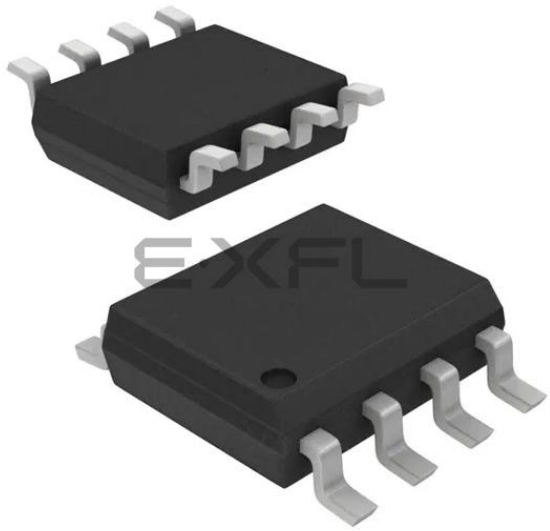


**Welcome to E-XFL.COM**

## 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 | 4MHz |
| Connectivity | SPI |
| Peripherals | Brown-out Detect/Reset, POR, WDT |
| Number of I/O | 3 |
| Program Memory Size | 2KB (1K x 16) |
| Program Memory Type | FLASH |
| EEPROM Size | 128 x 8 |
| RAM Size | 128 x 8 |
| Voltage - Supply (Vcc/Vdd) | 2.7V ~ 6V |
| Data Converters | - |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C |
| Mounting Type | Surface Mount |
| Package / Case | 8-SOIC (0.209", 5.30mm Width) |
| Supplier Device Package | 8-SOIC |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/at90ls2323-4si |

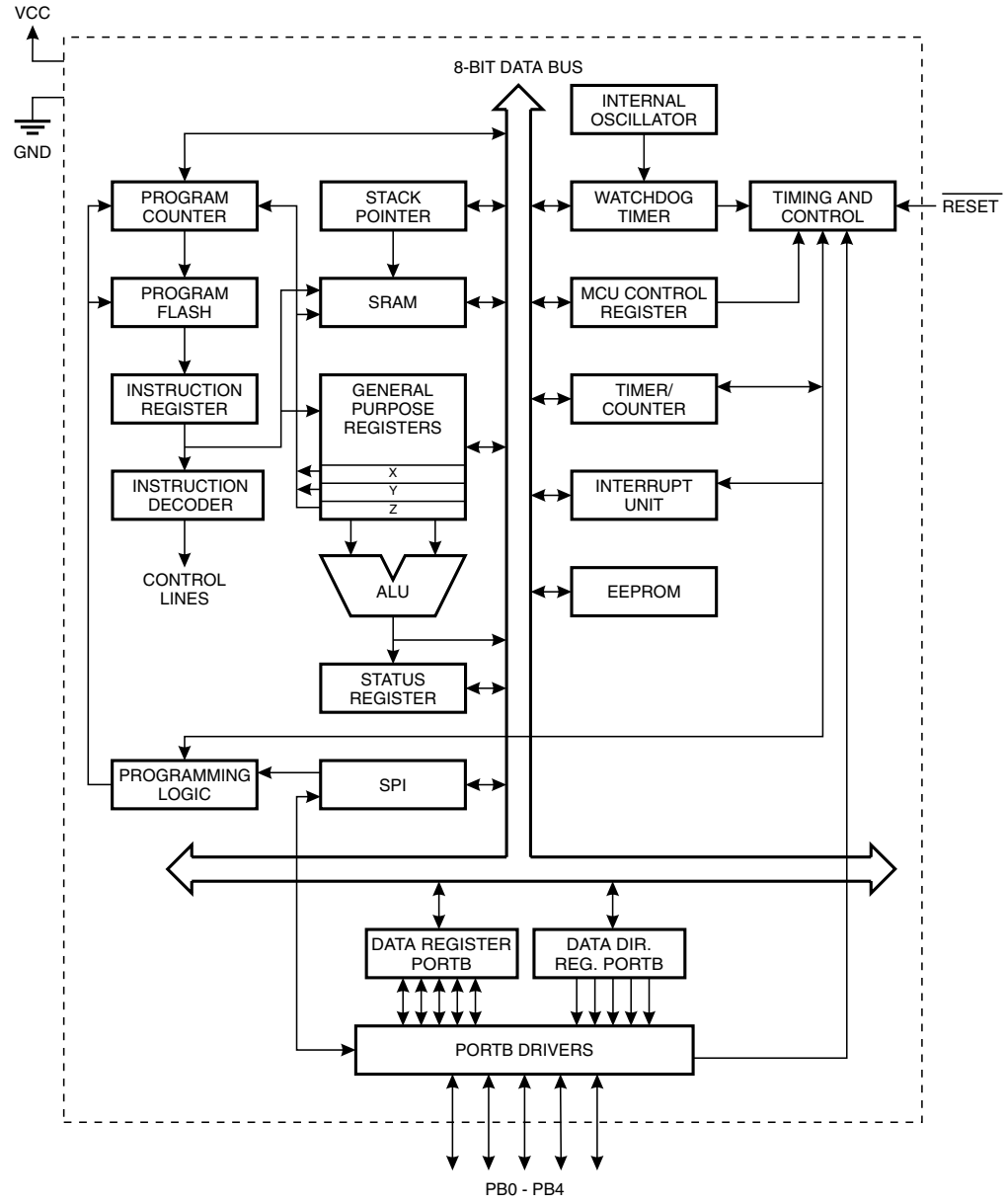
## Description

The AT90S/LS2323 and AT90S/LS2343 are low-power, CMOS, 8-bit microcontrollers based on the AVR RISC architecture. By executing powerful instructions in a single clock cycle, the AT90S2323/2343 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

The AVR core combines a rich instruction set with 32 general-purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

## Block Diagram

**Figure 1.** The AT90S/LS2343 Block Diagram



chip, the Atmel AT90S2323/2343 is a powerful microcontroller that provides a highly flexible and cost-effective solution to many embedded control applications.

The AT90S2323/2343 AVR is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, in-circuit emulators and evaluation kits.

## Comparison between AT90S/LS2323 and AT90S/LS2343

The AT90S/LS2323 is intended for use with external quartz crystal or ceramic resonator as the clock source. The start-up time is fuse-selectable as either 1 ms (suitable for ceramic resonator) or 16 ms (suitable for crystal). The device has three I/O pins.

The AT90S/LS2343 is intended for use with either an external clock source or the internal RC oscillator as clock source. The device has five I/O pins.

Table 1 summarizes the differences in features of the two devices.

**Table 1.** Feature Difference Summary

| Part | AT90S/LS2323 | AT90S/LS2343 |
|---|---|---|
| On-chip Oscillator Amplifier | yes | no |
| Internal RC Clock | no | yes |
| PB3 available as I/O pin | never | internal clock mode |
| PB4 available as I/O pin | never | always |
| Start-up time | 1 ms/16 ms | 16 µs fixed |

## Pin Descriptions AT90S/LS2323

**VCC** Supply voltage pin.

**GND** Ground pin.

**Port B (PB2..PB0)** Port B is a 3-bit bi-directional I/O port with internal pull-up resistors. The Port B output buffers can sink 20 mA. As inputs, Port B pins that are externally pulled low, will source current if the pull-up resistors are activated.

Port B also serves the functions of various special features.

Port pins can provide internal pull-up resistors (selected for each bit). The Port B pins are tri-stated when a reset condition becomes active.

**$\overline{\text{RESET}}$** Reset input. An external reset is generated by a low level on the $\overline{\text{RESET}}$ pin. Reset pulses longer than 50 ns will generate a reset, even if the clock is not running. Shorter pulses are not guaranteed to generate a reset.

**XTAL1** Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

**XTAL2** Output from the inverting oscillator amplifier.

The AVR has Harvard architecture – with separate memories and buses for program and data. The program memory is accessed with a two-stage pipeline. While one instruction is being executed, the next instruction is pre-fetched from the program memory. This concept enables instructions to be executed in every clock cycle. The program memory is in-system downloadable Flash memory.

