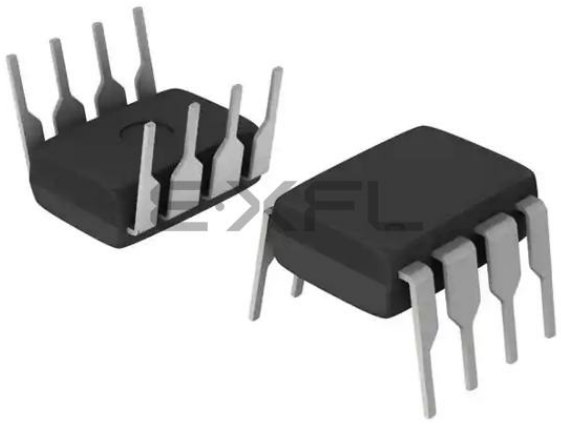


Welcome to E-XFL.COM

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	10MHz
Connectivity	SPI
Peripherals	Brown-out Detect/Reset, POR, WDT
Number of I/O	5
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)	4V ~ 6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C
Mounting Type	Through Hole
Package / Case	8-DIP (0.300", 7.62mm)
Supplier Device Package	8-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at90s2343-10pi

Pin Descriptions
AT90S/LS2343

VCC Supply voltage pin.

GND Ground pin.

Port B (PB4..PB0) Port B is a 5-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.

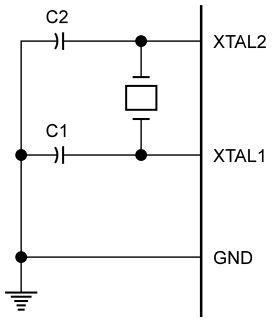
$\overline{\text{RESET}}$ Reset input. An external reset is generated by a low level on the $\overline{\text{RESET}}$ pin. Reset pulses longer than 50 ns will generate a reset, even if the clock is not running. Shorter pulses are not guaranteed to generate a reset.

CLOCK Clock signal input in external clock mode.

Clock Options

Crystal Oscillator The AT90S/LS2323 contains an inverting amplifier that can be configured for use as an On-chip oscillator, as shown in Figure 3. XTAL1 and XTAL2 are input and output respectively. Either a quartz crystal or a ceramic resonator may be used. It is recommended that the AT90S/LS2343 be used if an external clock source is used, since this gives an extra I/O pin.

Figure 3. Oscillator Connection

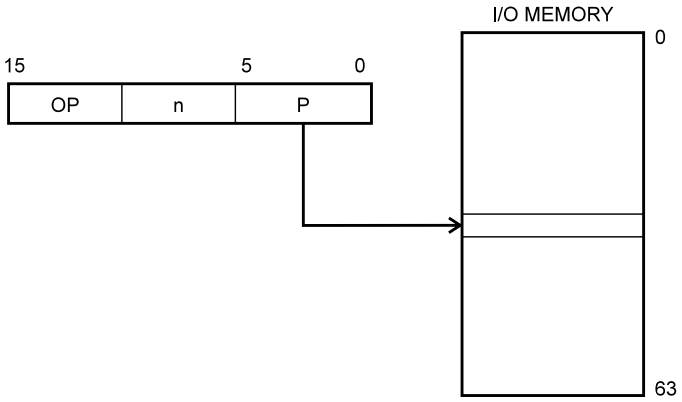


External Clock The AT90S/LS2343 can be clocked by an external clock signal, as shown in Figure 4, or by the On-chip RC oscillator. This RC oscillator runs at a nominal frequency of 1 MHz (V_{CC} = 5V). A fuse bit (RCEN) in the Flash memory selects the On-chip RC oscillator as the clock source when programmed (“0”). The AT90S/LS2343 is shipped with this bit programmed. The AT90S/LS2343 is recommended if an external clock source is used, because this gives an extra I/O pin.

The AT90S/LS2323 can be clocked by an external clock as well, as shown in Figure 4. No fuse bit selects the clock source for AT90S/LS2323.

