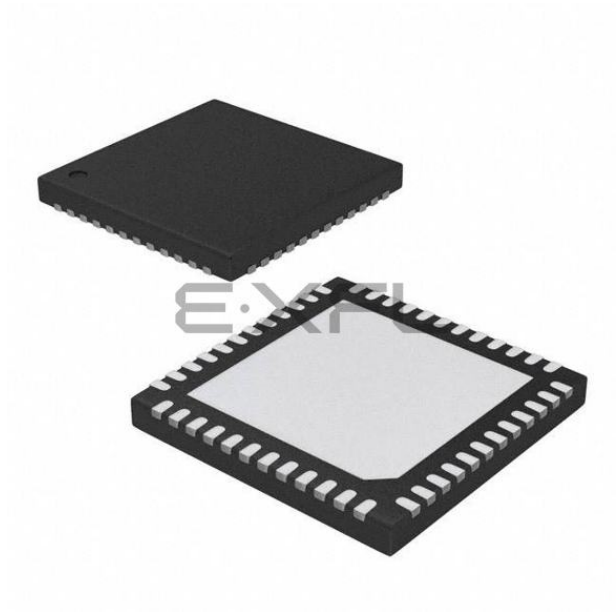


Welcome to E-XFL.COM

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.

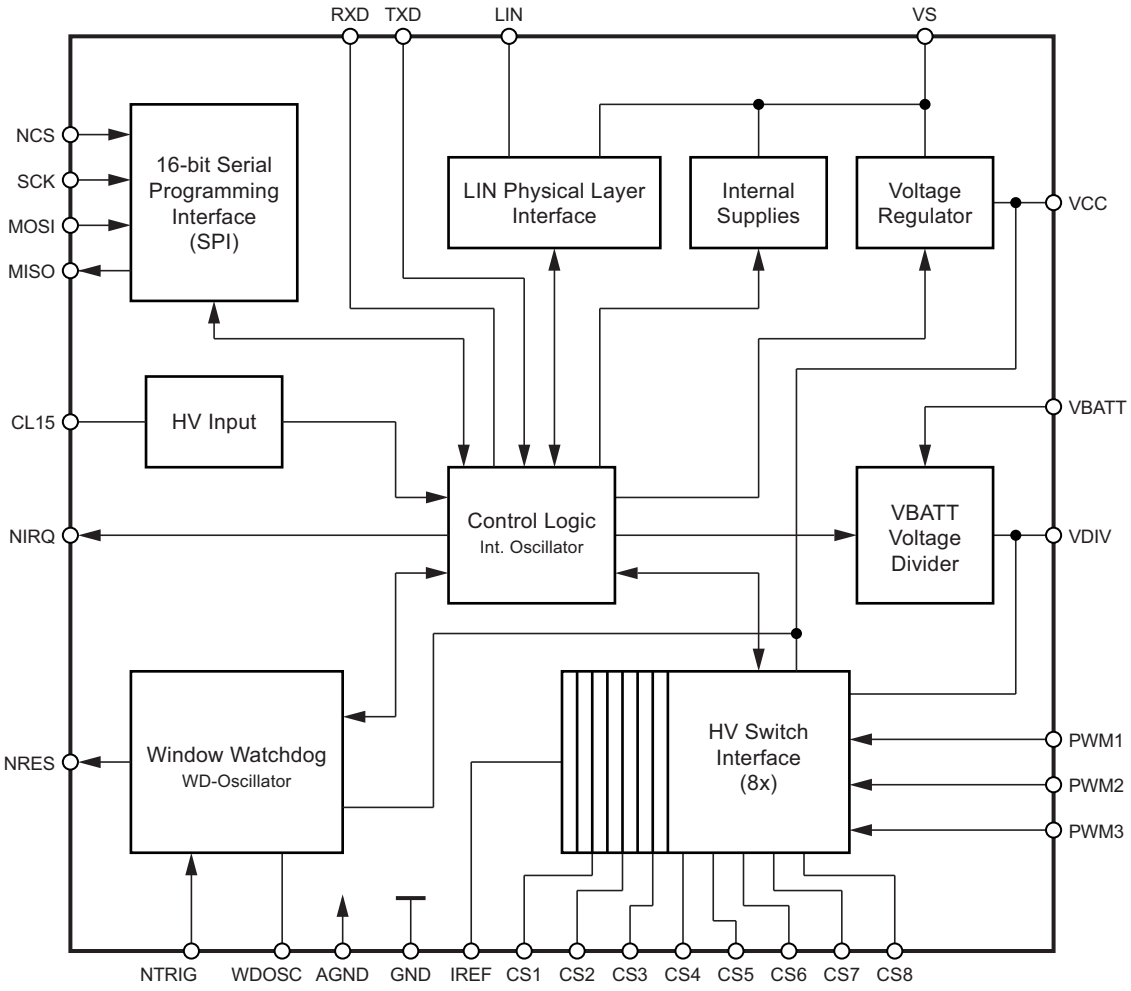
Applications of "Embedded - Microcontrollers"

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, WDT
Number of I/O	16
Program Memory Size	16KB (16K x 8)
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 (TC)
Mounting Type	Surface Mount
Package / Case	48-VFQFN Exposed Pad
Supplier Device Package	48-VQFN (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/ata664251-wgqw-1

4.2 Description

The LIN-SBC includes an eight-channel high voltage switch interface, a LIN 2.1 and SAEJ2602-2-compliant LIN transceiver, low-drop voltage regulator, and an adjustable Window Watchdog. The voltage regulator has an output voltage of 5V and is able to drive 80mA. This chip combination is especially designed for LIN switch applications. It is designed to handle low data-rate communication in vehicles (such as in convenience electronics). Improved slope control at the LIN driver ensures secure data communication up to 20kBaud. Sleep Mode and Active Low-power Mode guarantee minimal current consumption even in the case of a floating bus line or a short circuit on the LIN bus to GND.

Figure 4-1. Block Diagram

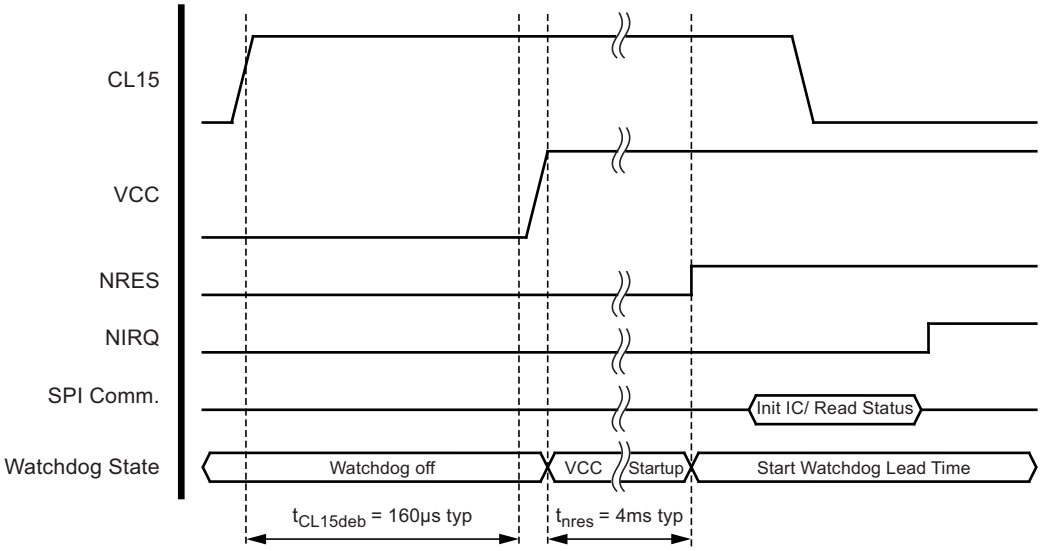


4.4.2.2 Wake-up from Sleep Mode via CL15

Voltage above V_{CL15H} at pin CL15 activates a CL15 wake-up detection phase. This state must persist for at least t_{CLdeb} in order to detect a wake-up. If the pulse is too short, the IC remains in Sleep Mode.