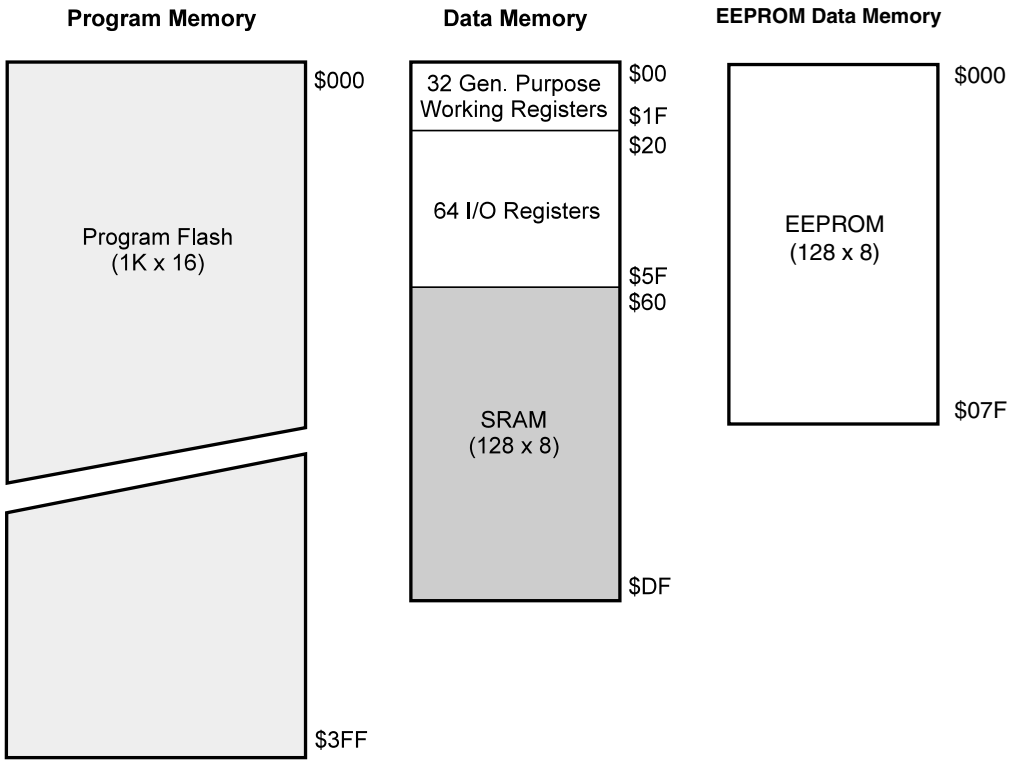
With the relative jump and call instructions, the whole 1K address space is directly accessed. Most AVR instructions have a single 16-bit word format. Every program memory address contains a 16- or 32-bit instruction.

During interrupts and subroutine calls, the return address Program Counter (PC) is stored on the stack. The stack is effectively allocated in the general data SRAM and consequently, the stack size is only limited by the total SRAM size and the usage of the SRAM. All user programs must initialize the SP in the reset routine (before subroutines or interrupts are executed). The 8-bit stack pointer (SP) is read/write-accessible in the I/O space.

The 128 bytes data SRAM + register file and I/O registers can be easily accessed through the five different addressing modes supported in the AVR architecture.

The memory spaces in the AVR architecture are all linear and regular memory maps.

**Figure 6.** Memory Maps



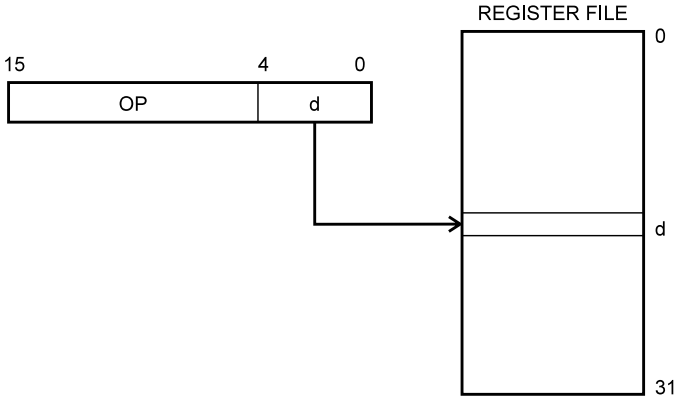

A flexible interrupt module has its control registers in the I/O space with an additional global interrupt enable bit in the status register. All the different interrupts have a separate interrupt vector in the interrupt vector table at the beginning of the program memory. The different interrupts have priority in accordance with their interrupt vector position. The lower the interrupt vector address, the higher the priority.

## Program and Data Addressing Modes

The AT90S2323/2343 AVR RISC microcontroller supports powerful and efficient addressing modes for access to the program memory (Flash) and data memory. This section describes the different addressing modes supported by the AVR architecture. In the figures, OP means the operation code part of the instruction word. To simplify, not all figures show the exact location of the addressing bits.

### Register Direct, Single Register Rd

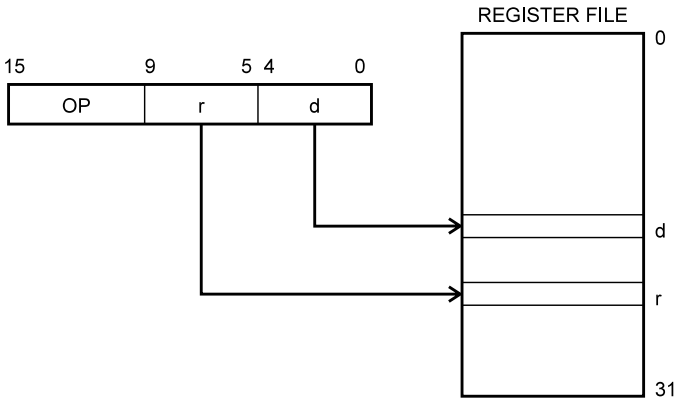
**Figure 10.** Direct Single Register Addressing




The operand is contained in register d (Rd).

### Register Direct, Two Registers Rd and Rr

**Figure 11.** Direct Register Addressing, Two Registers




Operands are contained in register r (Rr) and d (Rd). The result is stored in register d (Rd).

- **Bit 0 – C: Carry Flag**

The carry flag C indicates a carry in an arithmetical or logical operation. See the Instruction Set description for detailed information.

Note that the Status Register is not automatically stored when entering an interrupt routine and restored when returning from an interrupt routine. This must be handled by software.

**Stack Pointer – SPL**

An 8-bit register at I/O address $3D ($5D) forms the stack pointer of the AT90S2323/2343. Eight bits are used to address the 128 bytes of SRAM in locations $60 - $DF.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| $3D ($5D) | SP7 | SP6 | SP5 | SP4 | SP3 | SP2 | SP1 | SP0 | SPL |
| 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 | |

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 $60. The Stack Pointer is decremented by 1 when data is pushed onto the Stack with the PUSH instruction and it is decremented by 2 when an address is pushed onto the stack with subroutine calls and interrupts. The Stack Pointer is incremented by 1 when data is popped from the stack with the POP instruction and it is incremented by 2 when an address is popped from the stack with return from subroutine RET or return from interrupt RETI.

## Reset and Interrupt Handling

The AT90S2323/2343 provides two interrupt sources. These interrupts and the separate reset vector each have a separate program vector in the program memory space. Both interrupts are assigned individual enable bits that must be set (one) together with the I-bit in the Status Register in order to enable the interrupt.

The lowest addresses in the program memory space are automatically defined as the Reset and Interrupt vectors. The complete list of vectors is shown in Table 3. The list also determines the priority levels of the interrupts. The lower the address, the higher the priority level. RESET has the highest priority, and next is INT0 (the External Interrupt Request 0), etc.

