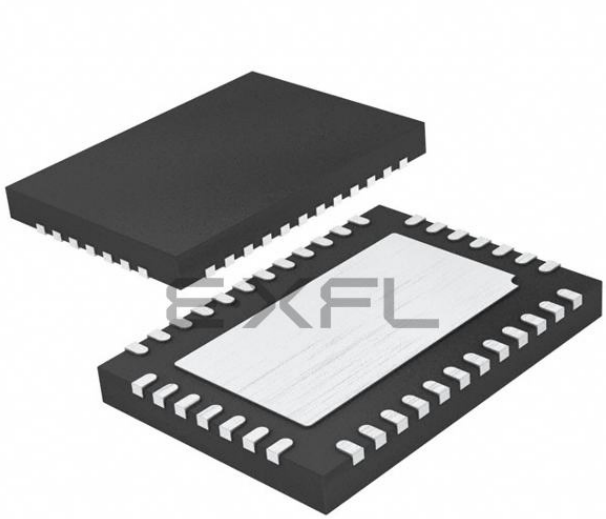


Welcome to E-XFL.COM

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	8KB (8K 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	38-VFQFN Exposed Pad
Supplier Device Package	38-VQFN (5x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/ata6616-p3pw

Figure 3-7. VCC Voltage Regulator: Ramp-up and Undervoltage Detection

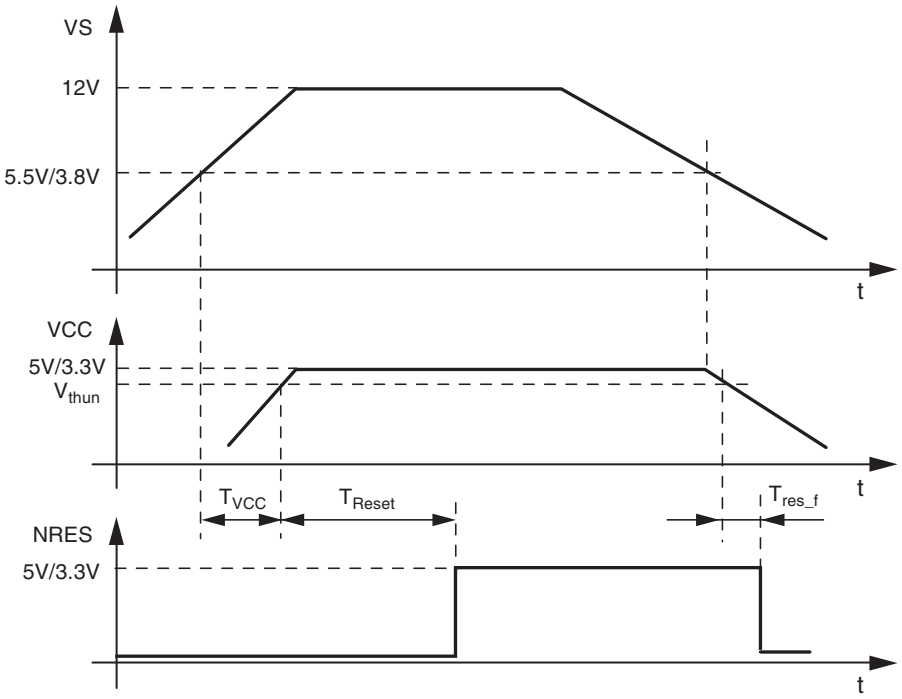
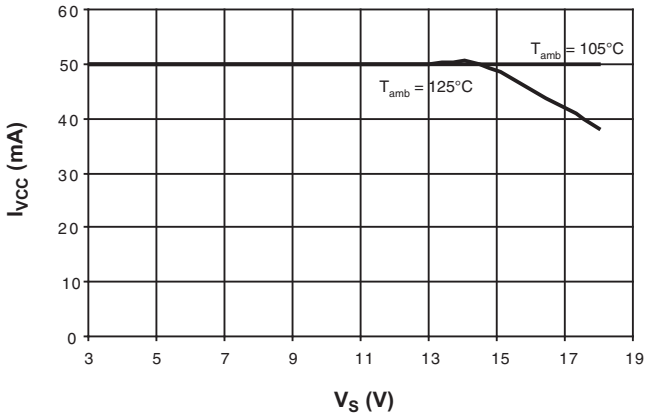


Figure 3-8. Power Dissipation: Safe Operating Area versus VCC Output Current and Supply Voltage V_S at Different Ambient Temperatures Due to R_{thja} = 25K/W



For programming purposes of the microcontroller it is potentially necessary to supply the V_{CC} output via an external power supply while the V_S Pin of the system basis chip is disconnected. This behavior is no problem for the system basis chip.

3.7 Absolute Maximum Ratings

Stresses beyond those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Parameters	Symbol	Min.	Typ.	Max.	Unit
Supply voltage V_S	V_S	–0.3		+40	V
Pulse time ≤ 500ms; T_a = 25°C Output current I_{VCC} ≤ 50mA	V_S			+40	V
Pulse time ≤ 2min; T_a = 25°C Output current I_{VCC} ≤ 50mA	V_S			27	V
WAKE (with 33kΩ serial resistor) KL_15 (with 50kΩ/100nF) DC voltage Transient voltage due to ISO7637 (coupling 1nF)		 –1 –150		 +40 +100	 V V
INH - DC voltage		 –0.3		 V_S + 0.3	 V
LIN - DC voltage		 –27		 +40	 V
Logic pins (RxD, TxD, EN, NRES, NTRIG, WD_OSC, MODE, TM)		–0.3		+5.5	V
Output current NRES	I_{NRES}			+2	mA
PVCC DC voltage VCC DC voltage		–0.3 –0.3		+5.5 +6.5	V V
ESD according to IBEE LIN EMC Test Spec. 1.0 following IEC 61000-4-2 - Pin VS, LIN to GND - Pin WAKE (33kΩ serial resistor) to GND		 ±6 ±5			 KV KV
ESD HBM following STM5.1 with 1.5kΩ 100pF - Pin VS, LIN, WAKE to GND		±8			KV
Junction temperature	T_j	–40		+150	°C
Storage temperature	T_s	–55		+150	°C

4.3.4.1 *The X-register, Y-register, and Z-register*

The registers R26..R31 have some added functions to their general purpose usage. These registers are 16-bit address pointers for indirect addressing of the data space. The three indirect address registers X, Y, and Z are defined as described in Figure 4-4 on page 35.

