x0003), and the maximum resolution is 16-bit (ICR1 or OCR1A set to MAX). The PWM resolution in bits can be calculated by using the following equation:

$$R_{FPWM} = \frac{\log(TOP+1)}{\log(2)}$$

In fast PWM mode the counter is incremented until the counter value matches either one of the fixed values 0x00FF, 0x01FF, or 0x03FF (WGM13:0 = 5, 6, or 7), the value in ICR1 (WGM13:0 = 14), or the value in OCR1A (WGM13:0 = 15). The counter is then cleared at the following timer clock cycle. The timing diagram for the fast PWM mode is shown in Figure 12-8. The figure shows fast PWM mode when OCR1A or ICR1 is used to define TOP. The TCNT1 value is in the timing diagram shown as a histogram for illustrating the single-slope operation. The diagram includes non-inverted and inverted PWM outputs. The small horizontal line marks on the TCNT1 slopes represent compare matches between OCR1A/B and TCNT1. The OC1A/B interrupt flag will be set when a compare match occurs.

Figure 12-8. Fast PWM Mode, Timing Diagram



The timer/counter overflow flag (TOV1) is set each time the counter reaches TOP. In addition the OC1A or ICF1 flag is set at the same timer clock cycle as TOV1 is set when either OCR1A or ICR1 is used for defining the TOP value. If one of the interrupts are enabled, the interrupt handler routine can be used for updating the TOP and compare values.

When changing the TOP value the program must ensure that the new TOP value is higher or equal to the value of all of the compare registers. If the TOP value is lower than any of the compare registers, a compare match will never occur between the TCNT1 and the OCR1A/B. Note that when using fixed TOP values the unused bits are masked to zero when any of the OCR1A/B registers are written.

12.11.2 Timer/Counter1 Control Register B – TCCR1B

Bit	7	6	5	4	3	2	1	0	
	ICNC1	ICES1	–	WGM13	WGM12	CS12	CS11	CS10	TCCR1B
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – ICNC1: Input Capture Noise Canceler**

Setting this bit (to one) activates the input capture noise canceler. When the noise canceler is activated, the input from the input capture pin (ICP1) is filtered. The filter function requires four successive equal valued samples of the ICP1 pin for changing its output. The input capture is therefore delayed by four oscillator cycles when the noise canceler is enabled.

- **Bit 6 – ICES1: Input Capture Edge Select**

This bit selects which edge on the input capture pin (ICP1) that is used to trigger a capture event. When the ICES1 bit is written to zero, a falling (negative) edge is used as trigger, and when the ICES1 bit is written to one, a rising (positive) edge will trigger the capture.

When a capture is triggered according to the ICES1 setting, the counter value is copied into the input capture register (ICR1). The event will also set the input capture flag (ICF1), and this can be used to cause an Input capture interrupt, if this interrupt is enabled.

When the ICR1 is used as TOP value (see description of the WGM13:0 bits located in the TCCR1A and the TCCR1B register), the ICP1 is disconnected and consequently the input capture function is disabled.

- **Bit 5 – Reserved Bit**

This bit is reserved for future use. For ensuring compatibility with future devices, this bit must be written to zero when TCCR1B is written.

- **Bit 4:3 – WGM13:2: Waveform Generation Mode**

See TCCR1A register description.

- **Bit 2:0 – CS12:0: Clock Select**

The three Clock Select bits select the clock source to be used by the Timer/Counter, see Figure 12-11 on page 122 and Figure 12-12 on page 122.

Table 12-5. Clock Select Bit Description

CS12	CS11	CS10	Description
0	0	0	No clock source (timer/counter stopped).
0	0	1	$clk_{I/O}$/1 (no prescaling)
0	1	0	$clk_{I/O}$/8 (from prescaler)
0	1	1	$clk_{I/O}$/64 (from prescaler)
1	0	0	$clk_{I/O}$/256 (from prescaler)
1	0	1	$clk_{I/O}$/1024 (from prescaler)
1	1	0	External clock source on T1 pin. Clock on falling edge.
1	1	1	External clock source on T1 pin. Clock on rising edge.

If external pin modes are used for the timer/counter1, transitions on the T1 pin will clock the counter even if the pin is configured as an output. This feature allows software control of the counting.