**Table 3.** Reset and Interrupt Vectors

| Vector No. | Program Address | Source | Interrupt Definition |
|---|---|---|---|
| 1 | $000 | RESET | Hardware Pin, Power-on Reset and Watchdog Reset |
| 2 | $001 | INT0 | External Interrupt Request 0 |
| 3 | $002 | TIMER0, OVF0 | Timer/Counter0 Overflow |

**Table 4.** Reset Characteristics ($V_{CC}$ = 5.0V)

| Symbol | Parameter | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| $V_{POT}$(1) | Power-on Reset Threshold Voltage, rising | 1.0 | 1.4 | 1.8 | V |
| | Power-on Reset Threshold Voltage, falling | 0.4 | 0.6 | 0.8 | V |
| $V_{RST}$ | $\overline{\text{RESET}}$ Pin Threshold Voltage | | 0.6 $V_{CC}$ | | V |
| $t_{TOUT}$ | Reset Delay Time-out Period AT90S/LS2323 FSTRT Programmed | 1.0 | 1.1 | 1.2 | ms |
| $t_{TOUT}$ | Reset Delay Time-out Period AT90S/LS2323 FSTRT Unprogrammed | 11.0 | 16.0 | 21.0 | ms |
| $t_{TOUT}$ | Reset Delay Time-out Period AT90S/LS2343 | 11.0 | 16.0 | 21.0 | µs |

Note: 1. The Power-on Reset will not work unless the supply voltage has been below $V_{POT}$ (falling).

**Table 5.** Reset Characteristics ($V_{CC}$ = 3.0V)

| Symbol | Parameter | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| $V_{POT}$(1) | Power-on Reset Threshold Voltage, rising | 1.0 | 1.4 | 1.8 | V |
| | Power-on Reset Threshold Voltage, falling | 0.4 | 0.6 | 0.8 | V |
| $V_{RST}$ | $\overline{\text{RESET}}$ Pin Threshold Voltage | | 0.6 $V_{CC}$ | | V |
| $t_{TOUT}$ | Reset Delay Time-out Period AT90S/LS2323 FSTRT Programmed | 2.0 | 2.2 | 2.4 | ms |
| $t_{TOUT}$ | Reset Delay Time-out Period AT90S/LS2323 FSTRT Unprogrammed | 22.0 | 32.0 | 42.0 | ms |
| $t_{TOUT}$ | Reset Delay Time-out Period AT90S/LS2343 | 22.0 | 32.0 | 42.0 | µs |

Note: 1. The Power-on Reset will not work unless the supply voltage has been below $V_{POT}$ (falling).

**Power-on Reset**

The AT90S2323/2343 is designed for use in systems where it can operate from the internal RC oscillator (AT90S/LS2343), on-chip oscillator (AT90S/LS2323), or in applications where a clock signal is provided by an external clock source. After $V_{CC}$ has reached $V_{POT}$, the device will start after the time $t_{TOUT}$ (see Figure 25). If the clock signal is provided by an external clock source, the clock must not be applied until $V_{CC}$ has reached the minimum voltage defined for the applied frequency.

For AT90S2323, the user can select the start-up time according to typical oscillator start-up. The number of WDT oscillator cycles used for each time-out is shown in Table 6. For AT90S2343, the start-up time is one Watchdog cycle only. The frequency of the Watchdog oscillator is voltage-dependent as shown in “Typical Characteristics” on page 49.

**Table 6.** Number of Watchdog Oscillator Cycles

| FSTRT | Time-out at $V_{CC}$ = 5V | Number of WDT Cycles |
|---|---|---|
| Programmed | 1.1 ms | 1K |
| Unprogrammed | 16.0 ms | 16K |

**Table 8.** Reset Source Identification

| PORF | EXTRF | Reset Source |
|---|---|---|
| 0 | 0 | Watchdog Reset |
| 0 | 1 | External Reset |
| 1 | 0 | Power-on Reset |
| 1 | 1 | Power-on Reset |

## Interrupt Handling

The AT90S2323/2343 has two 8-bit interrupt mask control registers; GIMSK (General Interrupt Mask register) and TIMSK (Timer/Counter Interrupt Mask register).

When an interrupt occurs, the Global Interrupt Enable I-bit is cleared (zero) and all interrupts are disabled. The user software can set (one) the I-bit to enable nested interrupts. The I-bit is set (one) when a Return from Interrupt instruction (RETI) is executed.

When the Program Counter is vectored to the actual interrupt vector in order to execute the interrupt handling routine, hardware clears the corresponding flag that generated the interrupt. Some of the interrupt flags can also be cleared by writing a logical “1” to the flag bit position(s) to be cleared. If an interrupt condition occurs when the corresponding interrupt enable bit is cleared (zero), the interrupt flag will be set and remembered until the interrupt is enabled or the flag is cleared by software.

If one or more interrupt conditions occur when the global interrupt enable bit is cleared (zero), the corresponding interrupt flag(s) will be set and remembered until the global interrupt enable bit is set (one) and will be executed by order of priority.

Note that external level interrupt does not have a flag and will only be remembered for as long as the interrupt condition is active.

Note that the Status Register is not automatically stored when entering an interrupt routine and restored when returning from an interrupt routine. This must be handled by software.

## General Interrupt Mask Register – GIMSK

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| $3B ($5B) | – | INT0 | – | – | – | – | – | – | GIMSK |
| Read/Write | R | R/W | R | R | R | R | R | R | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

- **Bit 7 – Res: Reserved Bit**

This bit is a reserved bit in the AT90S2323/2343 and always reads as zero.

- **Bit 6 – INT0: External Interrupt Request 0 Enable**

When the INT0 bit is set (one) and the I-bit in the Status Register (SREG) is set (one), the external pin interrupt is enabled. The Interrupt Sense Control0 bits 1/0 (ISC01 and ISC00) in the MCU general Control Register (MCUCR) define whether the external interrupt is activated on rising or falling edge of the INT0 pin or level sensed. Activity on the pin will cause an interrupt request even if INT0 is configured as an output. The corresponding interrupt of External Interrupt Request 0 is executed from program memory address $001. See also “External Interrupts.”

- **Bits 5..0 – Res: Reserved Bits**

These bits are reserved bits in the AT90S2323/2343 and always read as zero.

- **Bit 0 – Res: Reserved Bit**

This bit is a reserved bit in the AT90S2323/2343 and always reads zero.

**External Interrupt**

The external interrupt is triggered by the INT0 pin. Observe that, if enabled, the interrupt will trigger even if the INT0 pin is configured as an output. This feature provides a way of generating a software interrupt. The external interrupt can be triggered by a falling or rising edge or a low level. This is set up as indicated in the specification for the MCU Control Register (MCUCR). When the external interrupt is enabled and is configured as level-triggered, the interrupt will trigger as long as the pin is held low.

The external interrupt is set up as described in the specification for the MCU Control Register (MCUCR).

**Interrupt Response Time**

The interrupt execution response for all the enabled AVR interrupts is four clock cycles minimum. Four clock cycles after the interrupt flag has been set, the program vector address for the actual interrupt handling routine is executed. During these four clock cycles, the Program Counter (2 bytes) is popped back from the stack, the Stack Pointer is incremented by 2 and the I-flag in SREG is set. The vector is a relative jump to the interrupt routine and this jump takes two clock cycles. If an interrupt occurs during execution of a multi-cycle instruction, this instruction is completed before the interrupt is served.

A return from an interrupt handling routine (same as for a subroutine call routine) takes four clock cycles. During these four clock cycles, the Program Counter (2 bytes) is popped back from the stack and the Stack Pointer is incremented by 2. When the AVR exits from an interrupt, it will always return to the main program and execute one more instruction before any pending interrupt is served.

**MCU Control Register – MCUCR**

The MCU Control Register contains control bits for general MCU functions.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| $35 ($55) | – | – | SE | SM | – | – | ISC01 | ISC00 | MCUCR |
| Read/Write | R | R | R/W | R/W | R | R | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

- **Bits 7, 6 – Res: Reserved Bits**

These bits are reserved bits in the AT90S2323/2343 and always read as zero.

- **Bit 5 – SE: Sleep Enable**

The SE bit must be set (one) to make the MCU enter the Sleep mode when the SLEEP instruction is executed. To avoid the MCU entering the Sleep mode, unless it is the programmer's purpose, it is recommended to set the Sleep Enable (SE) bit just before the execution of the SLEEP instruction.