Figure 4-4. The X-, Y-, and Z-registers

In the different addressing modes these address registers have functions as fixed displacement, automatic increment, and automatic decrement (see the instruction set reference for details).

4.3.5 Stack Pointer

The Stack is mainly used for storing temporary data, for storing local variables and for storing return addresses after interrupts and subroutine calls. The Stack Pointer Register always points to the top of the Stack. Note that the Stack is implemented as growing from higher memory locations to lower memory locations. This implies that a Stack PUSH command decreases the Stack Pointer.

The Stack Pointer points to the data SRAM Stack area where the Subroutine and Interrupt Stacks are located. This Stack space in the data SRAM must be defined by the program before any subroutine calls are executed or interrupts are enabled. The Stack Pointer must be set to point above 0x60. The Stack Pointer is decremented by one when data is pushed onto the Stack with the PUSH instruction, and it is decremented by two when the return address is pushed onto the Stack with subroutine call or interrupt. The Stack Pointer is incremented by one when data is popped from the Stack with the POP instruction, and it is incremented by two when data is popped from the Stack with return from subroutine RET or return from interrupt RETI.

The AVR® Stack Pointer is implemented as two 8-bit registers in the I/O space. The number of bits actually used is implementation dependent. Note that the data space in some implementations of the AVR architecture is so small that only SPL is needed. In this case, the SPH Register will not be present

Notes: 1. ONLY the reset (watchdog reset included) disables this function. The Watchdog System Reset Flag (WDRF bit of MCUSR register) can be used to monitor the reset cause.

2. ONLY clock frequencies ≥ (4 * WatchDog Clock frequency) can be monitored.

Here is a “light” C-code of a clock switching function using automatic clock monitoring.

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
#define WD_ARL_ENABLE   0x06

#define WD_2048CYCLES   0x07

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);

  // Enable the watchdog in automatic reload mode
    WDTCSR = (1 << WDCE) | (1 << WDE);
    WDTCSR = (1 << WDE ) | WD_2048CYCLES;
    CLKCSR = 1 << CLKCCE;
    CLKCSR = WD_ARL_ENABLE;