When leaving Sleep Mode first the VCC voltage regulator is activated to enable the microcontroller supply. Then as soon as the VCC level reaches valid levels, the VCC startup timer is started. During this time, the NRES pin is kept low in order to keep the microcontroller from running. This ensures a proper voltage supply and signal stabilization in the application. With the rising edge at NRES, the SPI is ready for communication and the LIN-SBC can be initialized.

Figure 4-6. CL15 Wake-up from Sleep Mode



The wake-up behavior is analogous to a wake-up via the LIN bus as seen above. One difference is that no negative edge is required to start the wake-up procedure as is the case for LIN wake-ups. After the VCC startup time t_{WDnres} has elapsed, NRES is released and therefore pulled up, either by the internal or additional external resistors. The microcontroller can then configure the LIN-SBC and thus be notified about the actual status including the wake-up source. Here, the two status bits “IRQS1” and “IRQS0” read back as ‘10’.

4.15 Electrical Characteristics (Continued)

5V < V_S < 27V, –40°C < T_j < 150°C, chip configuration as default, unless otherwise specified. All values refer to GND pins

No.	Parameters	Test Conditions	Pin	Symbol	Min.	Typ.	Max.	Unit	Type*
20.23	Current source falling voltage slope, slope control disabled	V_{VS} = 14V I_{IREF} = 100µA V_{CSx} = 0V 90% to 25% CSSCD = 1	CSx	$dU_{CSx,fallfast}$	6.5		30	V/µs	C
20.9	Output voltage on IREF pin	V_{VS} ≥ 7V 10µA ≤ I_{IREF} ≤ 250µA At least one current source active	IREF	V_{IREF}	1.19	1.23	1.27	V	A
20.10	Internally generated IREF fail-safe current in case of open or shorted IREF pin	V_{IREF} = 0V I_{IREF} = 0µA	IREF	I_{IREFfs}	60 60		140 140	µA	A
20.11	Switch input debouncing time	Time from voltage level change on pin CSx to signal state change visible in SPI register	CSx	t_{CSxdeb}	2		12	µs	B
20.12	Switch input leakage current	Current source and voltage divider off V_{CSx} = 0 V V_{CSx} = V_{VS}	CSx	$I_{CSx,leak}$	–3		+3	µA	A
20.13	Current source enabling time	V_{VS} = 14V V_{CSx} = 0V(H)/V_{CSx} = 14V(L) I_{IREF} = 100µA Test time until abs(I_{CSx}) ≥ 9.5mA	CSx	t_{CSxon}	3		8	µs	A
20.14	Current source shutdown time	V_{VS} = 14V V_{CSx} = 0V(H)/V_{CSx} = 14V(L) I_{IREF} = 100µA Test time until abs(I_{CSx}) ≤ 0.5mA	CSx	$t_{CSx,of}$	3		8	µs	A
20.15	Voltage divider resistance	V_{CSx} = 4V	CSx	R_{CSxdiv}	50	95	150	kΩ	A
20.16	Voltage divider precision	V_{CSx} = 2V	CSx	p_{CSxdiv}	–3		+3	%	A
20.17	Maximum current source switching frequency		CSx	fCSx,max			20	kHz	D
20.18	Maximum voltage level for logic “low”		PWM1..3	$V_{PWML,max}$			0.33	V_{VCC}	A
20.19	Minimum voltage-level for logic “high”		PWM1..3	$V_{PWMH,min}$	0.66			V_{VCC}	A
20.20	PWM input leakage current, low level	V_{PWMy} = 0	PWM1..3	$I_{PWMleakL}$	–1			µA	A
20.21	PWM input pull-down resistor value	V_{PWMy} = V_{VCC}	PWM1..3	R_{PWM}	60	100	220	kΩ	A

*) Type means: A = 100% tested, B = 100% correlation tested, C = Characterized on samples, D = Design parameter

5.5.2.1 Default Clock Source

At reset, the CKSEL and SUT fuse settings are copied into the CLKSELR register. The device will then use the clock source and the start-up timings defined by the CLKSELR bits (CSEL3..0 and CSUT1:0).

The device is shipped with CKSEL Fuses = 0010_b, SUT Fuses = 10_b, and CKDIV8 Fuse programmed. The default clock source setting is therefore the Internal RC Oscillator running at 8MHz with the longest start-up time and an initial system clock divided by 8. This default setting ensures that all users can make their desired clock source setting using an In-System or High-voltage Programmer. This set-up must be taken into account when using ISP tools.

5.5.2.2 Calibrated Internal RC Oscillator

By default, the Internal RC Oscillator provides an approximate 8.0MHz clock. The frequency is nominal at 5V and 25°C. Though voltage and temperature dependent, this clock can be accurately calibrated by the user. See Table 5-80 on page 253 and Section 5.25.7 “Internal Oscillator Speed” on page 270 for more details.

If selected, it can operate without external components. At reset, hardware loads the pre-programmed calibration value into the OSCCAL Register and thereby automatically configuring the RC Oscillator. At 5V and 25° C, this calibration gives a frequency of 8MHz ±1%. The tolerance of the internal RC oscillator remains better than ±10% within the whole automotive temperature and voltage ranges (4.5V to 5.5V, –40°C to +125°C). The accuracy of this calibration is shown as Factory calibration in Table 5-80 on page 253.

By adjusting the OSCCAL register in software, see Section 5.5.5.1 “OSCCAL – Oscillator Calibration Register” on page 74, it is possible to get a higher calibration accuracy than by using the factory calibration. The accuracy of this calibration is shown as User calibration in Table 5-80 on page 253.

The Watchdog Oscillator will still be used for the Watchdog Timer and for the Reset Time-out even when this Oscillator is used as the device clock. For more information on the pre-programmed calibration value, see the section Section 5.22.4 “Calibration Byte” on page 237.

Table 5-6. Internal Calibrated RC Oscillator Operating Modes(1)

Frequency Range(2) (MHz)	CKSEL3..0(3, 4) CSEL3..0(5)
7.6 - 8.4	0010

Notes:
1. If 8MHz frequency exceeds the specification of the device (depends on Vcc), the CKDIV8 fuse can be programmed in order to divide the internal frequency by 8
2. The frequency ranges are guideline values
3. The device is shipped with this CKSEL = “0010”
4. Flash Fuse bits.
5. CLKSELR register bits

When this Oscillator is selected, start-up times are determined by the SUT Fuses or by CSUT field as shown in Table 5-7.

Table 5-7. Start-up Times for the Internal Calibrated RC Oscillator Clock Selection

SUT1..0(1) CSUT1..0(2)	Start-up Time from Power-down/save	Additional Delay from Reset (Vcc = 5.0V)	Recommended Usage
00(3)	6 CK	14CK	BOD enabled
01	6 CK	14CK + 4.1ms	Fast rising power
10(4)	6 CK	14CK + 65ms	Slowly rising power
11	Reserved		

Notes:
1. Flash Fuse bits
2. CLKSELR register bits
3. This setting is only available if RSTDISBL fuse is not set
4. The device is shipped with this option selected

Here is a “light” C-code that describes such a sequence of commands.