- **Bit 4 – SM: Sleep Mode**

This bit selects between the two available sleep modes. When SM is cleared (zero), Idle mode is selected as Sleep mode. When SM is set (one), Power-down mode is selected as sleep mode. For details, refer to the section "Sleep Modes".

- **Bits 3, 2 – Res: Reserved Bits**

These bits are reserved bits in the AT90S2323/2343 and always read as zero.

- **Bits 1, 0 – ISC01, ISC00: Interrupt Sense Control 0 Bit 1 and Bit 0**

The External Interrupt 0 is activated by the external pin INT0 if the SREG I-flag and the corresponding interrupt mask are set. The level and edges on the external INT0 pin that

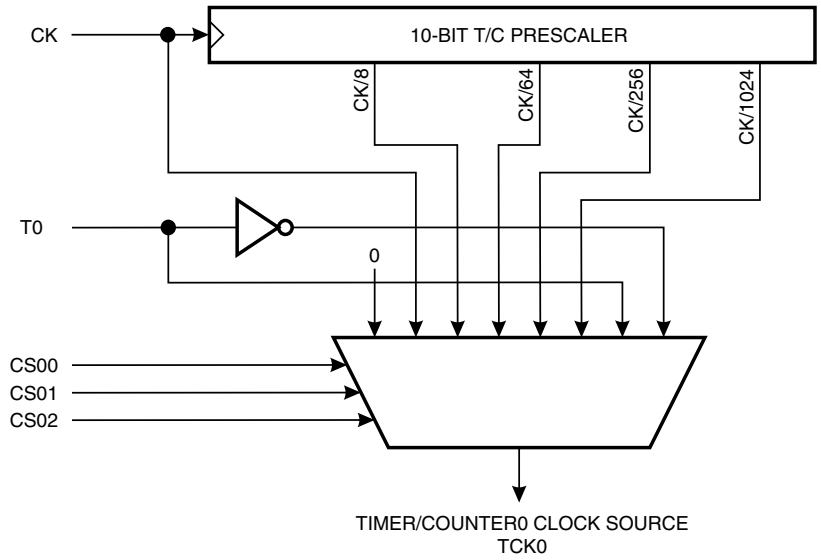
## Timer/Counter

The AT90S2323/2343 provides one general-purpose 8-bit Timer/Counter – Timer/Counter0. The Timer/Counter has prescaling selection from the 10-bit prescaling timer. The Timer/Counter can be used either as a timer with an internal clock time base or as a counter with an external pin connection that triggers the counting.

## Timer/Counter Prescaler

Figure 29 shows the Timer/Counter prescaler.

**Figure 29.** Timer/Counter0 Prescaler




The four different prescaled selections are: CK/8, CK/64, CK/256 and CK/1024, where CK is the oscillator clock. CK, external source and stop can also be selected as clock sources.

## 8-bit Timer/Counter0

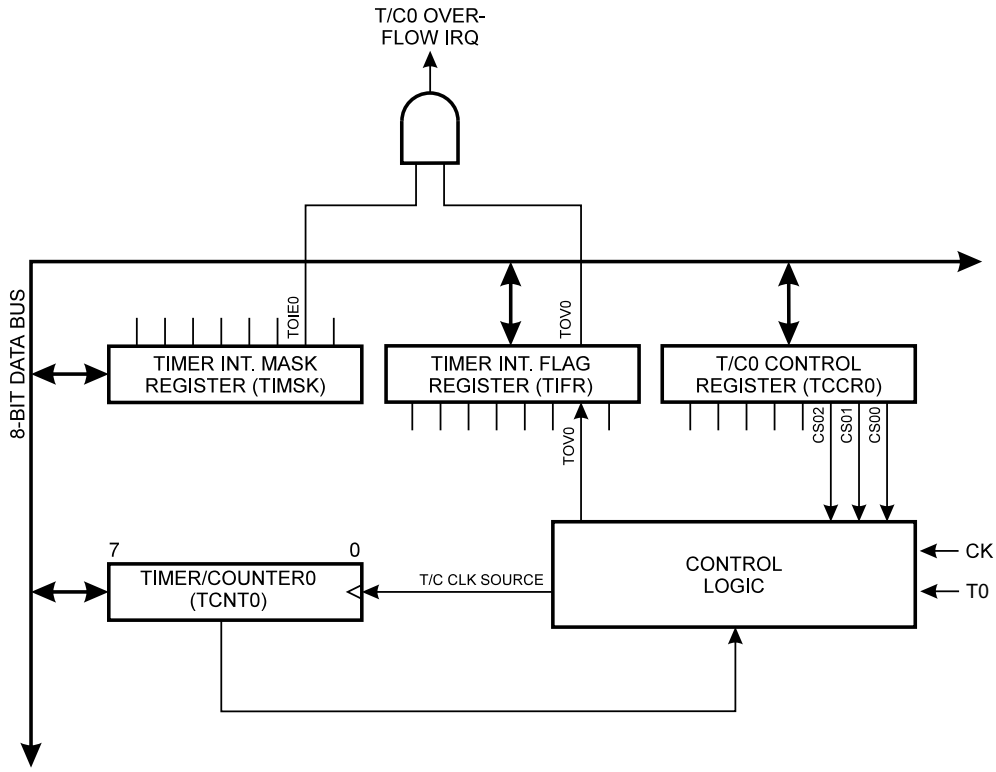
Figure 30 shows the block diagram for Timer/Counter0.

The 8-bit Timer/Counter0 can select clock source from CK, prescaled CK or an external pin. In addition, it can be stopped as described in the specification for the Timer/Counter0 Control Register (TCCR0). The overflow status flag is found in the Timer/Counter Interrupt Flag Register (TIFR). Control signals are found in the Timer/Counter0 Control Register (TCCR0). The interrupt enable/disable settings for Timer/Counter0 are found in the Timer/Counter Interrupt Mask Register (TIMSK).

When Timer/Counter0 is externally clocked, the external signal is synchronized with the oscillator frequency of the CPU. To ensure proper sampling of the external clock, the minimum time between two external clock transitions must be at least one internal CPU clock period. The external clock signal is sampled on the rising edge of the internal CPU clock.

The 8-bit Timer/Counter0 features both a high-resolution and a high-accuracy usage with the lower prescaling opportunities. Similarly, the high prescaling opportunities make the Timer/Counter0 useful for lower speed functions or exact timing functions with infrequent actions.

**Figure 30.** Timer/Counter 0 Block Diagram




## Timer/Counter0 Control Register – TCCR0

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| $33 ($53) | – | – | – | – | – | CS02 | CS01 | CS00 | TCCR0 |
| 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 – Res: Reserved Bits**

These bits are reserved bits in the AT90S2323/2343 and always read zero.

- **Bits 2, 1, 0 – CS02, CS01, CS00: Clock Select0, Bits 2, 1 and 0**

The Clock Select0 bits 2, 1 and 0 define the prescaling source of Timer/Counter0.

**Table 10.** Clock 0 Prescale Select

| CS02 | CS01 | CS00 | Description |
|---|---|---|---|
| 0 | 0 | 0 | Stop, the Timer/Counter0 is stopped. |
| 0 | 0 | 1 | CK |
| 0 | 1 | 0 | CK/8 |
| 0 | 1 | 1 | CK/64 |
| 1 | 0 | 0 | CK/256 |
| 1 | 0 | 1 | CK/1024 |
| 1 | 1 | 0 | External Pin T0, falling edge |
| 1 | 1 | 1 | External Pin T0, rising edge |

## EEPROM Read/Write Access

The EEPROM access registers are accessible in the I/O space.

The write access time is in the range of 2.5 - 4 ms, depending on the $V_{CC}$ voltages. A self-timing function, however, lets the user software detect when the next byte can be written.

In order to prevent unintentional EEPROM writes, a specific write procedure must be followed. Refer to the description of the EEPROM Control Register for details on this.

When the EEPROM is written, the CPU is halted for two clock cycles before the next instruction is executed. When the EEPROM is read, the CPU is halted for four clock cycles before the next instruction is executed.

### EEPROM Address Register – EEAR

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| $1E ($3E) | – | EEAR6 | EEAR5 | EEAR4 | EEAR3 | EEAR2 | EEAR1 | EEAR0 | EEAR |
| Read/Write | R | 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 – Res: Reserved Bit**

This bit is a reserved bit in the AT90S2323/2343 and will always read as zero.