  // 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;
}
```

4.5.4 System Clock Prescaler

4.5.4.1 Features

The Atmel® ATtiny87/167 system clock can be divided by setting the Clock Prescaler Register – CLKPR. This feature can be used to decrease power consumption when the requirement for processing power is low. This can be used with all clock source options, and it will affect the clock frequency of the CPU and all synchronous peripherals. $clk_{I/O}$, clk_{ADC}, clk_{CPU}, and clk_{FLASH} are divided by a factor as shown in Table 4-14 on page 63.

4.5.4.2 Switching Time

When switching between prescaler settings, the System Clock Prescaler ensures that no glitches occur in the clock system and that no intermediate frequency is higher than neither the clock frequency corresponding to the previous setting, nor the clock frequency corresponding to the new setting.

The ripple counter that implements the prescaler runs at the frequency of the undivided clock, which may be faster than the CPU's clock frequency. Hence, it is not possible to determine the state of the prescaler – even if it were readable, and the exact time it takes to switch from one clock division to another cannot be exactly predicted.

From the time the CLKPS values are written, it takes between T1 + T2 and T1 + 2*T2 before the new clock frequency is active. In this interval, 2 active clock edges are produced. Here, T1 is the previous clock period, and T2 is the period corresponding to the new prescaler setting.

4.5.5 Register Description

4.5.5.1 OSCCAL – Oscillator Calibration Register

Bit	7	6	5	4	3	2	1	0	
	CAL7	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	OSCCAL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	Device Specific Calibration Value								

- **Bits 7:0 – CAL7:0: Oscillator Calibration Value**

The Oscillator Calibration Register is used to trim the Calibrated Internal RC Oscillator to remove process variations from the oscillator frequency. The factory-calibrated value is automatically written to this register during chip reset, giving an oscillator frequency of 8.0 MHz at 25°C. The application software can write this register to change the oscillator frequency. The oscillator can be calibrated to any frequency in the range 7.3 - 8.1MHz within ±2% accuracy. Calibration outside that range is not guaranteed.

Note that this oscillator is used to time EEPROM and Flash write accesses, and these write times will be affected accordingly. If the EEPROM or Flash are written, do not calibrate to more than 8.8MHz. Otherwise, the EEPROM or Flash write may fail.

The CAL7 bit determines the range of operation for the oscillator. Setting this bit to 0 gives the lowest frequency range, setting this bit to 1 gives the highest frequency range. The two frequency ranges are overlapping, in other words a setting of OSCCAL = 0x7F gives a higher frequency than OSCCAL = 0x80.

4.7.3.3 *Watchdog Timer Control Register - WDTCR*

Bit	7	6	5	4	3	2	1	0	
	WDIF	**WDIE**	**WDP3**	**WDCE**	**WDE**	**WDP2**	**WDP1**	**WDP0**	**WDTCR**
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	X	0	0	0	

- **Bit 7 - WDIF: Watchdog Interrupt Flag**

This bit is set when a time-out occurs in the Watchdog Timer and the Watchdog Timer is configured for interrupt. WDIF is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, WDIF is cleared by writing a logic one to the flag. When the I-bit in SREG and WDIE are set, the Watchdog Time-out Interrupt is executed.

- **Bit 6 - WDIE: Watchdog Interrupt Enable**

When this bit is written to one and the I-bit in the Status Register is set, the Watchdog Interrupt is enabled. If WDE is cleared in combination with this setting, the Watchdog Timer is in Interrupt Mode, and the corresponding interrupt is executed if time-out in the Watchdog Timer occurs.

If WDE is set, the Watchdog Timer is in Interrupt and System Reset Mode. The first time-out in the Watchdog Timer will set WDIF. Executing the corresponding interrupt vector will clear WDIE and WDIF automatically by hardware (the Watchdog goes to System Reset Mode). This is useful for keeping the Watchdog Timer security while using the interrupt. To stay in Interrupt and System Reset Mode, WDIE must be set after each interrupt. This should however not be done within the interrupt service routine itself, as this might compromise the safety-function of the Watchdog System Reset mode. If the interrupt is not executed before the next time-out, a System Reset will be applied.