I/O Direct

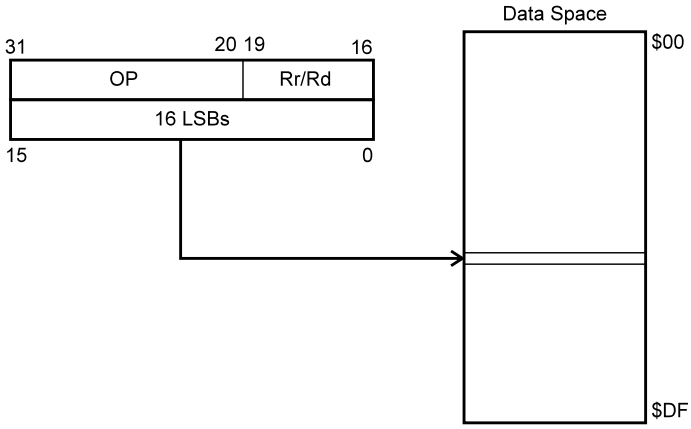
Figure 12. I/O Direct Addressing



Operand address is contained in six bits of the instruction word. n is the destination or source register address.

Data Direct

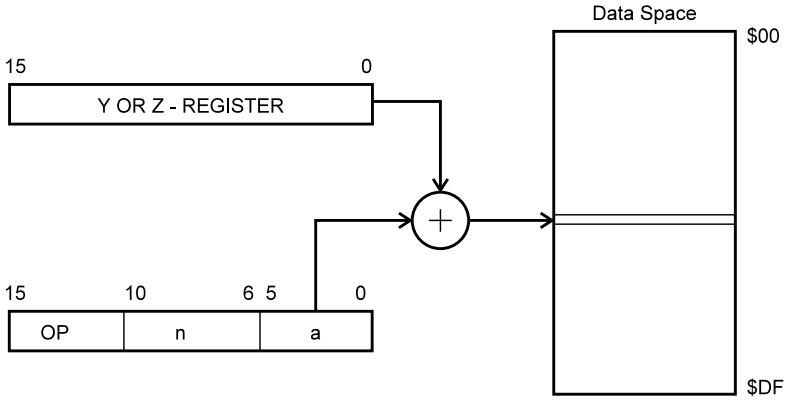
Figure 13. Direct Data Addressing



A 16-bit data address is contained in the 16 LSBs of a 2-word instruction. Rd/Rr specify the destination or source register.

Data Indirect with Displacement

Figure 14. Data Indirect with Displacement



Operand address is the result of the Y- or Z-register contents added to the address contained in six bits of the instruction word.

Figure 25. MCU Start-up, $\overline{\text{RESET}}$ Tied to V_{CC}.

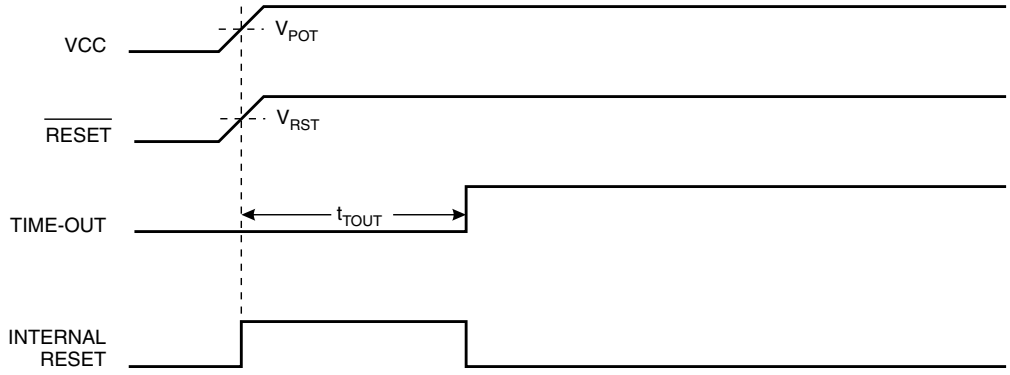
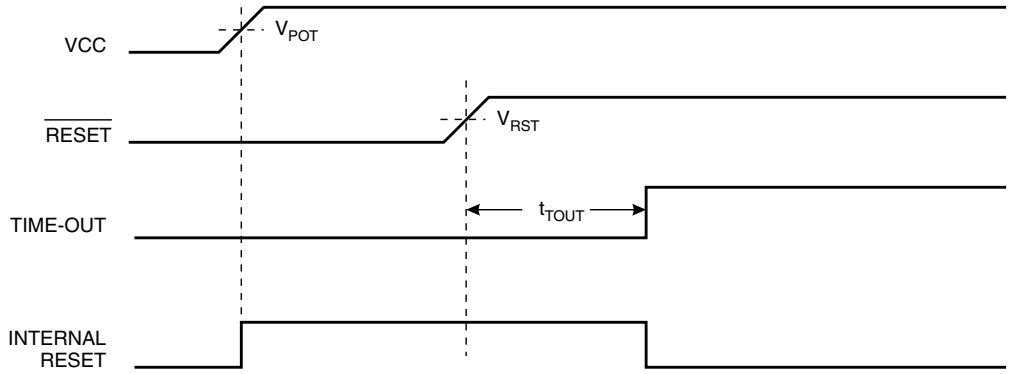