- **Bit 6..0 – EEAR6..0: EEPROM Address**

The EEPROM Address Register (EEAR6..0) specifies the EEPROM address in the 128-byte EEPROM space. The EEPROM data bytes are addressed linearly between 0 and 127.

### EEPROM Data Register – EEDR

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| $1D ($3D) | MSB | | | | | | | LSB | EEDR |
| 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..0 – EEDR7..0: EEPROM Data**

For the EEPROM write operation, the EEDR register contains the data to be written to the EEPROM in the address given by the EEAR register. For the EEPROM read operation, the EEDR contains the data read out from the EEPROM at the address given by EEAR.

3. $002: $03 (indicates AT90S/LS2343 when signature byte $001 is $91)

Note: When both Lock bits are programmed (Lock mode 3), the signature bytes cannot be read in the low-voltage Serial mode. Reading the signature bytes will return: $00, $01 and $02.

## Programming the Flash and EEPROM

Atmel's AT90S2323/2343 offers 2K bytes of In-System Programmable Flash program memory and 128 bytes of EEPROM data memory.

The AT90S2323/2343 is shipped with the On-chip Flash program and EEPROM data memory arrays in the erased state (i.e., contents = $FF) and ready to be programmed.

The device supports a high-voltage (12V) Serial Programming mode and a low-voltage Serial Programming mode. The +12V is used for programming enable only and no current of significance is drawn by this pin. The low-voltage Serial Programming mode provides a convenient way to download program and data into the device inside the user's system.

The program and EEPROM memory arrays in the AT90S2323/2343 are programmed byte-by-byte in either programming modes. For the EEPROM, an auto-erase cycle is provided within the self-timed write instruction in the low-voltage Serial Programming mode.

During programming, the supply voltage must be in accordance with Table 15.

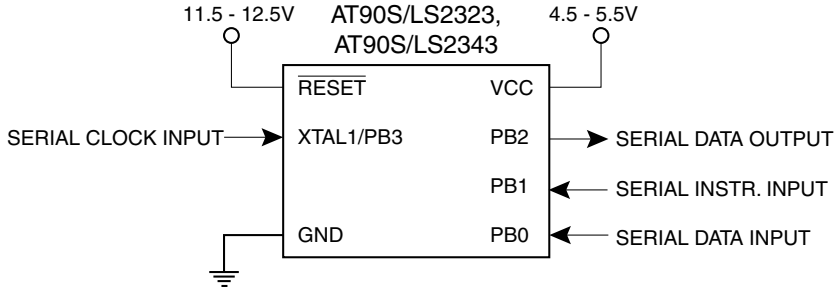
**Table 15.** Supply Voltage during Programming

| Part | Low-voltage Serial Programming | High-voltage Serial Programming |
|---|---|---|
| AT90S2323 | 4.0 - 6.0V | 4.5 - 5.5V |
| AT90LS2323 | 2.7 - 6.0V | 4.5 - 5.5V |
| AT90S2323 | 4.0 - 6.0V | 4.5 - 5.5V |
| AT90LS2323 | 2.7 - 6.0V | 4.5 - 5.5V |

## High-voltage Serial Programming

This section describes how to program and verify Flash program memory, EEPROM data memory, Lock bits and Fuse bits in the AT90S2323/2343.

**Figure 32.** High-voltage Serial Programming



## High-voltage Serial Programming Characteristics

**Figure 34.** High-voltage Serial Programming Timing

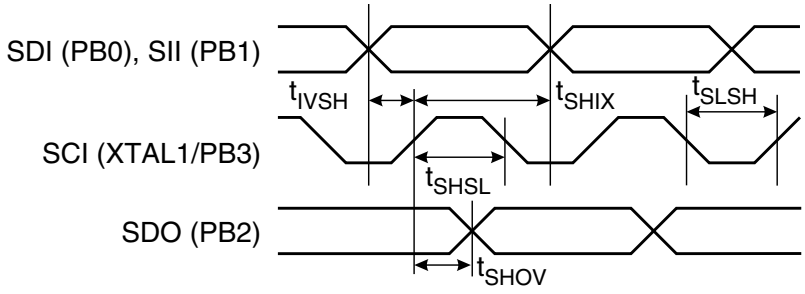


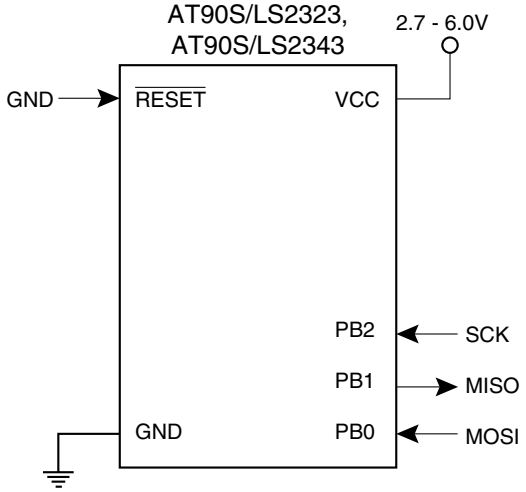

**Table 17.** High-voltage Serial Programming Characteristics, $T_A$ = 25°C ± 10%, $V_{CC}$ = 5.0V ± 10% (unless otherwise noted)

| Symbol | Parameter | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| $t_{SHSL}$ | SCI (XTAL1/PB3) Pulse Width High | 100.0 | | | ns |
| $t_{SLSH}$ | SCI (XTAL1/PB3) Pulse Width Low | 100.0 | | | ns |
| $t_{IVSH}$ | SDI (PB0), SII (PB1) Valid to SCI (XTAL1/PB3) High | 50.0 | | | ns |
| $t_{SHIX}$ | SDI (PB0), SII (PB1) Hold after SCI (XTAL1/PB3) High | 50.0 | | | ns |
| $t_{SHOV}$ | SCI (XTAL1/PB3) High to SDO (PB2) Valid | 10.0 | 16.0 | 32.0 | ns |
| $t_{WLWH\_CE}$ | Wait after Instr.3 for Chip Erase | 5.0 | 10.0 | 15.0 | ms |
| $t_{WLWH\_PFB}$ | Wait after Instr.3 for Write Fuse Bits | 1.0 | 1.5 | 1.8 | ms |

## Low-voltage Serial Downloading

Both the program and data memory arrays can be programmed using the serial SPI bus while $\overline{\text{RESET}}$ is pulled to GND. The serial interface consists of pins SCK, MOSI (input) and MISO (output) (see Figure 35). After $\overline{\text{RESET}}$ is set low, the Programming Enable instruction needs to be executed first before program/erase instructions can be executed.

**Figure 35.** Low-voltage Serial Programming and Verify



For the EEPROM, an auto-erase cycle is provided within the self-timed Write instruction and there is no need to first execute the Chip Erase instruction. The Chip Erase instruction turns the content of every memory location in both the program and EEPROM arrays into $FF.

The program and EEPROM memory arrays have separate address spaces: $0000 to $03FF for Flash program memory and $000 to $07F for EEPROM data memory.

Either an external clock is applied to the XTAL1/PB3 pin or the device must be clocked from the internal RC oscillator (AT90S/LS2343 only). The minimum low and high periods for the serial clock (SCK) input are defined as follows:

Low: > 2 MCU clock cycles

High: > 2 MCU clock cycles

**Low-voltage Serial Programming Algorithm**

When writing serial data to the AT90S2323/2343, data is clocked on the rising edge of SCK.

When reading data from the AT90S2323/2343, data is clocked on the falling edge of SCK. See Figure 36, Figure 37 and Table 20 for timing details.