If the Watchdog Timer is used as clock monitor (c.f. Section • “Bits 3:0 – CLKC3:0: Clock Control Bits 3 - 0” on page 64), the System Reset Mode is enabled and the Interrupt Mode is automatically disabled.

Table 4-18. Watchdog Timer Configuration

Clock Monitor	WDTON	WDE	WDIE	Mode	Action on Time-out
x	0	0	0	Stopped	None
On	y(1)	y(1)	y(1)	System Reset Mode	Reset
Off	0	0	1	Interrupt Mode	Interrupt
	0	1	0	System Reset Mode	Reset
	0	1	1	Interrupt and System Reset Mode	Interrupt, then go to System Reset Mode
	1	x	x	System Reset Mode	Reset

Note: 1. At least one of these three enables (WDTON, WDE & WDIE) equal to 1

- **Bit 4 - WDCE: Watchdog Change Enable**

This bit is used in timed sequences for changing WDE and prescaler bits. To clear the WDE bit, and/or change the prescaler bits, WDCE must be set.

Once written to one, hardware will clear WDCE after four clock cycles.

4.10.4 Register Description for I/O Ports

4.10.4.1 Port A Data Register – PORTA

Bit	7	6	5	4	3	2	1	0	
	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	PORTA
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	

4.10.4.2 Port A Data Direction Register – DDRA

Bit	7	6	5	4	3	2	1	0	
	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	DDRA
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	

4.10.4.3 Port A Input Pins Register – PINA

Bit	7	6	5	4	3	2	1	0	
	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0	PINA
Read/Write	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	
Initial Value	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	

4.10.4.4 Port B Data Register – PORTB

Bit	7	6	5	4	3	2	1	0	
	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	PORTB
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	

4.10.4.5 Port B Data Direction Register – DDRB

Bit	7	6	5	4	3	2	1	0	
	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	DDRB
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	

4.10.4.6 Port B Input Pins Register – PINB

Bit	7	6	5	4	3	2	1	0	
	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	PINB
Read/Write	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	
Initial Value	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	

The OCR0A Register is double buffered when using any of the Pulse Width Modulation (PWM) modes. For the Normal and Clear Timer on Compare (CTC) modes of operation, the double buffering is disabled. The double buffering synchronizes the update of the OCR0A Compare Register to either top or bottom of the counting sequence. The synchronization prevents the occurrence of odd-length, non-symmetrical PWM pulses, thereby making the output glitch-free.

The OCR0A Register access may seem complex, but this is not case. When the double buffering is enabled, the CPU has access to the OCR0A Buffer Register, and if double buffering is disabled the CPU will access the OCR0A directly.

4.11.5.1 Force Output Compare

In non-PWM waveform generation modes, the match output of the comparator can be forced by writing a one to the Force Output Compare (FOC0A) bit. Forcing compare match will not set the OCF0A flag or reload/clear the timer, but the OC0A pin will be updated as if a real compare match had occurred (the COM0A1:0 bits settings define whether the OC0A pin is set, cleared or toggled).

4.11.5.2 Compare Match Blocking by TCNT0 Write

All CPU write operations to the TCNT0 Register will block any compare match that occurs in the next timer clock cycle, even when the timer is stopped. This feature allows OCR0A to be initialized to the same value as TCNT0 without triggering an interrupt when the Timer/Counter clock is enabled.

4.11.5.3 Using the Output Compare Unit

Since writing TCNT0 in any mode of operation will block all compare matches for one timer clock cycle, there are risks involved when changing TCNT0 when using the Output Compare channel, independently of whether the Timer/Counter is running or not. If the value written to TCNT0 equals the OCR0A value, the compare match will be missed, resulting in incorrect waveform generation. Similarly, do not write the TCNT0 value equal to BOTTOM when the counter is downcounting.

The setup of the OC0A should be performed before setting the Data Direction Register for the port pin to output. The easiest way of setting the OC0A value is to use the Force Output Compare (FOC0A) strobe bit in Normal mode. The OC0A Register keeps its value even when changing between Waveform Generation modes.

Be aware that the COM0A1:0 bits are not double buffered together with the compare value. Changing the COM0A1:0 bits will take effect immediately.

4.11.6 Compare Match Output Unit