The 4-bit counter can be both read and written via the data bus, and can generate an overflow interrupt. Both the USI data register and the counter are clocked simultaneously by the same clock source. This allows the counter to count the number of bits received or transmitted and generate an interrupt when the transfer is complete. Note that when an external clock source is selected the counter counts both clock edges. In this case the counter counts the number of edges, and not the number of bits. The clock can be selected from three different sources: The USCK pin, timer/counter0 compare match or from software.

The Two-wire clock control unit can generate an interrupt when a start condition is detected on the two-wire bus. It can also generate wait states by holding the clock pin low after a start condition is detected, or after the counter overflows.

14.3 Functional Descriptions

14.3.1 Three-wire Mode

The USI three-wire mode is compliant to the serial peripheral interface (SPI) mode 0 and 1, but does not have the slave select ($\overline{SS}$) pin functionality. However, this feature can be implemented in software if necessary. Pin names used by this mode are: DI, DO, and USCK.

Figure 14-2. Three-wire Mode Operation, Simplified Diagram

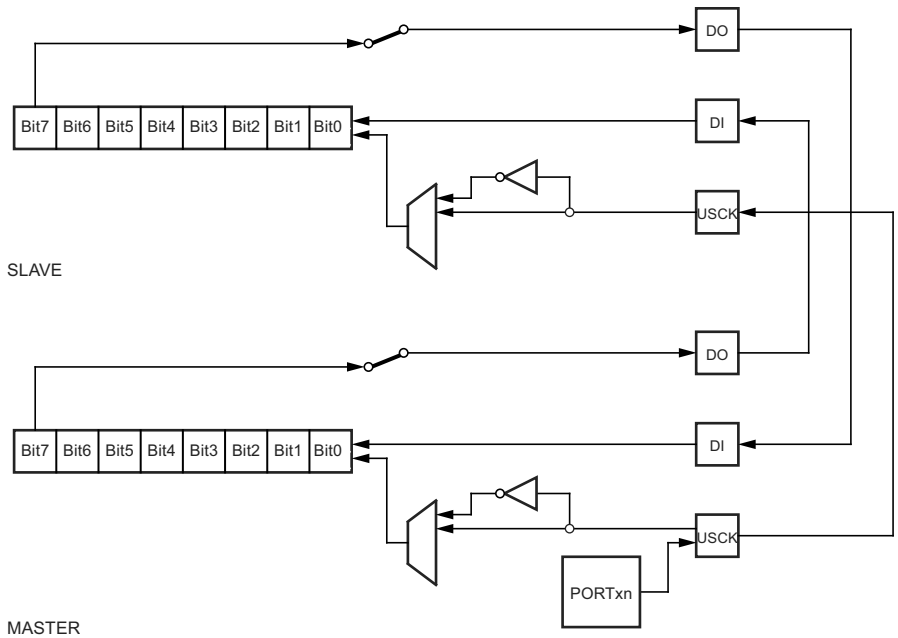


Figure 14-2 shows two USI units operating in three-wire mode, one as master and one as slave. The two USI data register are interconnected in such way that after eight USCK clocks, the data in each register are interchanged. The same clock also increments the USI's 4-bit counter. The counter overflow (interrupt) flag, or USIOIF, can therefore be used to determine when a transfer is completed.

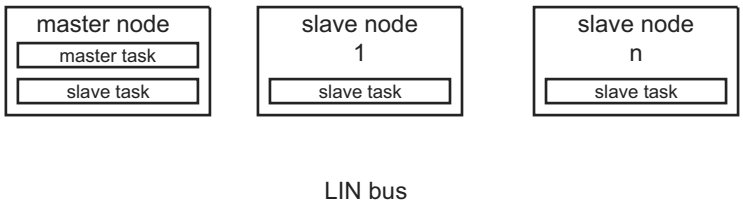
The clock is generated by the master device software by toggling the USCK pin via the PORT register or by writing a one to the USITC bit in USICR.

15.3 LIN Protocol

15.3.1 Master and Slave

A LIN cluster consists of one master task and several slave tasks. A master node contains the master task as well as a slave task. All other nodes contain a slave task only.