To program and verify the AT90S2323/2343 in the low-voltage Serial Programming mode, the following sequence is recommended (see 4-byte instruction formats in Table 19):

1. Power-up sequence:

   Apply power between $V_{CC}$ and GND while RESET and SCK are set to “0”. (If the programmer cannot guarantee that SCK is held low during power-up, RESET must be given a positive pulse after SCK has been set to “0”.) If the device is programmed for external clocking, apply a 0 - 8 MHz clock to the XTAL1/PB3 pin. If the internal RC oscillator is selected as the clock source, no external clock source needs to be applied (AT90S/LS2343 only).
2. Wait for at least 20 ms and enable serial programming by sending the Programming Enable serial instruction to the MOSI (PB0) pin. Refer to the above section for minimum low and high periods for the serial clock input, SCK.
3. The serial programming instructions will not work if the communication is out of synchronization. When in sync, the second byte ($53) will echo back when issuing the third byte of the Programming Enable instruction. Whether the echo is correct or not, all four bytes of the instruction must be transmitted. If the $53 did not echo back, give SCK a positive pulse and issue a new Programming Enable instruction. If the $53 is not seen within 32 attempts, there is no functional device connected.
4. If a Chip Erase is performed (must be done to erase the Flash), wait $t_{WD\_ERASE}$ after the instruction, give $\overline{\text{RESET}}$ a positive pulse and start over from step 2. See Table 21 on page 46 for $t_{WD\_ERASE}$ value.
5. The Flash or EEPROM array is programmed one byte at a time by supplying the address and data together with the appropriate Write instruction. An EEPROM memory location is first automatically erased before new data is written. Use Data Polling to detect when the next byte in the Flash or EEPROM can be written. If polling is not used, wait $t_{WD\_PROG}$ before transmitting the next instruction. See Table 22 on page 46 for $t_{WD\_PROG}$ value. In an erased device, no $FFs in the data file(s) need to be programmed.
6. Any memory location can be verified by using the Read instruction, which returns the content at the selected address at the serial output MISO (PB1) pin.

# Electrical Characteristics

## Absolute Maximum Ratings*

| | |
|---|---|
| Operating Temperature | -55°C to +125°C |
| Storage Temperature | -65°C to +150°C |
| Voltage on Any Pin except $\overline{\text{RESET}}$ with respect to Ground | -1.0V to $V_{CC}$ + 0.5V |
| Voltage on $\overline{\text{RESET}}$ with Respect to Ground | -1.0V to +13.0V |
| Maximum Operating Voltage | 6.6V |
| DC Current per I/O Pin | 40.0 mA |
| DC Current $V_{CC}$ and GND Pins | 200.0 mA |

*NOTICE: 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 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.

## DC Characteristics

$T_A$ = -40°C to 85°C, $V_{CC}$ = 2.7V to 6.0V (unless otherwise noted)

| Symbol | Parameter | Condition | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input Low Voltage | (Except XTAL) | -0.5 | | 0.3 $V_{CC}$(1) | V |
| $V_{IL1}$ | Input Low Voltage | XTAL | -0.5 | | 0.1(1) | V |
| $V_{IH}$ | Input High Voltage | (Except XTAL, $\overline{\text{RESET}}$) | 0.6 $V_{CC}$(2) | | $V_{CC}$ + 0.5 | V |
| $V_{IH1}$ | Input High Voltage | XTAL | 0.7 $V_{CC}$(2) | | $V_{CC}$ + 0.5 | V |
| $V_{IH2}$ | Input High Voltage | RESET | 0.85 $V_{CC}$(2) | | $V_{CC}$ + 0.5 | V |
| $V_{OL}$ | Output Low Voltage Ports B | $I_{OL}$ = 20 mA, $V_{CC}$ = 5V<br>$I_{OL}$ = 10 mA, $V_{CC}$ = 3V | | | 0.5<br>0.4 | V<br>V |
| $V_{OH}$ | Output High Voltage Ports B | $I_{OH}$ = -3 mA, $V_{CC}$ = 5V<br>$I_{OH}$ = -1.5 mA, $V_{CC}$ = 3V | 4.2<br>2.4 | | | V<br>V |
| $I_{IL}$ | Input Leakage Current I/O Pin | $V_{CC}$ = 6V, Pin Low (absolute value) | | | 8.0 | µA |
| $I_{IH}$ | Input Leakage Current I/O Pin | $V_{CC}$ = 6V, Pin High (absolute value) | | | 8.0 | µA |
| RRST | Reset Pull-up | | 100.0 | | 500.0 | kΩ |
| $R_{I/O}$ | I/O Pin Pull-up | | 30.0 | | 150.0 | kΩ |
| $I_{CC}$ | Power Supply Current AT90S2343 | Active 4 MHz, $V_{CC}$ = 3V | | | 3.0 | mA |
| | | Idle 4 MHz, $V_{CC}$ = 3V | | | 1.1 | mA |
| | | Power-down 4 MHz(3), $V_{CC}$ = 3V WDT Enabled | | | 25.0 | µA |
| | | Power-down 4 MHz(3), $V_{CC}$ = 3V WDT Disabled | | | 20.0 | µA |
| | Power Supply Current AT90S2323 | Active 4 MHz, $V_{CC}$ = 3V | | | 4.0 | mA |
| | | Idle 4 MHz, $V_{CC}$ = 3V | | 1.0 | 1.2 | mA |
| | | Power-down(3), $V_{CC}$ = 3V WDT Enabled | | 9.0 | 15.0 | µA |
| | | Power-down(3), $V_{CC}$ = 3V WDT Disabled | | <1.0 | 2.0 | µA |

Notes: 1. "Max" means the highest value where the pin is guaranteed to be read as low.
2. "Min" means the lowest value where the pin is guaranteed to be read as high.
3. Minimum $V_{CC}$ for Power-down is 2V.