The Compare Output mode (COM0A1:0) bits have two functions. The Waveform Generator uses the COM0A1:0 bits for defining the Output Compare (OC0A) state at the next compare match. Also, the COM0A1:0 bits control the OC0A pin output source. Figure 4-33 shows a simplified schematic of the logic affected by the COM0A1:0 bit setting. The I/O Registers, I/O bits, and I/O pins in the figure are shown in bold. Only the parts of the general I/O port control registers (DDR and PORT) that are affected by the COM0A1:0 bits are shown. When referring to the OC0A state, the reference is for the internal OC0A Register, not the OC0A pin.

4.13.7.1 *Force Output Compare*

In non-PWM Waveform Generation modes, the match output of the comparator can be forced by writing a one to the Force Output Compare (FOC1A/B) bit. Forcing compare match will not set the OCF1A/B flag or reload/clear the timer, but the OC1A/Bi pins will be updated as if a real compare match had occurred (the COM1A/B1:0 bits settings define whether the OC1A/Bi pins are set, cleared or toggled - if the respective OCnxi bit is set).

4.13.7.2 *Compare Match Blocking by TCNT1 Write*

All CPU writes to the TCNT1 Register will block any compare match that occurs in the next timer clock cycle, even when the timer is stopped. This feature allows OCR1A/B to be initialized to the same value as TCNT1 without triggering an interrupt when the Timer/Counter clock is enabled.

4.13.7.3 *Using the Output Compare Unit*

Since writing TCNT1 in any mode of operation will block all compare matches for one timer clock cycle, there are risks involved when changing TCNT1 when using any of the Output Compare channels, independent of whether the Timer/Counter is running or not. If the value written to TCNT1 equals the OCR1A/B value, the compare match will be missed, resulting in incorrect waveform generation. Do not write the TCNT1 equal to TOP in PWM modes with variable TOP values. The compare match for the TOP will be ignored and the counter will continue to 0xFFFF. Similarly, do not write the TCNT1 value equal to BOTTOM when the counter is downcounting.

The setup of the OC1A/B should be performed before setting the Data Direction Register for the port pin to output. The easiest way of setting the OC1A/B value is to use the Force Output Compare (FOC1A/B) strobe bits in Normal mode. The OC1A/B Register keeps its value even when changing between Waveform Generation modes.

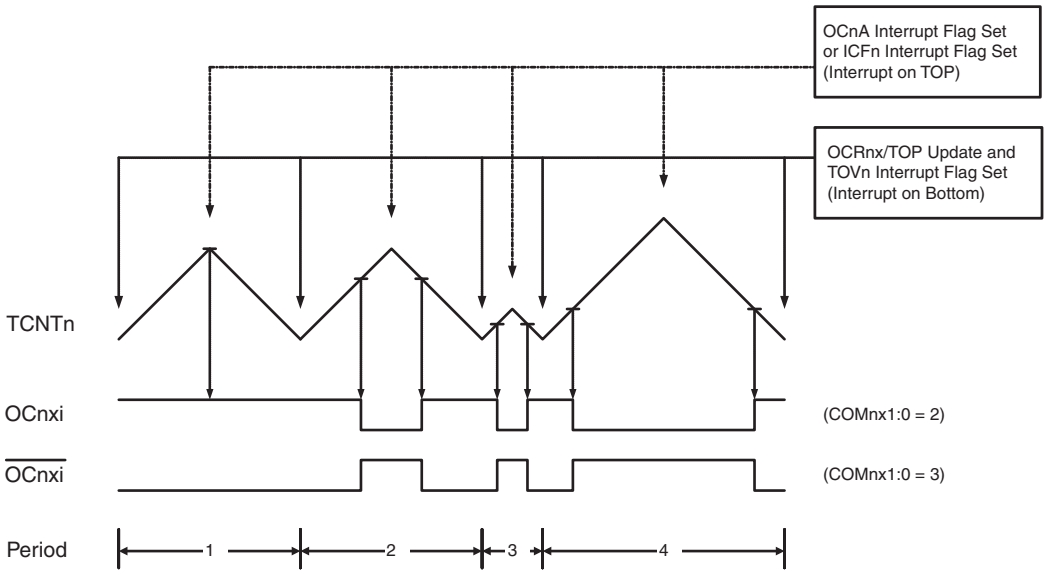
Be aware that the COM1A/B1:0 bits are not double buffered together with the compare value. Changing the COM1A/B1:0 bits will take effect immediately.

4.13.8 Compare Match Output Unit

The Compare Output mode (COM1A/B1:0) bits have two functions. The Waveform Generator uses the COM1A/B1:0 bits for defining the Output Compare (OC1A/B) state at the next compare match. Secondly the COM1A/B1:0 and OCnxi bits control the OC1A/Bi pin output source. Figure 4-49 shows a simplified schematic of the logic affected by the COM1A/B1:0 and OCnxi bit setting. The I/O Registers, I/O bits, and I/O pins in the figure are shown in bold. Only the parts of the general I/O port control registers (DDR and PORT) that are affected by the COM1A/B1:0 and OCnxi bits are shown. When referring to the OC1A/B state, the reference is for the internal OC1A/B Register, not the OC1A/Bi pin. If a system reset occur, the OC1A/B Register is reset to “0”.

In phase and frequency correct PWM mode the counter is incremented until the counter value matches either the value in ICR1 (WGM13:0 = 8), or the value in OCR1A (WGM13:0 = 9). The counter has then reached the TOP and changes the count direction. The TCNT1 value will be equal to TOP for one timer clock cycle. The timing diagram for the phase correct and frequency correct PWM mode is shown on Figure 4-53. The figure shows phase and frequency correct 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 dual-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 4-53. Phase and Frequency Correct PWM Mode, Timing Diagram



The Timer/Counter Overflow Flag (TOV1) is set at the same timer clock cycle as the OCR1A/B Registers are updated with the double buffer value (at BOTTOM). When either OCR1A or ICR1 is used for defining the TOP value, the OC1A or ICF1 flag set when TCNT1 has reached TOP. The interrupt flags can then be used to generate an interrupt each time the counter reaches the TOP or BOTTOM value.

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.

As Figure 4-53 shows the output generated is, in contrast to the phase correct mode, symmetrical in all periods. Since the OCR1A/B Registers are updated at BOTTOM, the length of the rising and the falling slopes will always be equal. This gives symmetrical output pulses and is therefore frequency correct.