C Code Example

```
void ClockSwiching (unsigned char clk_number, unsigned char sut) {

#define CLOCK_RECOVER                   0x05
#define CLOCK_ENABLE                    0x02
#define CLOCK_SWITCH                    0x04
#define CLOCK_DISABLE                   0x01

unsigned char previous_clk, temp;

        // Disable interrupts
        temp = SREG; asm ("cli");
        // Save the current system clock source
        CLKCSR = 1 << CLKCCE;
        CLKCSR = CLOCK_RECOVER;
        previous_clk = CLKSELR & 0x0F;
        // Enable the new clock source
        CLKSELR = ((sut << 4 ) & 0x30) | (clk_number & 0x0F);
        CLKCSR = 1 << CLKCCE;
        CLKCSR = CLOCK_ENABLE;
        // Wait for clock validity
        while ((CLKCSR & (1 << CLKRDY)) == 0);
        // Switch clock source
        CLKCSR = 1 << CLKCCE;
        CLKCSR = CLOCK_SWITCH;
        // Wait for effective switching
        while (1)     {
           CLKCSR = 1 << CLKCCE;
           CLKCSR = CLOCK_RECOVER;
           if ((CLKSELR & 0x0F) == (clk_number & 0x0F)) break;
        }
        // Shut down unneeded clock source
        if (previous_clk != (clk_number & 0x0F)) {
           CLKSELR = previous_clk;
           CLKCSR = 1 << CLKCCE;
           CLKCSR = CLOCK_DISABLE;
        }
        // Re-enable interrupts
        SREG = temp;
}
```

Warning: In the Atmel® ATtiny87/167, only one among the three external clock sources can be enabled at a given time. Moreover, the enables of the external clock and of the external low-frequency oscillator are shared with the asynchronous timer.

5.6.4 ADC Noise Reduction Mode

When the SM1..0 bits are written to 01, the SLEEP instruction makes the MCU enter ADC Noise Reduction mode, stopping the CPU but allowing the ADC, the external interrupts, the USI start condition, the asynchronous Timer/Counter and the Watchdog to continue operating (if enabled). This sleep mode basically halts $clk_{I/O}$, clk_{CPU}, and clk_{FLASH}, while allowing the other clocks to run.

This improves the noise environment for the ADC, enabling higher resolution measurements. If the ADC is enabled, a conversion starts automatically when this mode is entered. Apart from the ADC Conversion Complete interrupt, only an External Reset, a Watchdog System Reset, a Watchdog Interrupt, a Brown-out Reset, a USI start condition interrupt, an asynchronous Timer/Counter interrupt, an SPM/EEPROM ready interrupt, an external level interrupt on INT0 or INT1 or a pin change interrupt can wake up the MCU from ADC Noise Reduction mode.

5.6.5 Power-down Mode

When the SM1..0 bits are written to 10, the SLEEP instruction makes the MCU enter Power-down mode. In this mode, the external Oscillator is stopped, while the external interrupts, the USI start condition, and the Watchdog continue operating (if enabled). Only an External Reset, a Watchdog System Reset, a Watchdog Interrupt, a Brown-out Reset, the USI start condition interrupt, an external level interrupt on INT0 or INT1, or a pin change interrupt can wake up the MCU. This sleep mode basically halts all generated clocks, allowing operation of asynchronous modules only.

Note that if a level triggered interrupt is used for wake-up from Power-down mode, the changed level must be held for some time to wake up the MCU. Refer to Section 5.9 “External Interrupts” on page 96 for details.

When waking up from Power-down mode, there is a delay from the wake-up condition occurs until the wake-up becomes effective. This allows the clock to restart and become stable after having been stopped. The wake-up period is defined by the same CKSEL Fuses that define the Reset Time-out period, as described in Section 5.5.2 “Clock Sources” on page 63.

5.6.6 Power-save Mode

When the SM1..0 bits are written to 11, the SLEEP instruction makes the MCU enter Power-save mode. This mode is identical to Power-down, with one exception.

If Timer/Counter0 is clocked asynchronously, i.e., the AS0 bit in ASSR is set, Timer/Counter0 will run during sleep. The device can wake up from either Timer Overflow or Output Compare event from Timer/Counter0 if the corresponding Timer/Counter0 interrupt enable bits are set in TIMSK0, and the global interrupt enable bit in SREG is set.

If the Asynchronous Timer is **NOT** clocked asynchronously, Power-down mode is recommended instead of Power-save mode because the contents of the registers in the asynchronous timer should be considered undefined after wake-up in Power-save mode if AS0 is 0.

This sleep mode basically halts all clocks except clk_{ASY}, allowing operation only of asynchronous modules, including Timer/Counter0 if clocked asynchronously.

5.6.7 Power Reduction Register

The Power Reduction Register (PRR), see “PRR – Power Reduction Register” on page 82, provides a method to stop the clock to individual peripherals to reduce power consumption. The current state of the peripheral is frozen and the I/O registers can not be read or written. Resources used by the peripheral when stopping the clock will remain occupied, hence the peripheral should in most cases be disabled before stopping the clock. Waking up a module, which is done by clearing the bit in PRR, puts the module in the same state as before shutdown.

Module shutdown can be used in Idle mode and Active mode to significantly reduce the overall power consumption. In all other sleep modes, the clock is already stopped.

5.6.8 Minimizing Power Consumption

There are several possibilities to consider when trying to minimize the power consumption in an AVR® controlled system. In general, sleep modes should be used as much as possible, and the sleep mode should be selected so that as few as possible of the device’s functions are operating. All functions not needed should be disabled. In particular, the following modules may need special consideration when trying to achieve the lowest possible power consumption.

5.6.8.1 Analog to Digital Converter

If enabled, the ADC will be enabled in all sleep modes. To save power, the ADC should be disabled before entering any sleep mode. When the ADC is turned off and on again, the next conversion will be an extended conversion. Refer to Section 5.18 “ADC – Analog to Digital Converter” on page 205 for details on ADC operation.

5.6.8.2 Analog Comparator

When entering Idle mode, the Analog Comparator should be disabled if not used. When entering ADC Noise Reduction mode, the Analog Comparator should be disabled. In other sleep modes, the Analog Comparator is automatically disabled. However, if the Analog Comparator is set up to use the Internal Voltage Reference as input, the Analog Comparator should be disabled in all sleep modes. Otherwise, the Internal Voltage Reference will be enabled, independent of sleep mode. Refer to Section 5.19 “AnaComp – Analog Comparator” on page 223 for details on how to configure the Analog Comparator.

5.6.8.3 Brown-out Detector

If the Brown-out Detector is not needed by the application, this module should be turned off. If the Brown-out Detector is enabled by the BODLEVEL Fuses, it will be enabled in all sleep modes, and hence, always consume power. In the deeper sleep modes, this will contribute significantly to the total current consumption. Refer to Section 5.7.1.5 “Brown-out Detection” on page 85 for details on how to configure the Brown-out Detector.

5.6.8.4 Internal Voltage Reference

The Internal Voltage Reference will be enabled when needed by the Brown-out Detection, the Analog Comparator or the ADC. If these modules are disabled as described in the sections above, the internal voltage reference will be disabled and it will not be consuming power. When turned on again, the user must allow the reference to start up before the output is used. If the reference is kept on in sleep mode, the output can be used immediately. Refer to Section 5.7.2 “Internal Voltage Reference” on page 87 for details on the start-up time.

Output the internal voltage reference is not needed in the deeper sleep modes. This module should be turned off to reduce significantly to the total current consumption. Refer to Section 5.17.3.1 “AMISCR – Analog Miscellaneous Control Register” on page 205 for details on how to disable the internal voltage reference output.