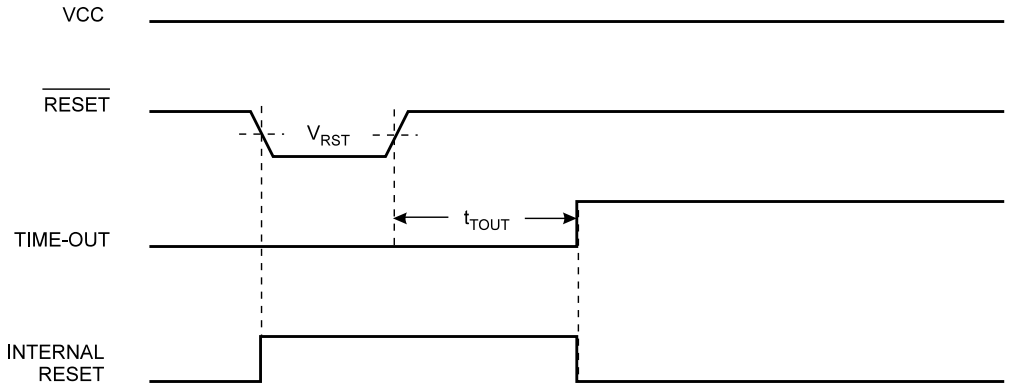
Figure 26. MCU Start-up, $\overline{\text{RESET}}$ Controlled Externally



External Reset

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. When the applied signal reaches the Reset Threshold Voltage (V_{RST}) on its positive edge, the delay timer starts the MCU after the Time-out period t_{TOUT} has expired.

Figure 27. External Reset during Operation



The Stop condition provides a Timer Enable/Disable function. The CK down divided modes are scaled directly from the CK oscillator clock. If the external pin modes are used for Timer/Counter0, transitions on PB2/(T0) will clock the counter even if the pin is configured as an output. This feature can give the user software control of the counting.

Timer/Counter0 – TCNT0

Bit	7	6	5	4	3	2	1	0	
$32 ($52)	MSB							LSB	TCNT0
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	

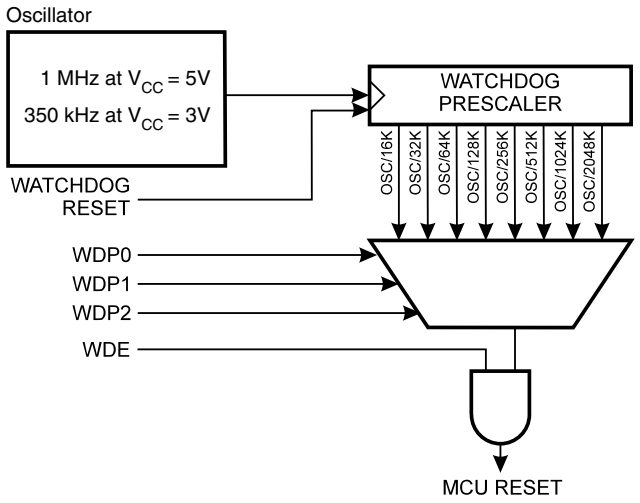
The Timer/Counter0 is realized as an up-counter with read and write access. If the Timer/Counter0 is written and a clock source is present, the Timer/Counter0 continues counting in the timer clock cycle following the write operation.