Using the ICR1 Register for defining TOP works well when using fixed TOP values. By using ICR1, the OCR1A Register is free to be used for generating a PWM output on OC1A. However, if the base PWM frequency is actively changed by changing the TOP value, using the OCR1A as TOP is clearly a better choice due to its double buffer feature.

Table 4-44. CPOL Functionality

	Leading Edge	Trailing Edge	SPI Mode
CPOL=0, CPHA=0	Sample (Rising)	Setup (Falling)	0
CPOL=0, CPHA=1	Setup (Rising)	Sample (Falling)	1
CPOL=1, CPHA=0	Sample (Falling)	Setup (Rising)	2
CPOL=1, CPHA=1	Setup (Falling)	Sample (Rising)	3

Figure 4-60. SPI Transfer Format with CPHA = 0

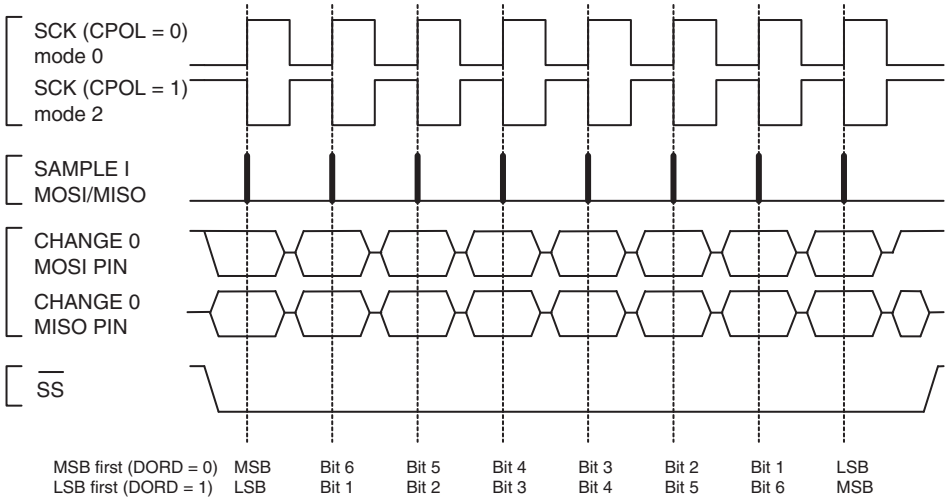
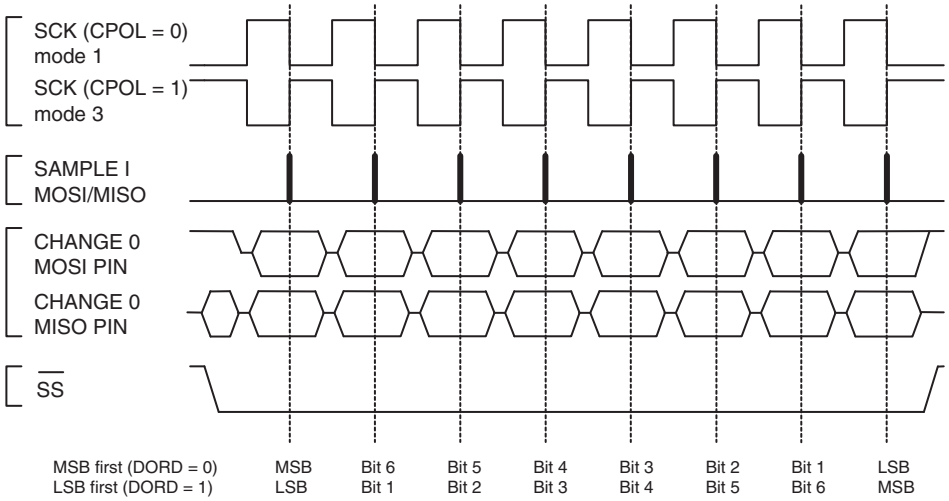


Figure 4-61. SPI Transfer Format with CPHA = 1



4.15 USI – Universal Serial Interface

4.15.1 Features

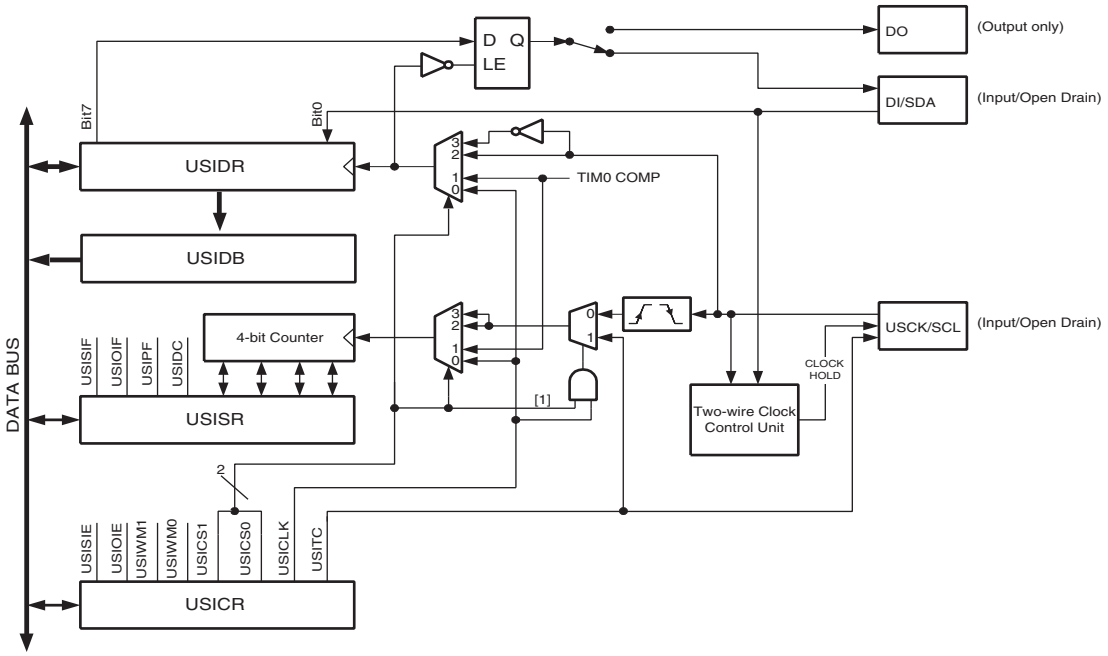
- **Two-wire Synchronous Data Transfer (Master or Slave)**
- **Three-wire Synchronous Data Transfer (Master or Slave)**
- **Data Received Interrupt**
- **Wakeup from Idle Mode**
- **In Two-wire Mode: Wake-up from All Sleep Modes, Including Power-down Mode**
- **Two-wire Start Condition Detector with Interrupt Capability**

4.15.2 Overview

The Universal Serial Interface, or USI, provides the basic hardware resources needed for serial communication. Combined with a minimum of control software, the USI allows significantly higher transfer rates and uses less code space than solutions based on software only. Interrupts are included to minimize the processor load.

A simplified block diagram of the USI is shown on Figure 4-62. 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 the “Register Descriptions” on page 181.

Figure 4-62. Universal Serial Interface, Block Diagram



The 8-bit USI Data Register is directly accessible via the data bus and contains the incoming and outgoing data. The register has no buffering so the data must be read as quickly as possible to ensure that no data is lost. The USI Data Register is a serial shift register and the most significant bit that is the output of the serial shift register is connected to one of two output pins depending of the wire mode configuration. A transparent latch is inserted between the USI Data Register Output and output pin, which delays the change of data output to the opposite clock edge of the data input sampling. The serial input is always sampled from the Data Input (DI) pin independent of the configuration.

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.

4.15.3 Functional Descriptions

4.15.3.1 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 4-63. Three-wire Mode Operation, Simplified Diagram

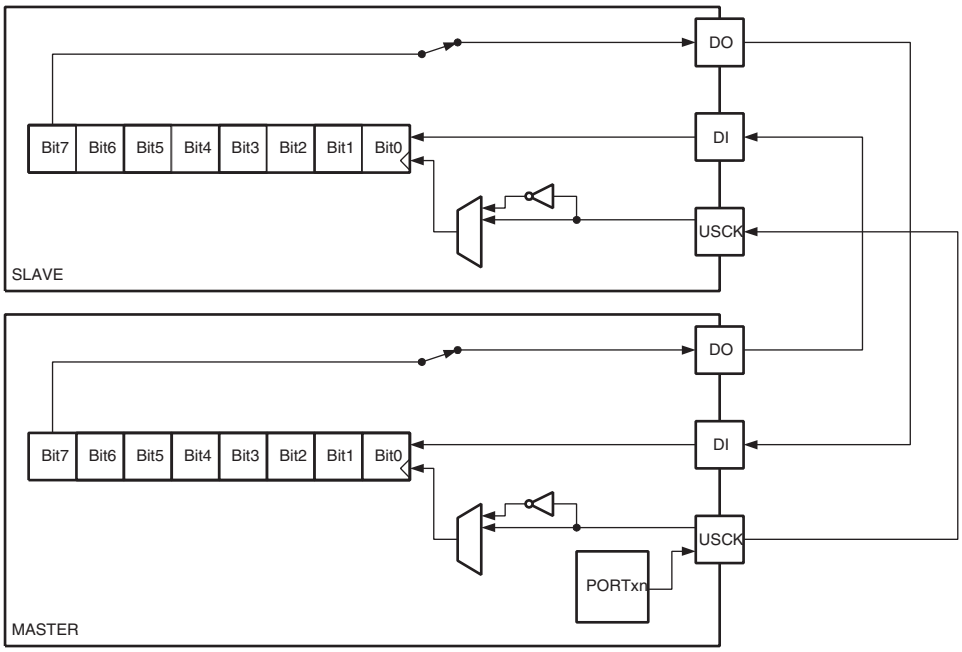


Figure 4-63 shows two USI units operating in Three-wire mode, one as Master and one as Slave. The two USI Data Register are interconnected in such way that after eight USCK clocks, the data in each register are interchanged. The same clock also increments the USI's 4-bit counter. The Counter Overflow (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.

Table 4-53. Example of Resistor Values(±5%) for a 8-address System (AVcc = 5V[1])

Physical Address	Resistor Value R_{load} (Ohm)	Typical Measured Voltage (V)	Minimum Reading with a 2.56V ref	Typical Reading with a 2.56V ref	Maximum Reading with a 2.56V ref
0	1 000	0.1		40	
1	2 200	0.22		88	
2	3 300	0.33		132	
3	4 700	0.47		188	
4	6 800	0.68		272	
5	10 000	1		400	
6	15 000	1.5		600	
7	22 000	2.2		880	

Note: 1. 5V range: Max R_{load} 30KΩ
3V range: Max R_{load} 15KΩ

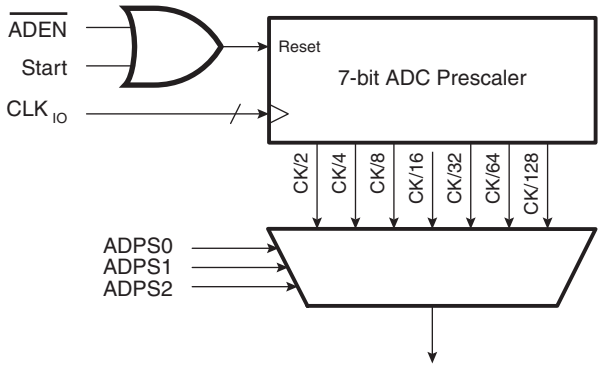
Table 4-54. Example of Resistor Values(±1%) for a 16-address System (AVcc = 5V[1])

Physical Address	Resistor Value R_{load} (Ohm)	Typical Measured Voltage (V)	Minimum Reading with a 2.56V ref	Typical Reading with a 2.56V ref	Maximum Reading with a 2.56V ref
0	1 000	0.1	38	40	45
1	1 200	0.12	46	48	54
2	1500	0.15	57	60	68
3	1800	0.18	69	72	81
4	2200	0.22	84	88	99
5	2700	0.27	104	108	122
6	3300	0.33	127	132	149
7	4700	0.47	181	188	212
8	6 800	0.68	262	272	306
9	8 200	0.82	316	328	369
10	10 000	1.0	386	400	450
11	12 000	1.2	463	480	540
12	15 000	1.5	579	600	675
13	18 000	1.8	694	720	810
14	22 000	2.2	849	880	989
15	27 000	2.7	1023	1023	1023

Note: 1. 5V range: Max R_{load} 30KΩ
3V range: Max R_{load} 15KΩ

4.18.5 Prescaling and Conversion Timing

Figure 4-84. ADC Prescaler