5.6.8.5 Internal Current Source

The Internal Current Source is not needed in the deeper sleep modes. This module should be turned off to reduce significantly to the total current consumption. Refer to Section 5.17.3.1 “AMISCR – Analog Miscellaneous Control Register” on page 205 for details on how to disable the Internal Current Source.

5.6.8.6 Watchdog Timer

If the Watchdog Timer is not needed in the application, the module should be turned off. If the Watchdog Timer is enabled, it will be enabled in all sleep modes and hence always consume power. In the deeper sleep modes, this will contribute significantly to the total current consumption. Refer to Section 5.7.3 “Watchdog Timer” on page 87 for details on how to configure the Watchdog Timer.

5.6.8.7 Port Pins

When entering a sleep mode, all port pins should be configured to use minimum power. The most important is then to ensure that no pins drive resistive loads. In sleep modes where both the I/O clock ($clk_{I/O}$) and the ADC clock (clk_{ADC}) are stopped, the input buffers of the device will be disabled. This ensures that no power is consumed by the input logic when not needed. In some cases, the input logic is needed for detecting wake-up conditions, and it will then be enabled. Refer to the section Section 5.10.2.6 “Digital Input Enable and Sleep Modes” on page 104 for details on which pins are enabled. If the input buffer is enabled and the input signal is left floating or have an analog signal level close to Vcc/2, the input buffer will use excessive power.

For analog input pins, the digital input buffer should be disabled at all times. An analog signal level close to Vcc/2 on an input pin can cause significant current even in active mode. Digital input buffers can be disabled by writing to the Digital Input Disable Registers (DIDR1 and DIDR0). Refer to Section 5.18.12.6 “DIDR1 – Digital Input Disable Register 1” on page 222 and Section 5.18.12.5 “DIDR0 – Digital Input Disable Register 0” on page 221 for details.

5.6.8.8 On-chip Debug System

If the On-chip debug system is enabled by the DWEN Fuse and the chip enters sleep mode, the main clock source is enabled and hence always consumes power. In the deeper sleep modes, this will contribute significantly to the total current consumption.

5.7.2 Internal Voltage Reference

Atmel® ATtiny87/167 features an internal bandgap reference. This reference is used for Brown-out Detection, and it can be used as an input to the Analog Comparator or the ADC.

5.7.2.1 Voltage Reference Enable Signals and Start-up Time

The voltage reference has a start-up time that may influence the way it should be used. The start-up time is given in Table 5-86 on page 255. To save power, the reference is not always turned on. The reference is on during the following situations:

1. When the BOD is enabled (by programming the BODLEVEL [2:0] Fuses).
2. When the bandgap reference is connected to the Analog Comparator (by setting the ACIRS bit in ACSR).
3. When the ADC is enabled.

Thus, when the BOD is not enabled, after setting the ACIRS bit or enabling the ADC, the user must always allow the reference to start up before the output from the Analog Comparator or ADC is used. To reduce power consumption in Power-down mode or in Power-save, the user can avoid the three conditions above to ensure that the reference is turned off before entering in these power reduction modes.

5.7.3 Watchdog Timer

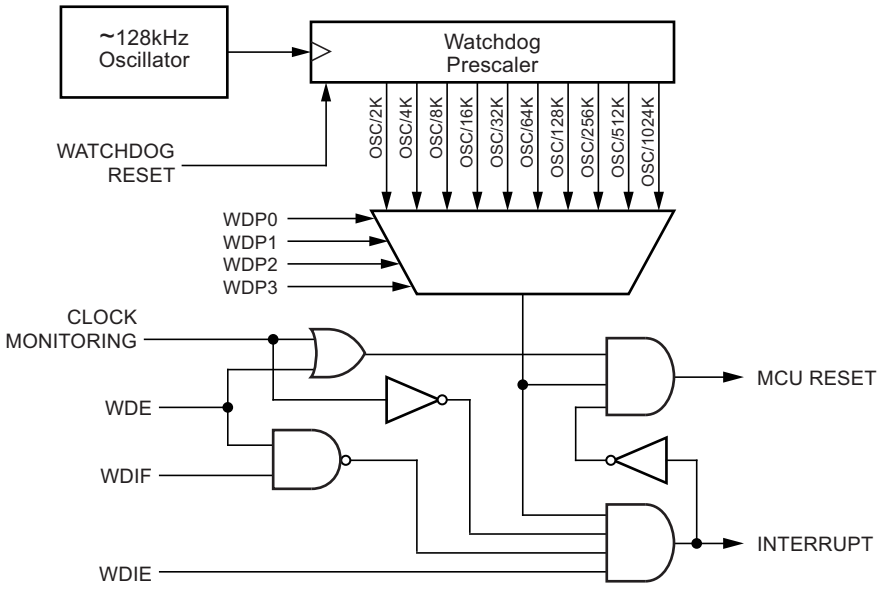
Atmel® ATtiny87/167 has an Enhanced Watchdog Timer (WDT). The main features are:

- Clocked from separate On-chip Oscillator
- Four operating modes
  - Interrupt
  - System Reset
  - Interrupt and System Reset
  - Clock Monitoring
- Selectable Time-out period from 16ms to 8s
- Possible Hardware fuse Watchdog always on (WDTON) for fail-safe mode

5.7.3.1 Watchdog Timer Behavior

The Watchdog Timer (WDT) is a timer counting cycles of a separate on-chip 128kHz oscillator.

Figure 5-22. Watchdog Timer



5.10.3.4 Alternate Functions of Port B

The Port B pins with alternate functions are shown in Table 5-26.

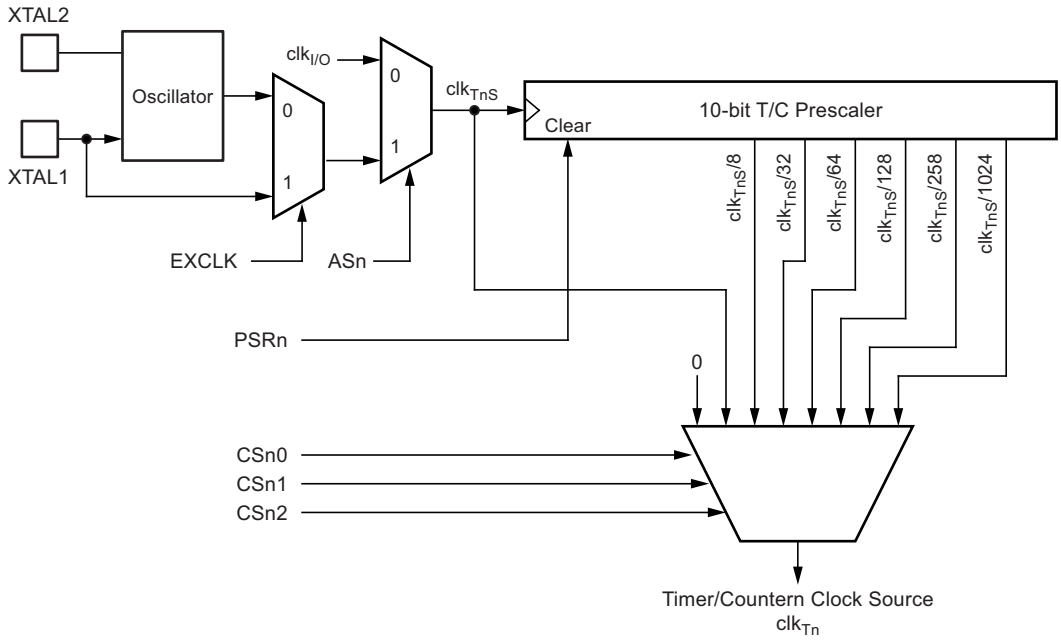
Table 5-26. Port B Pins Alternate Functions