**Figure 40.** Active Supply Current vs. $V_{CC}$

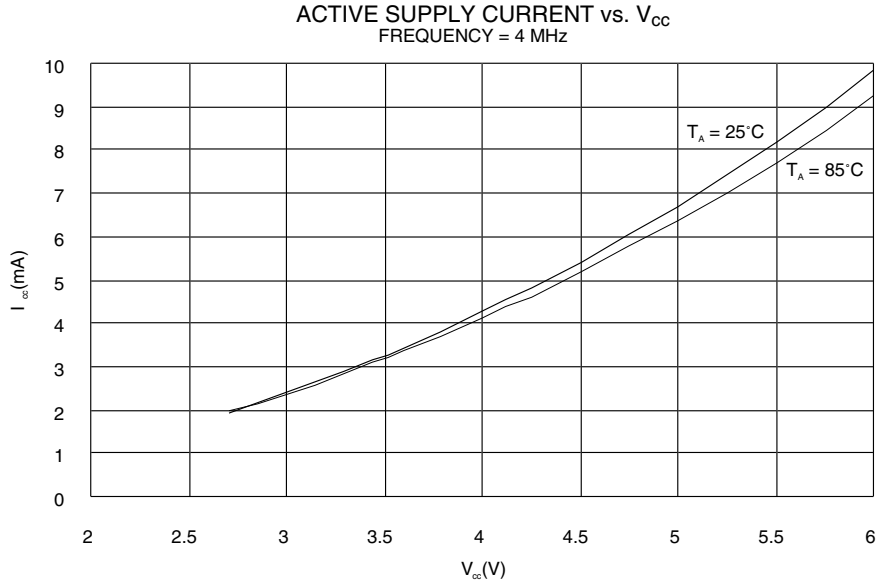



**Figure 41.** Active Supply Current vs. $V_{CC}$

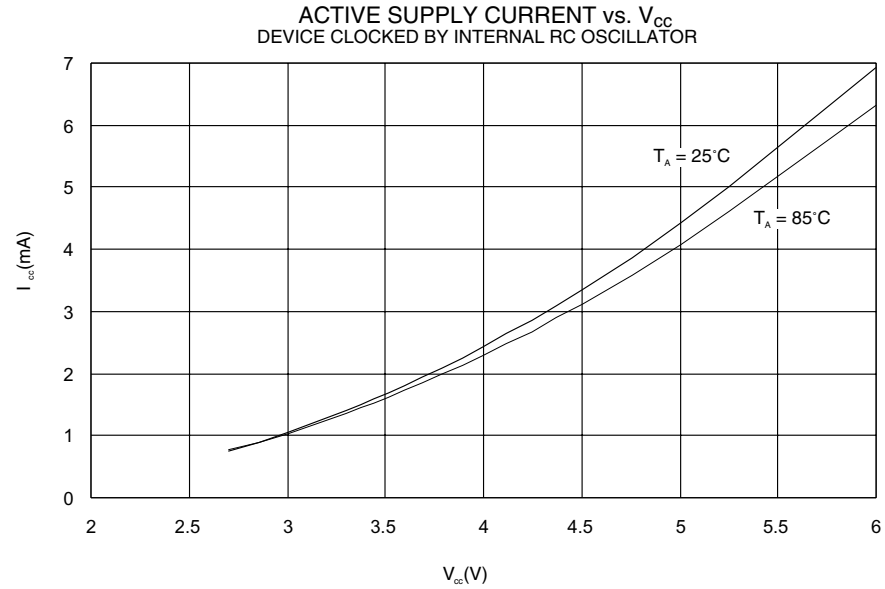


**Figure 44.** Idle Supply Current vs. $V_{CC}$

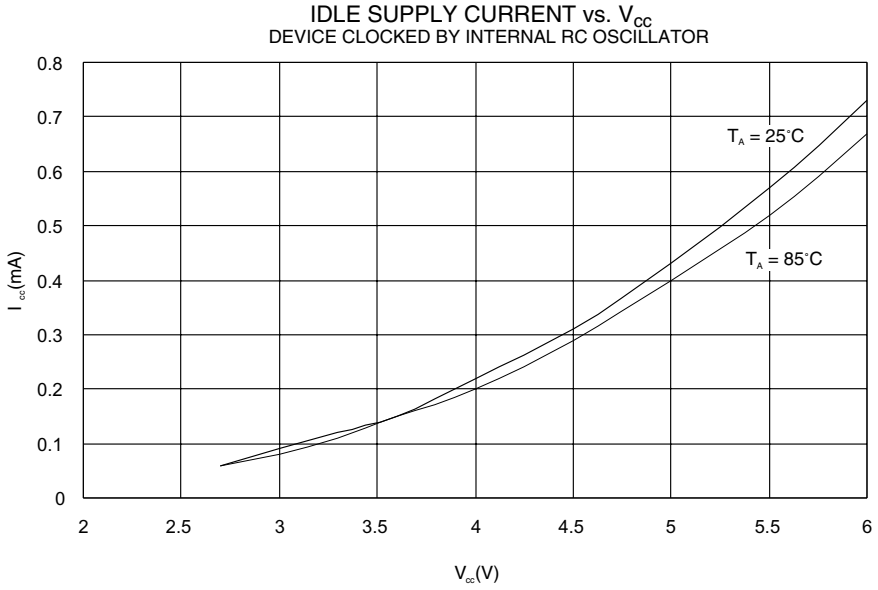
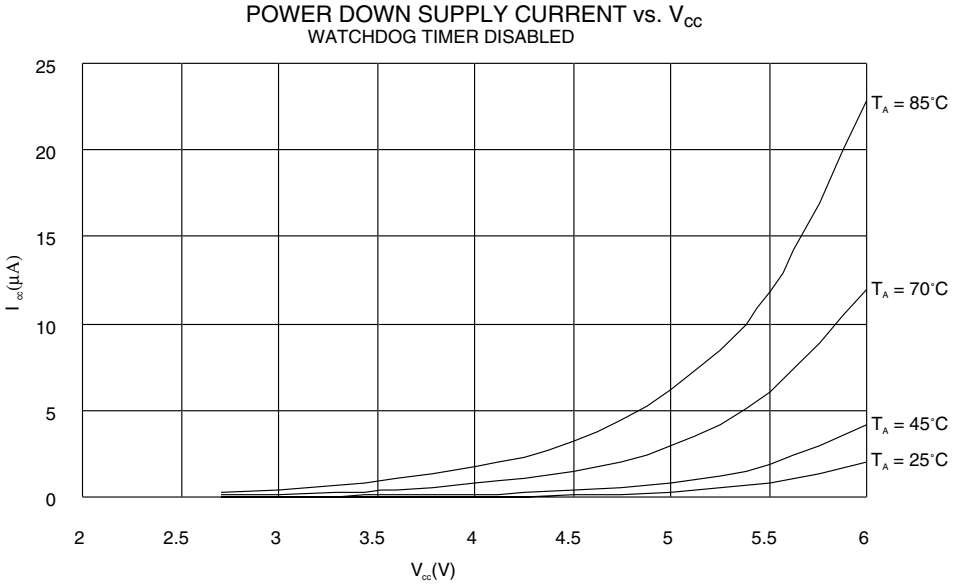



**Figure 45.** Power-down Supply Current vs. $V_{CC}$



Note: Sink and source capabilities of I/O ports are measured on one pin at a time.

**Figure 48.** Pull-up Resistor Current vs. Input Voltage

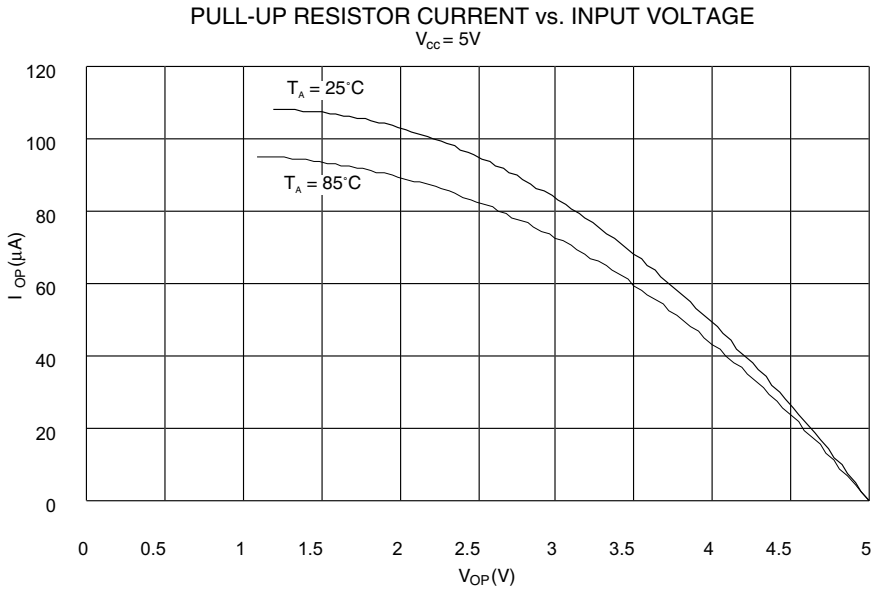
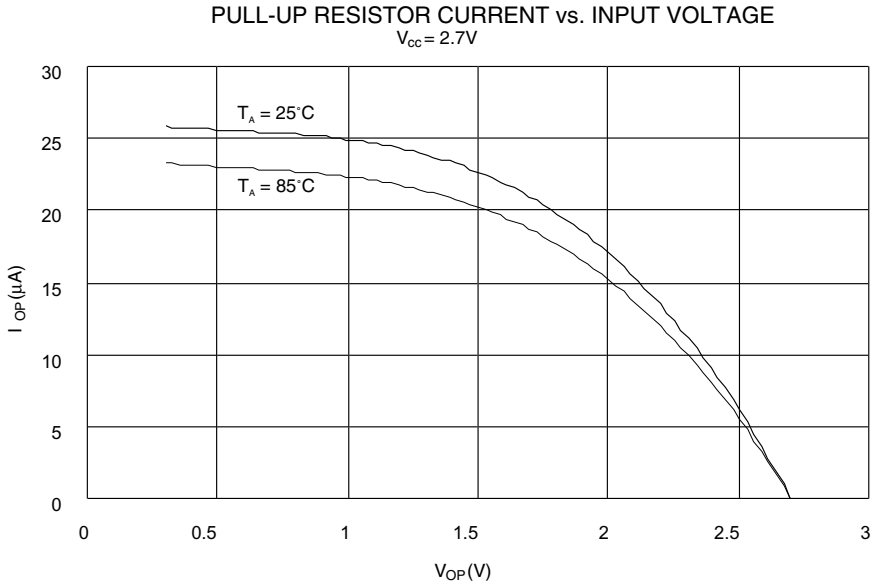