Figure 15-1. LIN Cluster with One Master Node and “n” Slave Nodes



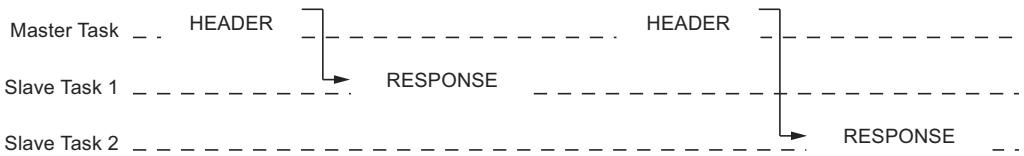
The master task decides when and which frame shall be transferred on the bus. The slave tasks provide the data transported by each frame. Both the master task and the slave task are parts of the frame handler

15.3.2 Frames

A frame consists of a header (provided by the master task) and a response (provided by a slave task).

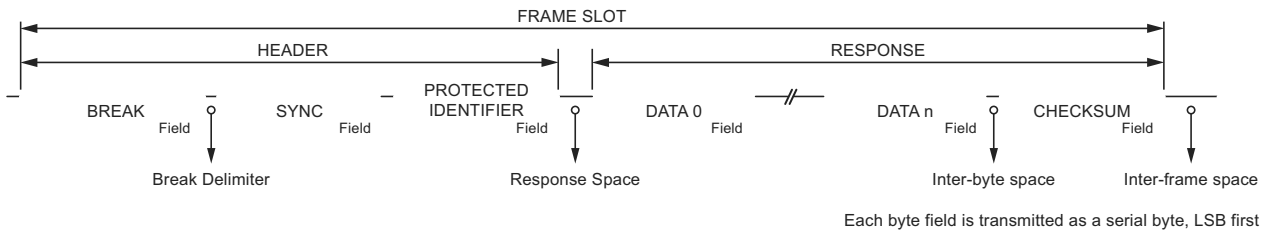
The header consists of a BREAK and SYNC pattern followed by a PROTECTED IDENTIFIER. The identifier uniquely defines the purpose of the frame. The slave task appointed for providing the response associated with the identifier transmits it. The response consists of a DATA field and a CHECKSUM field.

Figure 15-2. Master and Slave Tasks Behavior in LIN Frame



The slave tasks waiting for the data associated with the identifier receives the response and uses the data transported after verifying the checksum.

Figure 15-3. Structure of a LIN Frame



15.3.3 Data Transport

Two types of data may be transported in a frame; signals or diagnostic messages.

- Signals
  Signals are scalar values or byte arrays that are packed into the data field of a frame. A signal is always present at the same position in the data field for all frames with the same identifier.
- Diagnostic messages
  Diagnostic messages are transported in frames with two reserved identifiers. The interpretation of the data field depends on the data field itself as well as the state of the communicating nodes.

17.11.2 ADCSRA – ADC Control and Status Register A

Bit	7	6	5	4	3	2	1	0	
	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	ADCSRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – ADEN: ADC Enable**

Writing this bit to one enables the ADC. By writing it to zero, the ADC is turned off. Turning the ADC off while a conversion is in progress, will terminate this conversion.

- **Bit 6 – ADSC: ADC Start Conversion**

In single conversion mode, write this bit to one to start each conversion. In free running mode, write this bit to one to start the first conversion. The first conversion after ADSC has been written after the ADC has been enabled, or if ADSC is written at the same time as the ADC is enabled, will take 25 ADC clock cycles instead of the normal 13. This first conversion performs initialization of the ADC.

ADSC will read as one as long as a conversion is in progress. When the conversion is complete, it returns to zero. Writing zero to this bit has no effect.

- **Bit 5 – ADATE: ADC Auto Trigger Enable**

When this bit is written to one, auto triggering of the ADC is enabled. The ADC will start a conversion on a positive edge of the selected trigger signal. The trigger source is selected by setting the ADC trigger select bits, ADTS in ADCSRB.

- **Bit 4 – ADIF: ADC Interrupt Flag**

This bit is set when an ADC conversion completes and the data registers are updated. The ADC conversion complete interrupt is executed if the ADIE bit and the I-bit in SREG are set. ADIF is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, ADIF is cleared by writing a logical one to the flag. Beware that if doing a read-modify-write on ADCSRA, a pending interrupt can be disabled. This also applies if the SBI and CBI instructions are used.