Port Pin	Alternate Functions
PB7	PCINT15 (Pin Change Interrupt 15) ADC10 (ADC Input Channel 10) OC1BX (Output Compare and PWM Output B-X for Timer/Counter1) $\overline{\text{RESET}}$ (Reset input pin) dW (debugWIRE I/O)
PB6	PCINT14 (Pin Change Interrupt 14) ADC9 (ADC Input Channel 9) OC1AX (Output Compare and PWM Output A-X for Timer/Counter1) INT0 (External Interrupt0 Input)
PB5	PCINT13 (Pin Change Interrupt 13) ADC8 (ADC Input Channel 8) OC1BW (Output Compare and PWM Output B-W for Timer/Counter1) XTAL2 (Chip clock Oscillator pin 2) CLKO (System clock output)
PB4	PCINT12 (Pin Change Interrupt 12) OC1AW (Output Compare and PWM Output A-W for Timer/Counter1) XTAL1 (Chip clock Oscillator pin 1) CLKI (External clock input)
PB3	PCINT11 (Pin Change Interrupt 11) OC1BV (Output Compare and PWM Output B-V for Timer/Counter1)
PB2	PCINT10 (Pin Change Interrupt 10) OC1AV (Output Compare and PWM Output A-V for Timer/Counter1) USCK (Three-wire Mode USI *Default* Clock Input) SCL (Two-wire Mode USI *Default* Clock Input)
PB1	PCINT9 (Pin Change Interrupt 9) OC1BU (Output Compare and PWM Output B-U for Timer/Counter1) DO (Three-wire Mode USI *Default* Data Output)
PB0	PCINT8 (Pin Change Interrupt 8) OC1AU (Output Compare and PWM Output A-U for Timer/Counter1) DI (Three-wire Mode USI *Default* Data Input) SDA (Two-wire Mode USI *Default* Data Input / Output)

- During asynchronous operation, the synchronization of the interrupt flags for the asynchronous timer takes three processor cycles plus one timer cycle. The timer is therefore advanced by at least one before the processor can read the timer value causing the setting of the interrupt flag. The Output Compare pin is changed on the timer clock and is not synchronized to the processor clock.

5.11.10 Timer/Counter0 Prescaler

Figure 5-41. Prescaler for Timer/Counter0



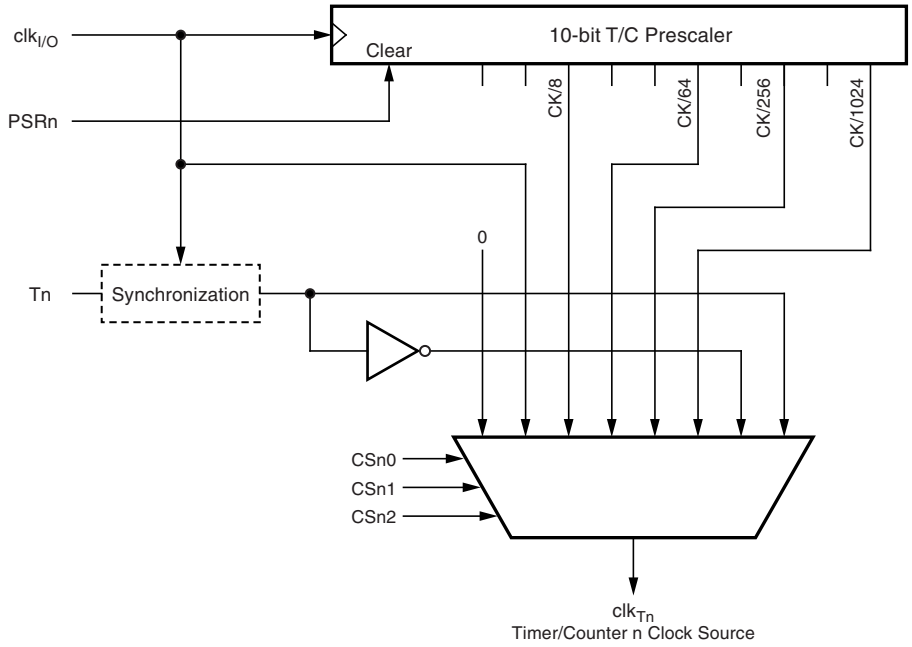
The clock source for Timer/Counter0 is named clk_{T0S}. clk_{T0S} is by default connected to the main system I/O clock clk_{IO}. By setting the AS0 bit in ASSR, Timer/Counter0 is asynchronously clocked from the XTAL oscillator or XTAL1 pin. This enables use of Timer/Counter0 as a Real Time Counter (RTC).

A crystal can then be connected between the XTAL1 and XTAL2 pins to serve as an independent clock source for Timer/Counter0.

A external clock can also be used using XTAL1 as input. Setting AS0 and EXCLK enables this configuration.

For Timer/Counter0, the possible prescaled selections are: $clk_{T0S}/8$, $clk_{T0S}/32$, $clk_{T0S}/64$, $clk_{T0S}/128$, $clk_{T0S}/256$, and $clk_{T0S}/1024$. Additionally, clk_{T0S} as well as 0 (stop) may be selected. Setting the PSR0 bit in GTCCR resets the prescaler. This allows the user to operate with a predictable prescaler.

Figure 5-43. Prescaler for Timer/Counter1[1]



Note: 1. The synchronization logic on the input pin (T1) is shown in Figure 5-42 on page 134.

5.12.2 Timer/Counter1 Prescalers Register Description

5.12.2.1 General Timer/Counter Control Register – GTCCR

Bit	7	6	5	4	3	2	1	0	
	TSM	–	–	–	–	–	PSR0	PSR1	GTCCR
Read/Write	R	R	R	R	R	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – TSM: Timer/Counter Synchronization Mode**

  Writing the TSM bit to one activates the Timer/Counter Synchronization mode. In this mode, the value that is written to the PSR0 and PSR1 bits is kept, hence keeping the corresponding prescaler reset signals asserted. This ensures that the corresponding Timer/Counters are halted and can be configured to the same value without the risk of one of them advancing during configuration. When the TSM bit is written to zero, the PSR0 and PSR1 bits are cleared by hardware, and the Timer/Counters start counting simultaneously.

- **Bit 0 – PSR1: Prescaler Reset Timer/Counter1**

  When this bit is one, Timer/Counter1 prescaler will be reset. This bit is normally cleared immediately by hardware, except if the TSM bit is set.

5.13 16-bit Timer/Counter1

The 16-bit Timer/Counter unit allows accurate program execution timing (event management), wave generation, and signal timing measurement. The main features are:

5.13.1 Features

- True 16-bit Design (i.e., Allows 16-bit PWM)
- Two independent Output Compare Units
- Four Controlled Output Pins per Output Compare Unit
- Double Buffered Output Compare Registers
- One Input Capture Unit
- Input Capture Noise Canceler
- Clear Timer on Compare Match (Auto Reload)
- Glitch-free, Phase Correct Pulse Width Modulator (PWM)
- Variable PWM Period
- Frequency Generator
- External Event Counter
- Four independent interrupt Sources (TOV1, OCF1A, OCF1B, and ICF1)

5.13.2 Overview

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 1. However, when using the register or bit defines in a program, the precise form must be used, i.e., TCNT1 for accessing Timer/Counter1 counter value and so on.
- A lower case “x” replaces the Output Compare unit channel, in this case A or B. However, when using the register or bit defines in a program, the precise form must be used, i.e., OCR1A for accessing Timer/Counter1 output compare channel A value and so on.
- A lower case “i” replaces the index of the Output Compare output pin, in this case U, V, W or X. However, when using the register or bit defines in a program, the precise form must be used.