**Figure 49.** Pull-up Resistor Current vs. Input Voltage



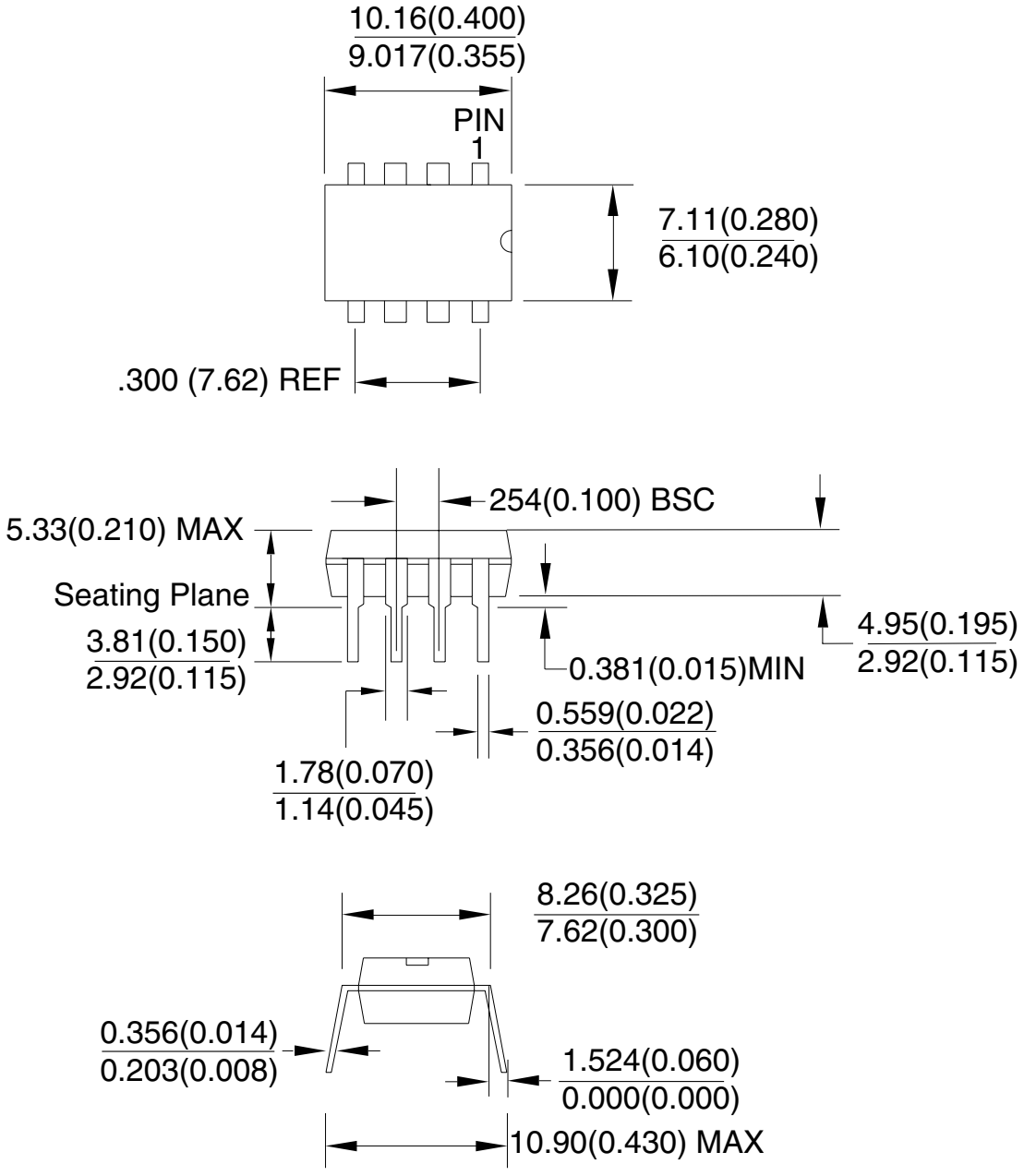
## Instruction Set Summary (Continued)

| Mnemonic | Operands | Description | Operation | Flags | # Clocks |
|---|---|---|---|---|---|
| **DATA TRANSFER INSTRUCTIONS** | | | | | |
| MOV | Rd, Rr | Move between Registers | Rd ← Rr | None | 1 |
| LDI | Rd, K | Load Immediate | Rd ← K | None | 1 |
| LD | Rd, X | Load Indirect | Rd ← (X) | None | 2 |
| LD | Rd, X+ | Load Indirect and Post-inc. | Rd ← (X), X ← X + 1 | None | 2 |
| LD | Rd, -X | Load Indirect and Pre-dec. | X ← X – 1, Rd ← (X) | None | 2 |
| LD | Rd, Y | Load Indirect | Rd ← (Y) | None | 2 |
| LD | Rd, Y+ | Load Indirect and Post-inc. | Rd ← (Y), Y ← Y + 1 | None | 2 |
| LD | Rd, -Y | Load Indirect and Pre-dec. | Y ← Y – 1, Rd ← (Y) | None | 2 |
| LDD | Rd,Y+q | Load Indirect with Displacement | Rd ← (Y + q) | None | 2 |
| LD | Rd, Z | Load Indirect | Rd ← (Z) | None | 2 |
| LD | Rd, Z+ | Load Indirect and Post-inc. | Rd ← (Z), Z ← Z + 1 | None | 2 |
| LD | Rd, -Z | Load Indirect and Pre-dec. | Z ← Z - 1, Rd ← (Z) | None | 2 |
| LDD | Rd, Z+q | Load Indirect with Displacement | Rd ← (Z + q) | None | 2 |
| LDS | Rd, k | Load Direct from SRAM | Rd ← (k) | None | 2 |
| ST | X, Rr | Store Indirect | (X) ← Rr | None | 2 |
| ST | X+, Rr | Store Indirect and Post-inc. | (X) ← Rr, X ← X + 1 | None | 2 |
| ST | -X, Rr | Store Indirect and Pre-dec. | X ← X - 1, (X) ← Rr | None | 2 |
| ST | Y, Rr | Store Indirect | (Y) ← Rr | None | 2 |
| ST | Y+, Rr | Store Indirect and Post-inc. | (Y) ← Rr, Y ← Y + 1 | None | 2 |
| ST | -Y, Rr | Store Indirect and Pre-dec. | Y ← Y - 1, (Y) ← Rr | None | 2 |
| STD | Y+q, Rr | Store Indirect with Displacement | (Y + q) ← Rr | None | 2 |
| ST | Z, Rr | Store Indirect | (Z) ← Rr | None | 2 |
| ST | Z+, Rr | Store Indirect and Post-inc. | (Z) ← Rr, Z ← Z + 1 | None | 2 |
| ST | -Z, Rr | Store Indirect and Pre-dec. | Z ← Z - 1, (Z) ← Rr | None | 2 |
| STD | Z+q, Rr | Store Indirect with Displacement | (Z + q) ← Rr | None | 2 |
| STS | k, Rr | Store Direct to SRAM | (k) ← Rr | None | 2 |
| LPM | | Load Program Memory | R0 ← (Z) | None | 3 |
| IN | Rd, P | In Port | Rd ← P | None | 1 |
| OUT | P, Rr | Out Port | P ← Rr | None | 1 |
| PUSH | Rr | Push Register on Stack | STACK ← Rr | None | 2 |
| POP | Rd | Pop Register from Stack | Rd ← STACK | None | 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 Two's Complement Overflow | V ← 1 | V | 1 |
| CLV | | Clear Two's 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 |
| NOP | | No Operation | | None | 1 |
| SLEEP | | Sleep | (see specific descr. for Sleep function) | None | 1 |
| WDR | | Watchdog Reset | (see specific descr. for WDR/timer) | None | 1 |

## Packaging Information

### 8P3

8P3, 8-lead, Plastic Dual Inline
Package (PDIP), 0.300" Wide.
Dimensions in Millimeters and (Inches)*
JEDEC STANDARD MS-001 BA




*Controlling dimension: Inches

REV. A 04/11/2001