By default, the successive approximation circuitry requires an input clock frequency between 50kHz and 200kHz to get maximum resolution. If a lower resolution than 10bits is needed, the input clock frequency to the ADC can be higher than 200kHz to get a higher sample rate.

The ADC module contains a prescaler, which generates an acceptable ADC clock frequency from any CPU frequency above 100kHz. The prescaling is set by the ADPS bits in ADCSRA register. The prescaler starts counting from the moment the ADC is switched on by setting the ADEN bit in ADCSRA register. The prescaler keeps running for as long as the ADEN bit is set, and is continuously reset when ADEN is low.

When initiating a single ended conversion by setting the ADSC bit in ADCSRA register, the conversion starts at the following rising edge of the ADC clock cycle.

A normal conversion takes 13 ADC clock cycles. The first conversion after the ADC is switched on (ADEN in ADCSRA register is set) takes 25 ADC clock cycles in order to initialize the analog circuitry.

The actual sample-and-hold takes place 1.5 ADC clock cycles after the start of a normal conversion and 14.5 ADC clock cycles after the start of an first conversion. When a conversion is complete, the result is written to the ADC Data Registers, and ADIF is set. In Single Conversion mode, ADSC is cleared simultaneously. The software may then set ADSC again, and a new conversion will be initiated on the first rising ADC clock edge.

When Auto Triggering is used, the prescaler is reset when the trigger event occurs. This assures a fixed delay from the trigger event to the start of conversion. In this mode, the sample-and-hold takes place 2 ADC clock cycles after the rising edge on the trigger source signal. Three additional CPU clock cycles are used for synchronization logic.

In Free Running mode, a new conversion will be started immediately after the conversion completes, while ADSC remains high. For a summary of conversion times, see Table 4-55.

4.18.9 Temperature Measurement

The temperature measurement is based on an on-chip temperature sensor that is coupled to a single ended ADC input. MUX[4..0] bits in ADMUX register enables the temperature sensor. The internal 1.1V voltage reference must also be selected for the ADC voltage reference source in the temperature sensor measurement. When the temperature sensor is enabled, the ADC converter can be used in single conversion mode to measure the voltage over the temperature sensor.

The measured voltage has a linear relationship to the temperature as described in Table 4-56. The voltage sensitivity is approximately 1LSB/°C and the accuracy of the temperature measurement is ±10°C using manufacturing calibration values (TS_GAIN, TS_OFFSET). The values described in Table 4-56 are typical values. However, due to the process variation the temperature sensor output varies from one chip to another.

Table 4-56. Temperature vs. Sensor Output Voltage (Typical Case): Example ADC Values

Temperature/°C	–40°C	+25°C	+85°C
	0x00F6	0x0144	0c01B8

4.18.10 Manufacturing Calibration

Calibration values determined during test are available in the signature row.

The temperature in degrees Celsius can be calculated using the formula:

$$T = \frac{([(ADCH \ll 8) | ADCL] - (273 + 25 - TS\_OFFSET)) \times 128}{TS\_GAIN} + 25$$

Where:

a. ADCH & ADCL are the ADC data registers,

b. is the temperature sensor gain

c. TSOFFSET is the temperature sensor offset correction term
TS_GAIN is the unsigned fixed point 8-bit temperature sensor gain factor in 1/128th units stored in the signature row
TS_OFFSET is the signed twos complement temperature sensor offset reading stored in the signature row. See Table 4-65 on page 246 for signature row parameter address.

4.23 Electrical Characteristics

Note: All Characteristics contained in this data sheet are based on simulation and characterization of Atmel® ATtiny87/167 AVR® microcontrollers manufactured in a typical process technology. These values are preliminary values representing design targets, and will be updated after characterization of actual Automotive silicon.

4.23.1 Absolute Maximum Ratings

Stresses beyond those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Parameters	Value	Unit
Operating Temperature	–40 to +125	°C
Storage Temperature	–65 to +150	°C
Voltage on any Pin except $\overline{\text{RESET}}$ with respect to Ground	–0.5 to Vcc + 0.5	V
Voltage on $\overline{\text{RESET}}$ with respect to Ground	–0.5 to +13.0	V
Voltage on Vcc with respect to Ground	–0.5 to 6.0	V
DC Current per I/O Pin	40.0	mA
DC Current Vcc and GND Pins	200.0	mA
Injection Current at VCC = 0V to 5V(2)	±5(1)	mA