A simplified block diagram of the 16-bit Timer/Counter is shown in Figure 5-44 on page 137. CPU accessible I/O Registers, including I/O bits and I/O pins, are shown in bold. The device-specific I/O Register and bit locations are listed in Section 5.13.11 “16-bit Timer/Counter Register Description” on page 156.

Figure 5-55 shows the same timing data, but with the prescaler enabled.

Figure 5-55. Timer/Counter Timing Diagram, Setting of OCF1A/B, with Prescaler ($f_{clk\_I/O}/8$)

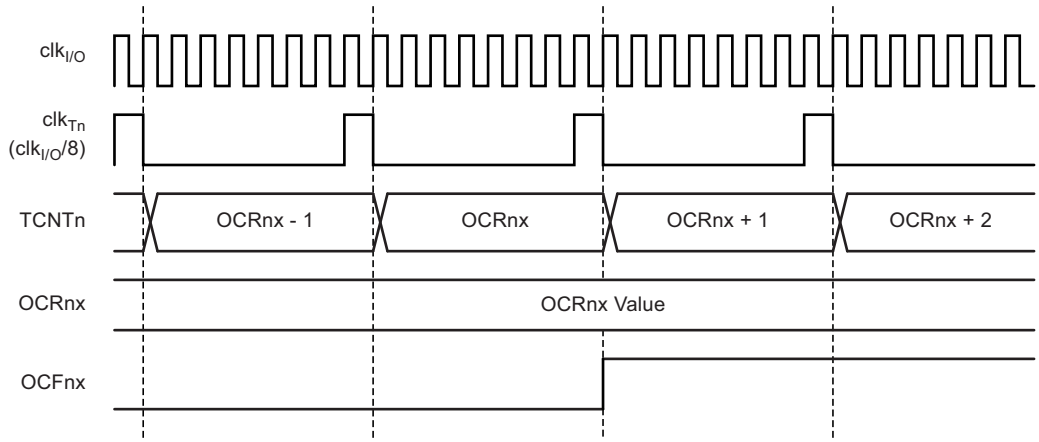


Figure 5-56 shows the count sequence close to TOP in various modes. When using phase and frequency correct PWM mode the OCR1A/B Register is updated at BOTTOM. The timing diagrams will be the same, but TOP should be replaced by BOTTOM, TOP-1 by BOTTOM+1 and so on. The same renaming applies for modes that set the TOV1 flag at BOTTOM.

Figure 5-56. Timer/Counter Timing Diagram, no Prescaling

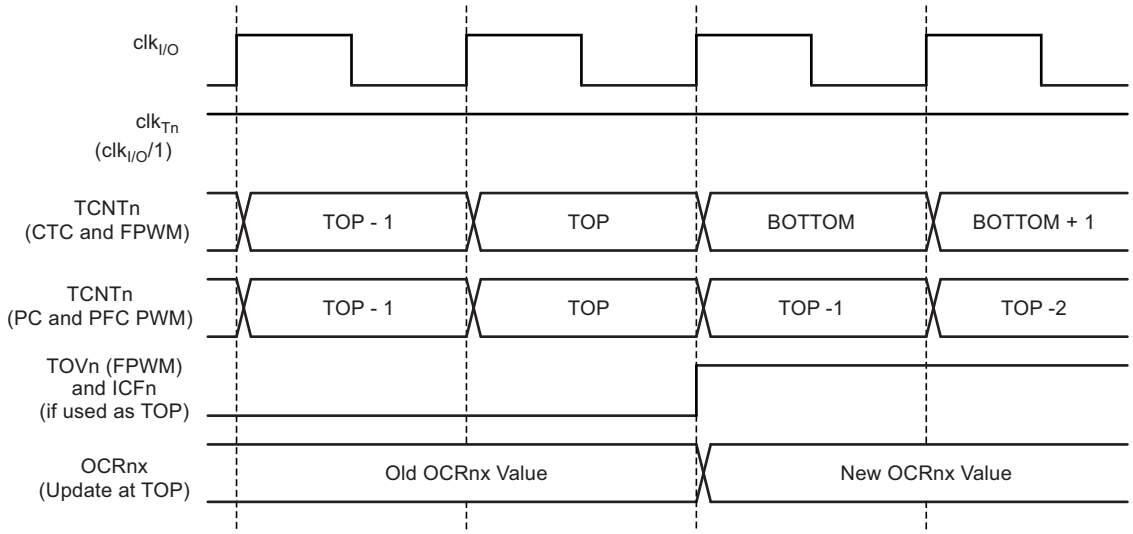


Table 5-37. Waveform Generation Mode Bit Description(1)

Mode	WGM13	WGM12 (CTC1)	WGM11 (PWM11)	WGM10 (PWM10)	Timer/Counter Mode of Operation	TOP	Update of OCR1A/B at	TOV1 Flag Set on
0	0	0	0	0	Normal	0xFFFF	Immediate	MAX
1	0	0	0	1	PWM, Phase Correct, 8-bit	0x00FF	TOP	BOTTOM
2	0	0	1	0	PWM, Phase Correct, 9-bit	0x01FF	TOP	BOTTOM
3	0	0	1	1	PWM, Phase Correct, 10-bit	0x03FF	TOP	BOTTOM
4	0	1	0	0	CTC	OCR1A	Immediate	MAX
5	0	1	0	1	Fast PWM, 8-bit	0x00FF	TOP	TOP
6	0	1	1	0	Fast PWM, 9-bit	0x01FF	TOP	TOP
7	0	1	1	1	Fast PWM, 10-bit	0x03FF	TOP	TOP
8	1	0	0	0	PWM, Phase and Frequency Correct	ICR1	BOTTOM	BOTTOM
9	1	0	0	1	PWM, Phase and Frequency Correct	OCR1A	BOTTOM	BOTTOM
10	1	0	1	0	PWM, Phase Correct	ICR1	TOP	BOTTOM
11	1	0	1	1	PWM, Phase Correct	OCR1A	TOP	BOTTOM
12	1	1	0	0	CTC	ICR1	Immediate	MAX
13	1	1	0	1	(Reserved)	–	–	–
14	1	1	1	0	Fast PWM	ICR1	TOP	TOP
15	1	1	1	1	Fast PWM	OCR1A	TOP	TOP

Note: 1. The CTC1 and PWM11:0 bit definition names are obsolete. Use the WGM12:0 definitions. However, the functionality and location of these bits are compatible with previous versions of the timer.

5.13.11.2 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.

5.18.12.6 DIDR1 – Digital Input Disable Register 1

Bit	7	6	5	4	3	2	1	0	
	-	ADC10D	ADC9D	ADC8D	-	-	-	-	DIDR1
Read/Write	R	R/W	R/W	R/W	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – Res: Reserved Bit**

  This bit is reserved for future use. For compatibility with future devices it must be written to zero when DIDR1 register is written.

- **Bits 6..4 – ADC10D..ADC8D: ADC10..8 Digital Input Disable**

  When this bit is written logic one, the digital input buffer on the corresponding ADC 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 ADC10:8 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.

- **Bits 3:0 – Reserved Bits**

  These bits are reserved for future use. For compatibility with future devices, they must be written to zero when DIDR1 is written.

5.18.12.7 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 5.18.11 on page 217. 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 5.18.11 on page 217. It is recommended to use DIDR register bit function (digital input disable) when XREFEN is set.