Watchdog Timer

The Watchdog Timer is clocked from a separate On-chip oscillator. By controlling the Watchdog Timer prescaler, the Watchdog reset interval can be adjusted as shown in Table 11. See characterization data for typical values at other V_{CC} levels. The WDR (Watchdog Reset) instruction resets the Watchdog Timer. Eight different clock cycle periods can be selected to determine the reset period. If the reset period expires without another Watchdog reset, the AT90S2323/2343 resets and executes from the reset vector. For timing details on the Watchdog reset, refer to page 23.

To prevent unintentional disabling of the Watchdog, a special turn-off sequence must be followed when the Watchdog is disabled. Refer to the description of the Watchdog Timer Control Register for details.

Figure 31. Watchdog Timer



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.

Prevent EEPROM Corruption

During periods of low V_{CC}, the EEPROM data can be corrupted because the supply voltage is too low for the CPU and the EEPROM to operate properly. These issues are the same as for board level systems using the EEPROM and the same design solutions should be applied.

An EEPROM data corruption can be caused by two situations when the voltage is too low. First, a regular write sequence to the EEPROM requires a minimum voltage to operate correctly. Secondly, the CPU itself can execute instructions incorrectly, if the supply voltage for executing instructions is too low.

EEPROM data corruption can easily be avoided by following these design recommendations (one is sufficient):

1. Keep the AVR RESET active (low) during periods of insufficient power supply voltage. This is best done by an external low V_{CC} Reset Protection circuit, often referred to as a Brown-out Detector (BOD). Please refer to application note AVR 180 for design considerations regarding power-on reset and low-voltage detection.
2. Keep the AVR core in Power-down Sleep mode during periods of low V_{CC}. This will prevent the CPU from attempting to decode and execute instructions, effectively protecting the EEPROM registers from unintentional writes.
3. Store constants in Flash memory if the ability to change memory contents from software is not required. Flash memory cannot be updated by the CPU and will not be subject to corruption.

Port B Input Pins Address – PINB

Bit	7	6	5	4	3	2	1	0	
$16 ($36)	–	–	–	PINB4	PINB3	PINB2	PINB1	PINB0	PINB
Read/Write	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	N/A	N/A	N/A	N/A	N/A	

The Port B Input Pins address (PINB) is not a register and this address enables access to the physical value on each Port B pin. When reading PORTB, the Port B Data Latch is read and when reading PINB, the logical values present on the pins are read.

Port B as General Digital I/O

All pins in port B have equal functionality when used as digital I/O pins.

PBn, general I/O pin: The DDBn bit in the DDRB register selects the direction of this pin, if DDBn is set (one), PBn is configured as an output pin. If DDBn is cleared (zero), PBn is configured as an input pin. If PORTBn is set (one) when the pin is configured as an input pin, the MOS pull-up resistor is activated. To switch the pull-up resistor off, the PORTBn has to be cleared (zero) or the pin has to be configured as an output pin. The port pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Table 13. DDBn Effects on Port B Pins

DDBn	PORTBn	I/O	Pull-up	Comment
0	0	Input	No	Tri-state (high-Z)
0	1	Input	Yes	PBn will source current if ext. pulled low
1	0	Output	No	Push-pull Zero Output
1	1	Output	No	Push-pull One Output

Alternate Functions of Port B

The alternate pin functions of Port B are as follows:

- **CLOCK – Port B, Bit 3**

Clock input: AT90S/LS2343 only. When the RCEN fuse is programmed and the device runs from the internal RC oscillator, this pin is a general I/O pin. When the RCEN fuse is unprogrammed, an external clock source must be connected to CLOCK.

- **SCK/T0 – Port B, Bit 2**

In Serial Programming mode, this bit serves as the serial clock input, SCK.

During normal operation, this pin can serve as the external counter clock input. See the timer/counter description for further details. If external timer/counter clocking is selected, activity on this pin will clock the counter even if it is configured as an output.

- **MISO/INT0 – Port B, Bit 1**

In Serial Programming mode, this bit serves as the serial data output, MISO.

During normal operation, this pin can serve as the external interrupt0 input. See the interrupt description for details on how to enable this interrupt. Note that activity on this pin will trigger the interrupt even if the pin is configured as an output.