Notes: 1. Maximum current per port = ±30mA
2. Functional corruption may occur

4.23.2 DC Characteristics

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 Low Voltage	Except XTAL1 and $\overline{\text{RESET}}$ pins	V_{IL}	–0.5		0.2 Vcc(2)	V
	XTAL1 pin - External Clock Selected	V_{IL1}	–0.5		0.1 Vcc(2)	V
	$\overline{\text{RESET}}$ pin	V_{IL2}	–0.5		0.2 Vcc(2)	V
	$\overline{\text{RESET}}$ pin as I/O	V_{IL3}	–0.5		0.2 Vcc(2)	V
Input High Voltage	Except XTAL1 and $\overline{\text{RESET}}$ pins	V_{IH}	0.7 Vcc(3)		Vcc + 0.5	V
	XTAL1 pin - External Clock Selected	V_{IH1}	0.8 Vcc(3)		Vcc + 0.5	V
	$\overline{\text{RESET}}$ pin	V_{IH2}	0.9 Vcc(3)		Vcc + 0.5	V
	$\overline{\text{RESET}}$ pin as I/O	V_{IH3}	0.7 Vcc(3)		Vcc + 0.5	V
Output Low Voltage(4) (Ports A, B,)	I_{OL} = 10mA, Vcc = 5V I_{OL} = 5mA, Vcc = 3V	V_{OL}			0.6 0.5	V
Output High Voltage(5) (Ports A, B)	I_{OH} = –10mA, Vcc = 5V I_{OH} = –5mA, Vcc = 3V	V_{OH}	4.3 2.5			V
Input Leakage Current I/O Pin	Vcc = 5.5V, pin low (absolute value)	I_{IL}		< 0.05	1	µA
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

4.23.10 SPI Timing Characteristics

See Figure 4-112 and Figure 4-113 for details.

Table 4-92. SPI Timing Parameters

No.	Description	Mode	Min.	Typ.	Max.	Unit
1	SCK period	Master		See Table 4-43		ns
2	SCK high/low	Master		50% duty cycle		
3	Rise/Fall time	Master		3.6		
4	Setup	Master		10		
5	Hold	Master		10		
6	Out to SCK	Master		$0.5 \bullet t_{sck}$		
7	SCK to out	Master		10		
8	SCK to out high	Master		10		
9	SS low to out	Slave		15		
10	SCK period	Slave	$4 \bullet t_{ck}$			
11	SCK high/low (1)	Slave	$2 \bullet t_{ck}$			
12	Rise/Fall time	Slave			1.6	µs
13	Setup	Slave	10			ns
14	Hold	Slave	t_{ck}			
15	SCK to out	Slave		15		
16	SCK to $\overline{SS}$ high	Slave	20			
17	$\overline{SS}$ high to tri-state	Slave		10		
18	SS low to SCK	Slave	$2 \bullet t_{ck}$			

Note: In SPI Programming mode the minimum SCK high/low period is:
- 2 t_{CLCL} for f_{CK} < 12MHz
- 3 t_{CLCL} for f_{CK} >12MHz

4.26 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									

4.27 Instruction Set Summary (Continued)

Mnemonics	Operands	Description	Operation	Flags	#Clocks
BRCS	k	Branch if Carry Set	if (C = 1) then PC ←PC + k + 1	None	1/2
BRCC	k	Branch if Carry Cleared	if (C = 0) then PC ←PC + k + 1	None	1/2
BRSH	k	Branch if Same or Higher	if (C = 0) then PC ←PC + k + 1	None	1/2
BRLO	k	Branch if Lower	if (C = 1) then PC ←PC + k + 1	None	1/2
BRMI	k	Branch if Minus	if (N = 1) then PC ←PC + k + 1	None	1/2
BRPL	k	Branch if Plus	if (N = 0) then PC ←PC + k + 1	None	1/2
BRGE	k	Branch if Greater or Equal, Signed	if (N ⊕ V= 0) then PC ←PC + k + 1	None	1/2
BRLT	k	Branch if Less Than Zero, Signed	if (N ⊕ V= 1) then PC ←PC + k + 1	None	1/2
BRHS	k	Branch if Half Carry Flag Set	if (H = 1) then PC ←PC + k + 1	None	1/2
BRHC	k	Branch if Half Carry Flag Cleared	if (H = 0) then PC ←PC + k + 1	None	1/2
BRTS	k	Branch if T Flag Set	if (T = 1) then PC ←PC + k + 1	None	1/2
BRTC	k	Branch if T Flag Cleared	if (T = 0) then PC ←PC + k + 1	None	1/2
BRVS	k	Branch if Overflow Flag is Set	if (V = 1) then PC ←PC + k + 1	None	1/2
BRVC	k	Branch if Overflow Flag is Cleared	if (V = 0) then PC ←PC + k + 1	None	1/2
BRIE	k	Branch if Interrupt Enabled	if (I = 1) then PC ←PC + k + 1	None	1/2
BRID	k	Branch if Interrupt Disabled	if (I = 0) then PC ←PC + k + 1	None	1/2
BIT AND BIT-TEST INSTRUCTIONS					
SBI	P,b	Set Bit in I/O Register	I/O(P,b) ←1	None	2
CBI	P,b	Clear Bit in I/O Register	I/O(P,b) ←0	None	2
LSL	Rd	Logical Shift Left	Rd(n+1) ←Rd(n), Rd(0) ←0	Z,C,N,V	1
LSR	Rd	Logical Shift Right	Rd(n) ←Rd(n+1), Rd(7) ←0	Z,C,N,V	1
ROL	Rd	Rotate Left Through Carry	Rd(0)←C,Rd(n+1)←Rd(n),C←Rd(7)	Z,C,N,V	1
ROR	Rd	Rotate Right Through Carry	Rd(7)←C,Rd(n)←Rd(n+1),C←Rd(0)	Z,C,N,V	1
ASR	Rd	Arithmetic Shift Right	Rd(n) ←Rd(n+1), n=0..6	Z,C,N,V	1
SWAP	Rd	Swap Nibbles	Rd(3..0)←Rd(7..4),Rd(7..4)←Rd(3..0)	None	1
BSET	s	Flag Set	SREG(s) ←1	SREG(s)	1
BCLR	s	Flag Clear	SREG(s) ←0	SREG(s)	1
BST	Rr, b	Bit Store from Register to T	T ←Rr(b)	T	1
BLD	Rd, b	Bit load from T to Register	Rd(b) ←T	None	1
SEC		Set Carry	C ←1	C	1
CLC		Clear Carry	C ←0	C	1
SEN		Set Negative Flag	N ←1	N	1
CLN		Clear Negative Flag	N ←0	N	1
SEZ		Set Zero Flag	Z ←1	Z	1
CLZ		Clear Zero Flag	Z ←0	Z	1
SEI		Global Interrupt Enable	I ←1	I	1
CLI		Global Interrupt Disable	I ←0	I	1
SES		Set Signed Test Flag	S ←1	S	1
CLS		Clear Signed Test Flag	S ←0	S	1
SEV		Set Twos Complement Overflow.	V ←1	V	1
CLV		Clear Twos Complement Overflow	V ←0	V	1
SET		Set T in SREG	T ←1	T	1
CLT		Clear T in SREG	T ←0	T	1
SEH		Set Half Carry Flag in SREG	H ←1	H	1
CLH		Clear Half Carry Flag in SREG	H ←0	H	1