- **Bit 3 – ADIE: ADC Interrupt Enable**

When this bit is written to one and the I-bit in SREG is set, the ADC conversion complete interrupt is activated.

- **Bits 2:0 – ADPS2:0: ADC Prescaler Select Bits**

These bits determine the division factor between the system clock frequency and the input clock to the ADC.

Table 17-6. ADC Prescaler Selections

ADPS2	ADPS1	ADPS0	Division Factor
0	0	0	2
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

17.11.7 AMISCR – Analog Miscellaneous Control Register

Bit	7	6	5	4	3	2	1	0	
	-	-	-	-	-	AREFEN	XREFEN	ISRCEN	AMISCR
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bits 7:3 – Reserved Bits**

These bits are reserved for future use. For compatibility with future devices, they must be written to zero when AMISCR is written.

- **Bit 2 – AREFEN: External Voltage Reference Input Enable**

When this bit is written logic one, the voltage reference for the ADC is input from AREF pin as described in Table 17.10 on page 188. If active channels are used, using AVcc or an external AREF higher than (AVcc - 1V) is not recommended, as this will affect ADC accuracy. The internal voltage reference options may not be used if an external voltage is being applied to the AREF pin. It is recommended to use DIDR register bit function (digital input disable) when AREFEN is set.

- **Bit 1 – XREFEN: Internal Voltage Reference Output Enable**

When this bit is written logic one, the internal voltage reference 1.1V or 2.56V is output on XREF pin as described in Table 17.10 on page 188. It is recommended to use DIDR register bit function (digital input disable) when XREFEN is set.

18.1.3 DIDR0 – Digital Input Disable Register 0

Bit	7	6	5	4	3	2	1	0	
	ADC7D / AIN1D	ADC6D / AIN0D	ADC5D	ADC4D	ADC3D	ADC2D	ADC1D	ADC0D	DIDR0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bits 7,6 – AIN1D, AIN0D: AIN1D and AIN0D Digital Input Disable**

When this bit is written logic one, the digital input buffer on the corresponding analog compare pin is disabled. The corresponding PIN register bit will always read as zero when this bit is set. When an analog signal is applied to the AIN0/1 pin and the digital input from this pin is not needed, this bit should be written logic one to reduce power consumption in the digital input buffer.

18.2 Analog Comparator Inputs

18.2.1 Analog Compare Positive Input

It is possible to select any of the inputs of the ADC positive input multiplexer to replace the positive input to the analog comparator. The ADC multiplexer is used to select this input, and consequently, the ADC must be switched off to utilize this feature. If the analog comparator multiplexer enable bit (ACME in ADCSRB register) is set and the ADC is switched off (ADEN in ADCSRA register is zero), MUX[4..0] in ADMUX register select the input pin to replace the positive input to the analog comparator, as shown in Table 18-2 on page 196. If ACME is cleared or ADEN is set, AIN1 pin is applied to the positive input to the analog comparator.

Table 18-2. Analog Comparator Positive Input

ACME	ADEN	MUX[4..0]	Analog Comparator Positive Input - Comment	
0	x	x xxxx$_b$	AIN1	ADC Switched On
x	1	x xxxx$_b$	AIN1	
1	0	0 0000$_b$	ADC0	ADC Switched Off.
1	0	0 0001$_b$	ADC1	
1	0	0 0010$_b$	ADC2	
1	0	0 0011$_b$	ADC3 / ISRC	
1	0	0 0100$_b$	ADC4	
1	0	0 0101$_b$	ADC5	
1	0	0 0110$_b$	ADC6	
1	0	0 0111$_b$	ADC7	
1	0	0 1000$_b$	ADC8	
1	0	0 1001$_b$	ADC9	
1	0	0 1010$_b$	ADC10	
1	0	Other	This doesn't make sense - Don't use.	