- **Bit 1 – PGERS: Page Erase**

  If this bit is written to one at the same time as SPMEN, the next SPM instruction within four clock cycles executes Page Erase. The page address is taken from the high part of the Zpointer. The data in R1 and R0 are ignored. The PGERS bit will auto-clear upon completion of a Page Erase, or if no SPM instruction is executed within four clock cycles. The CPU is halted during the entire Page Write operation.

- **Bit 0 – SPMEN: Self Programming Enable**

  This bit enables the SPM instruction for the next four clock cycles. If written to one together with either SIGRD, CTPB, RFLB, PGWRT, or PGERS, the following SPM instruction will have a special meaning, see description above. If only SPMEN is written, the following SPM instruction will store the value in R1:R0 in the temporary page buffer addressed by the Z-pointer. The LSB of the Z-pointer is ignored. The SPMEN bit will auto-clear upon completion of an SPM instruction, or if no SPM instruction is executed within four clock cycles. During Page Erase and Page Write, the SPMEN bit remains high until the operation is completed.

  Writing any other combination than “$10\ 0001_b$”, “$01\ 0001_b$”, “$00\ 1001_b$”, “$00\ 0101_b$”, “$00\ 0011_b$” or “$00\ 0001_b$” in the lower six bits will have no effect.

  Note: Only one SPM instruction should be active at any time.

5.21.2.2 EEPROM Write Prevents Writing to SPMCSR

Note that an EEPROM write operation will block all software programming to Flash. Reading the Fuses and Lock bits from software will also be prevented during the EEPROM write operation. It is recommended that the user checks the status bit (EEPE) in the EECR Register and verifies that the bit is cleared before writing to the SPMCSR Register.

5.21.2.3 Reading the Fuse and Lock Bits from Software

It is possible to read both the Fuse and Lock bits from software. To read the Lock bits, load the Z-pointer with 0x0001 and set the RFLB and SPMEN bits in SPMCSR. When an LPM instruction is executed within three CPU cycles after the RFLB and SPMEN bits are set in SPMCSR, the value of the Lock bits will be loaded in the destination register. The RFLB and SPMEN bits will auto-clear upon completion of reading the Lock bits or if no LPM instruction is executed within three CPU cycles or no SPM instruction is executed within four CPU cycles. When RFLB and SPMEN are cleared, LPM will work as described in the Instruction set Manual.

Bit	7	6	5	4	3	2	1	0
Rd (Z=0x0001)	–	–	–	–	–	–	LB2	LB1

The algorithm for reading the Fuse Low byte is similar to the one described above for reading the Lock bits. To read the Fuse Low byte, load the Z-pointer with 0x0000 and set the RFLB and SPMEN bits in SPMCSR. When an LPM instruction is executed within three cycles after the RFLB and SPMEN bits are set in the SPMCSR, the value of the Fuse Low byte (FLB) will be loaded in the destination register as shown below. See Table 5-70 on page 237 for a detailed description and mapping of the Fuse Low byte.

Bit	7	6	5	4	3	2	1	0
Rd (Z=0x0000)	FLB7	FLB6	FLB5	FLB4	FLB3	FLB2	FLB1	FLB0

Similarly, when reading the Fuse High byte (FHB), load 0x0003 in the Z-pointer. When an LPM instruction is executed within three cycles after the RFLB and SPMEN bits are set in the SPMCSR, the value of the Fuse High byte will be loaded in the destination register as shown below. See Table 5-69 on page 236 for detailed description and mapping of the Fuse High byte.

Bit	7	6	5	4	3	2	1	0
Rd (Z=0x0003)	FHB7	FHB6	FHB5	FHB4	FHB3	FHB2	FHB1	FHB0

Similarly, when reading the Extended Fuse byte (EFB), load 0x0002 in the Z-pointer. When an LPM instruction is executed within three cycles after the RFLB and SPMEN bits are set in the SPMCSR, the value of the Extended Fuse byte will be loaded in the destination register as shown below. See Table 5-68 on page 236 for detailed description and mapping of the Extended Fuse byte.

Bit	7	6	5	4	3	2	1	0
Rd (Z=0x0002)	–	–	–	–	–	–	–	EFB0

Fuse and Lock bits that are programmed, will be read as zero. Fuse and Lock bits that are unprogrammed, will be read as one.

5.22 Memory Programming

5.22.1 Program and Data Memory Lock Bits

The Atmel® ATtiny87/167 provides two Lock bits which can be left unprogrammed (“1”) or can be programmed (“0”) to obtain the additional features listed in Table 5-67. The Lock bits can only be erased to “1” with the Chip Erase command. The Atmel ATtiny87/167 has no separate Boot Loader section.

Table 5-66. Lock Bit Byte(1)

Lock Bit Byte	Bit No	Description	Default Value
	7	–	1 (unprogrammed)
	6	–	1 (unprogrammed)
	5	–	1 (unprogrammed)
	4	–	1 (unprogrammed)
	3	–	1 (unprogrammed)
	2	–	1 (unprogrammed)
LB2	1	Lock bit	1 (unprogrammed)
LB1	0	Lock bit	1 (unprogrammed)

Note: 1. “1” means unprogrammed, “0” means programmed.

Table 5-67. Lock Bit Protection Modes(1)(2)

Memory Lock Bits			Protection Type
LB Mode	LB2	LB1	
1	1	1	No memory lock features enabled.
2	1	0	Further programming of the Flash and EEPROM is disabled in Parallel and Serial Programming mode. The Fuse bits are locked in both Serial and Parallel Programming mode(1).
3	0	0	Further programming and verification of the Flash and EEPROM is disabled in Parallel and Serial Programming mode. The Fuse bits are locked in both Serial and Parallel Programming mode(1).

Notes: 1. Program the Fuse bits before programming the LB1 and LB2.
2. “1” means unprogrammed, “0” means programmed.

5.23.2 DC Characteristics (Continued)

T_A = –40°C to +125°C, Vcc = 2.7V to 5.5V (unless otherwise noted)

Parameters	Test Conditions	Symbol	Min.	Typ.(1)	Max.	Unit
Input Leakage Current I/O Pin	Vcc = 5.5V, pin high (absolute value)	I_{IH}		< 0.05	1	µA
Reset Pull-up Resistor		R_{RST}	30		60	kΩ
I/O Pin Pull-up Resistor		R_{pu}	20		50	kΩ
Power Supply Current(6) Active Mode (external clock)	16MHz, Vcc = 5V	I_{CC}		10	13	mA
	8MHz, Vcc = 5V			5.5	7.0	mA
	8MHz, Vcc = 3V			2.8	3.5	mA
	4MHz, Vcc = 3V			1.8	2.5	mA
Power Supply Current(6) Idle Mode (external clock)	16MHz, Vcc = 5V			3.5	5.0	mA
	8MHz, Vcc = 5V			1.8	2.5	mA
	8MHz, Vcc = 3V			1	1.5	mA
	4MHz, Vcc = 3V			0.5	0.8	mA
Power Supply Current(7) Power-down Mode	WDT enabled, Vcc = 5V			7	100	µA
	WDT disabled, Vcc = 5V			0.18	70	µA
	WDT enabled, Vcc = 3V			5	70	µA
	WDT disabled, Vcc = 3V			0.15	45	µA
Analog Comparator Input Offset Voltage	Vcc = 5V V_{in} = Vcc/2	V_{ACIO}	–10	+10	+40	mV
Analog Comparator Input Leakage Current	Vcc = 5V V_{in} = Vcc/2	I_{ACLK}	–50		+50	nA
Analog Comparator Propagation Delay Common Mode Vcc/2	Vcc = 2.7V	t_{ACID}		170		ns
	Vcc = 5.0V			180		ns

Notes:
1. “Typ.”, typical values at 25°C. Maximum values are characterized values and not test limits in production.
2. “Max.” means the highest value where the pin is guaranteed to be read as low.
3. “Min.” means the lowest value where the pin is guaranteed to be read as high.
4. Although each I/O port can sink more than the test conditions (10mA at Vcc = 5V, 5mA at Vcc = 3V) under steady state conditions (non-transient), the following must be observed:
   The sum of all IOL, for all ports, should not exceed 120mA.
   If IOL exceeds the test condition, VOL may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test condition.
5. Although each I/O port can source more than the test conditions (10mA at Vcc = 5V, 5mA at Vcc = 3V) under steady state conditions (non-transient), the following must be observed:
   The sum of all IOH, for all ports, should not exceed 120mA.
   If IOH exceeds the test condition, VOH may exceed the related specification. Pins are not guaranteed to source current greater than the listed test condition.
6. Values using methods described in Section 5.6.8 “Minimizing Power Consumption” on page 79. Power Reduction is enabled (PRR = 0xFF) and there is no I/O drive.
7. BOD disabled.