- **MOSI – Port B, Bit 0**

In Serial Programming mode, this pin serves as the serial data input, MOSI.

Table 16. High-voltage Serial Programming Instruction Set

Instruction		Instruction Format				Operation Remarks
		Instr.1	Instr.2	Instr.3	Instr.4	
Chip Erase	PB0 PB1 PB2	0_1000_0000_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_0100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0100_1100_00 x_xxxx_xxxx_xx	Wait t_{WLWH_CE} after Instr.3 for the Chip Erase cycle to finish.
Write Flash High and Low Address	PB0 PB1 PB2	0_0001_0000_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_0000_00**aa**_00 0_0001_1100_00 x_xxxx_xxxx_xx	0_**bbbb_bbbb**_00 0_0000_1100_00 x_xxxx_xxxx_xx		Repeat Instr.2 for a new 256-byte page. Repeat Instr.3 for each new address.
Write Flash Low Byte	PB0 PB1 PB2	0_**iiii_iiii**_00 0_0010_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_0100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1100_00 0_0000_0000_00		Wait after Instr.3 until PB2 goes high. Repeat Instr.1, Instr. 2 and Instr.3 for each new address.
Write Flash High Byte	PB0 PB1 PB2	0_**iiii_iiii**_00 0_0010_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0111_0100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0111_1100_00 0_0000_0000_00		Wait after Instr.3 until PB2 goes high. Repeat Instr.1, Instr. 2 and Instr.3 for each new address.
Read Flash High and Low Address	PB0 PB1 PB2	0_0000_0010_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_0000_00**aa**_00 0_0001_1100_00 x_xxxx_xxxx_xx	0_**bbbb_bbbb**_00 0_0000_1100_00 x_xxxx_xxxx_xx		Repeat Instr.2 and Instr.3 for each new address.
Read Flash Low Byte	PB0 PB1 PB2	0_0000_0000_00 0_0110_1000_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1100_00 **o_oooo_ooo**x_xx			Repeat Instr.1 and Instr.2 for each new address.
Read Flash High Byte	PB0 PB1P B2	0_0000_0000_00 0_0111_1000_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0111_1100_00 **o_oooo_ooo**x_xx			Repeat Instr.1 and Instr.2 for each new address.
Write EEPROM Low Address	PB0 PB1 PB2	0_0001_0001_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_0**bbb_bbbb**_00 0_0000_1100_00 x_xxxx_xxxx_xx			Repeat Instr.2 for each new address.
Write EEPROM Byte	PB0 PB1 PB2	0_**iiii_iiii**_00 0_0010_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_0100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1100_00 0_0000_0000_00		Wait after Instr.3 until PB2 goes high
Read EEPROM Low Address	PB0 PB1 PB2	0_0000_0011_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_0**bbb_bbbb**_00 0_0000_1100_00 x_xxxx_xxxx_xx			Repeat Instr.2 for each new address.
Read EEPROM Byte	PB0 PB1 PB2	0_0000_0000_00 0_0110_1000_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1100_00 **o_oooo_ooo**x_xx			Repeat Instr.2 for each new address
Write Fuse Bits (AT90S/ LS2323)	PB0 PB1 PB2	0_0100_0000_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_11**S**1_111**F**_00 0_0010_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_0100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1100_00 x_xxxx_xxxx_xx	Wait t_{WLWH_PFB} after Instr.3 for the Write Fuse bits cycle to finish. Set **S**,**F** = “0” to program, “1” to unprogram.
Write Fuse Bits (AT90S/ LS2343)	PB0 PB1 PB2	0_0100_0000_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_11**S**1_111**R**_00 0_0010_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_0100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1100_00 x_xxxx_xxxx_xx	Wait t_{WLWH_PFB} after Instr.3 for the Write Fuse bits cycle to finish. Set **S**,**R** = “0” to program, “1” to unprogram.
Write Lock Bits	PB0 PB1 PB2	0_0010_0000_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_1111_1***21***1_00 0_0010_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_0100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1100_00 0_0000_0000_00	Wait after Instr.4 until PB2 goes high. Write ***2***, ***1*** = “0” to program the Lock bit.