22.4.3 External Clock Drive

Table 22-2. External Clock Drive

Parameter	Symbol	Vcc = 2.7 - 5.5V		Vcc = 4.5 - 5.5V		Units
		Min.	Max.	Min.	Max.	
Oscillator frequency	$1/t_{CLCL}$	0	8	0	16	MHz
Clock period	t_{CLCL}	125		62.5		ns
High time	t_{CHCX}	50		25		ns
Low time	t_{CLCX}	50		25		ns
Rise time	t_{CLCH}		1.6		0.5	ms
Fall time	t_{CHCL}		1.6		0.5	ms
Change in period from one clock cycle to the next	Δt_{CLCL}		2		2	%

22.5 RESET Characteristics

Table 22-3. External Reset Characteristics

Parameter	Condition	Symbol	Min	Typ	Max	Units
$\overline{\text{RESET}}$ pin threshold voltage	V_{CC} = 5V	V_{RST}	0.1 Vcc		0.9 Vcc	V
Minimum pulse width on $\overline{\text{RESET}}$ pin	V_{CC} = 5V	t_{RST}			2.5	µs
Bandgap reference voltage	V_{CC} = 2.7V, T_A = 25°C	V_{BG}	1.0	1.1	1.2	V
Bandgap reference start-up time	V_{CC} = 2.7V, T_A = 25°C	t_{BG}		40	70	µs
Bandgap reference current consumption	V_{CC} = 2.7V, T_A = 25°C	I_{BG}		15		µA

Table 22-4. Power On Reset Characteristics

Parameter	Symbol	Min	Typ	Max	Units
Power-on reset threshold voltage (rising)	VPOT		1.4		V
Power-on reset threshold voltage (falling)(1)		1.0	1.3	1.6	V
VCC max. start voltage to ensure internal power-on reset signal	VPORMAX			0.4	V
VCC Min. start voltage to ensure internal power-on reset signal	VPORMIN	–0.1			V
VCC rise rate to ensure power-on reset	VCCRR	0.01			V/ms
$\overline{\text{RESET}}$ pin threshold voltage	VRST	0.1 Vcc		0.9 Vcc	V

Note: 1. Before rising, the supply has to be between VPORMIN and VPORMAX to ensure a reset.

24.3 Supply Current of I/O modules

The table below can be used to calculate the additional current consumption for the different I/O modules idle mode. The enabling or disabling of the I/O modules are controlled by the power reduction register. See Section 5.9.3 “PRR – Power Reduction Register” on page 46 for details.

Table 24-1. Additional Current Consumption for the different I/O modules (absolute values)

Module	Vcc = 5.0V Freq. = 16MHz	Vcc = 5.0V Freq. = 8MHz	Vcc = 3.0V Freq. = 8MHz	Vcc = 3.0V Freq. = 4MHz	Units
LIN/UART	0.77	0.37	0.20	0.10	mA
SPI	0.31	0.14	0.08	0.04	mA
TIMER-1	0.28	0.13	0.08	0.04	mA
TIMER-0	0.41	0.20	0.10	0.05	mA
USI	0.14	0.05	0.04	0.02	mA
ADC	0.48	0.22	0.10	0.05	mA

24.4 Power-down Supply Current

Figure 24-8. Power-down Supply Current versus Vcc (Watchdog Timer Disabled)

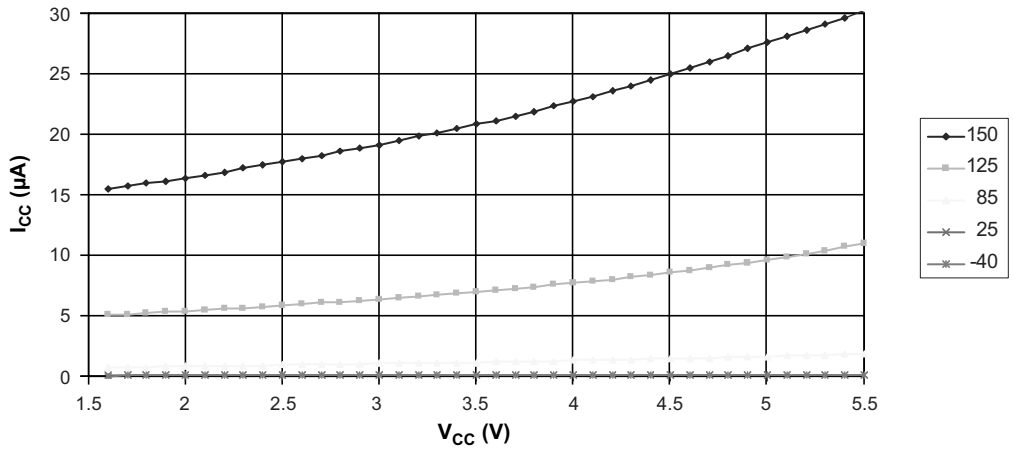


Figure 24-9. Power-down Supply Current versus Vcc (Watchdog Timer Enabled)