Figure 5-112. SPI Interface Timing Requirements (Master Mode)

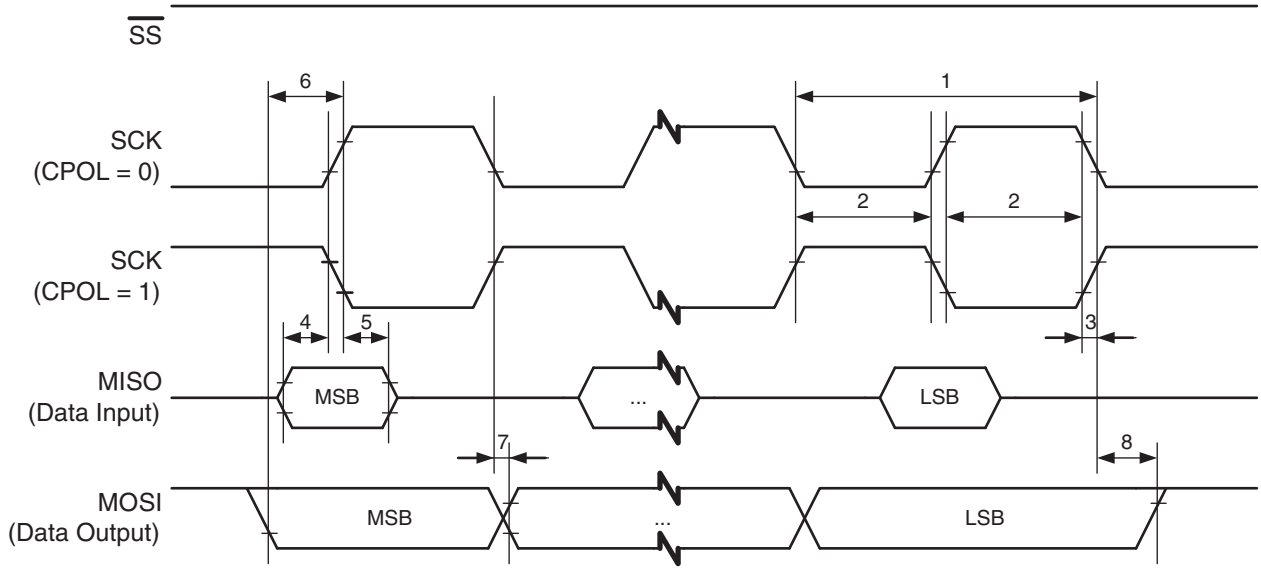
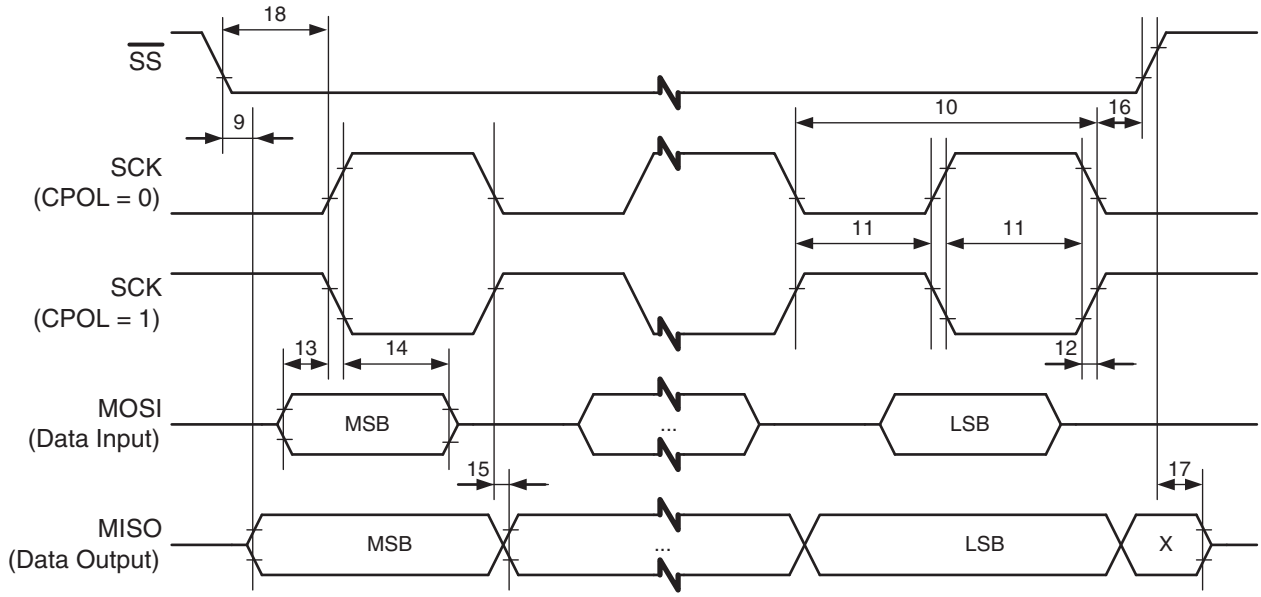


Figure 5-113. SPI Interface Timing Requirements (Slave Mode)



5.26 Register Summary

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
(0xFF)	Reserved									
(0xFE)	Reserved									
(0xFD)	Reserved									
(0xFC)	Reserved									
(0xFB)	Reserved									
(0xFA)	Reserved									
(0xF9)	Reserved									
(0xF8)	Reserved									
(0xF7)	Reserved									
(0xF6)	Reserved									
(0xF5)	Reserved									
(0xF4)	Reserved									
(0xF3)	Reserved									
(0xF2)	Reserved									
(0xF1)	Reserved									
(0xF0)	Reserved									
(0xEF)	Reserved									
(0xEE)	Reserved									
(0xED)	Reserved									
(0xEC)	Reserved									
(0xEB)	Reserved									
(0xEA)	Reserved									
(0xE9)	Reserved									
(0xE8)	Reserved									
(0xE7)	Reserved									
(0xE6)	Reserved									
(0xE5)	Reserved									
(0xE4)	Reserved									
(0xE3)	Reserved									
(0xE2)	Reserved									
(0xE1)	Reserved									
(0xE0)	Reserved									
(0xDF)	Reserved									
(0xDE)	Reserved									
(0xDD)	Reserved									
(0xDC)	Reserved									
(0xDB)	Reserved									
(0xDA)	Reserved									
(0xD9)	Reserved									
(0xD8)	Reserved									
(0xD7)	Reserved									
(0xD6)	Reserved									
(0xD5)	Reserved									
(0xD4)	Reserved									
(0xD3)	Reserved									