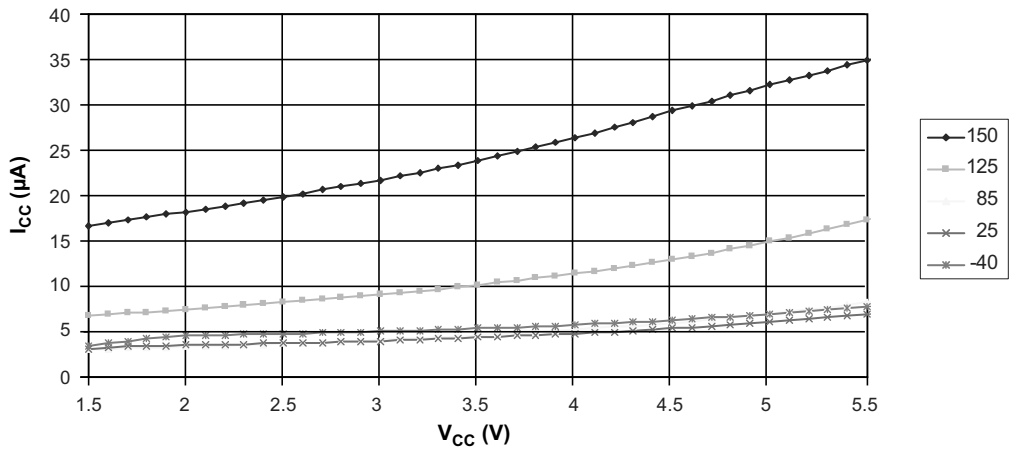
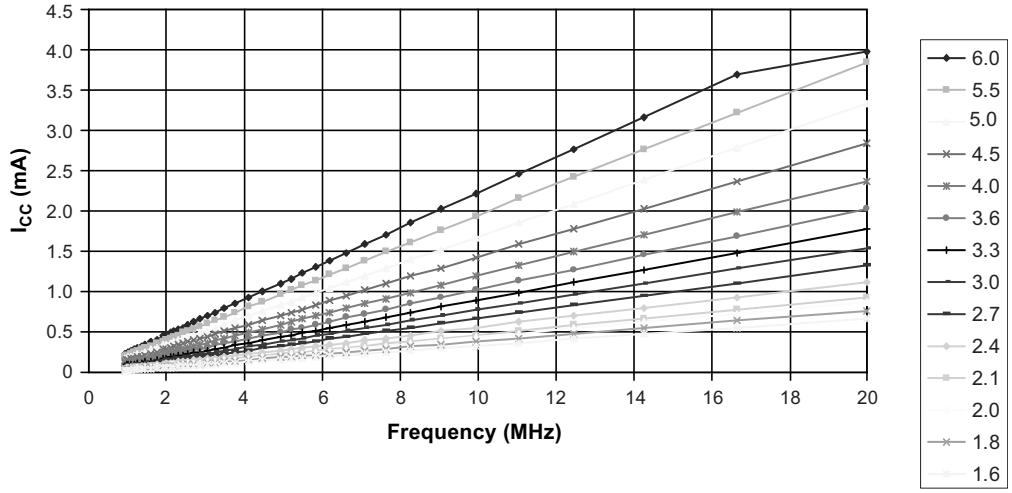


Figure 24-21. Reset Supply Current versus Vcc, Frequencies ≥ 1MHz (Excluding Current Through the Reset Pull-up)



Atmel | Enabling Unlimited Possibilities®

Atmel Corporation 1600 Technology Drive, San Jose, CA 95110 USA **T:** (+1)(408) 441.0311 **F:** (+1)(408) 436.4200 | **www.atmel.com**