Table 16. High-voltage Serial Programming Instruction Set (Continued)

Instruction		Instruction Format				Operation Remarks
		Instr.1	Instr.2	Instr.3	Instr.4	
Read Fuse and Lock Bits (AT90S/ LS2323)	PB0 PB1 PB2	0_0000_0100_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0111_1000_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0111_1100_00 ***1_2*****S**xx_xx**R**x_xx		Reading ***1***, ***2***, **S**, **R** = "0" means the Fuse/Lock bit is programmed.
Read Fuse and Lock Bits (AT90S/ LS2343)	PB0 PB1 PB2	0_0000_0100_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0111_1000_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0111_1100_00 ***1_2*****S**xx_xx**R**x_xx		Reading ***1***, ***2***, **S**, **R** = "0" means the Fuse/Lock bit is programmed.
Read Signature Bytes	PB0 PB1 PB2	0_0000_1000_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_0000_00**bb**_00 0_0000_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1000_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1100_00 **o_oooo_ooo**x_xx	Repeat Instr.2 - Instr.4 for each signature byte address.

Note: **a** = address high bits
b = address low bits
i = data in
o = data out
x = don't care
1 = Lock Bit1
2 = Lock Bit2
F = FSTRT Fuse
R = RCEN Fuse
S = SPIEN Fuse

Table 19. Low-voltage Serial Programming Instruction Set AT90S2323/2343

Instruction	Instruction Format				Operation
	Byte 1	Byte 2	Byte 3	Byte 4	
Programming Enable	1010 1100	0101 0011	xxxx xxxx	xxxx xxxx	Enable Serial programming while $\overline{\text{RESET}}$ is low.
Chip Erase	1010 1100	100x xxxx	xxxx xxxx	xxxx xxxx	Chip erase both Flash and EEPROM memory arrays.
Read Program Memory	0010 **H**000	0000 00**aa**	**bbbb bbbb**	**oooo oooo**	Read **H** (high or low) data **o** from program memory at word address **a**:**b**.
Write Program Memory	0100 **H**000	0000 00**aa**	**bbbb bbbb**	**iiii iiii**	Write **H** (high or low) data **i** to program memory at word address **a**:**b**.
Read EEPROM Memory	1010 0000	0000 0000	x**bbb bbbb**	**oooo oooo**	Read data **o** from EEPROM memory at address **b**.
Write EEPROM Memory	1100 0000	0000 0000	x**bbb bbbb**	**iiii iiii**	Write data **i** to EEPROM memory at address **b**.
Read Lock and Fuse Bits (AT90S/LS2323)	0101 1000	xxxx xxxx	xxxx xxxx	***12S***x xxx**F**	Read Lock and Fuse bits. “0” = programmed, “1” = unprogrammed
Read Lock and Fuse Bits (AT90S/LS2343)	0101 1000	xxxx xxxx	xxxx xxxx	***12S***x xxx**R**	Read Lock and Fuse bits. “0” = programmed, “1” = unprogrammed
Write Lock Bits	1010 1100	1111 1***21***1	xxxx xxxx	xxxx xxxx	Write Lock bits. Set bits ***1***,***2*** = “0” to program Lock bits.
Write FSTRT Bit (AT90S/LS2323)	1010 1100	1011 111**F**	xxxx xxxx	xxxx xxxx	Write FSTRT fuse. Set bit **F** = “0” to program, “1” to unprogram.(2)
Write RCEN Bit (AT90S/LS2343)	1010 1100	1011 111**R**	xxxx xxxx	xxxx xxxx	Write RCEN Fuse. Set bit **R** = ‘0’ to program, ‘1’ to unprogram.(2)
Read Signature Bytes	0011 0000	xxxx xxxx	xxxx xx**bb**	**oooo oooo**	Read signature byte **o** from address **b**.(3)

Notes: 1. **a** = address high bits
b = address low bits
H = 0 – Low byte, 1 – High byte
o = data out
i = data in
x = don’t care
1 = lock bit 1
2 = lock bit 2
F = FSTRT Fuse
R = RCEN Fuse
S = SPIEN Fuse
2. When the state of the RCEN/FSTRT bit is changed, the device must be power cycled for the changes to have any effect.
3. The signature bytes are not readable in Lock mode 3, i.e., both Lock bits programmed.

Figure 42. Idle Supply Current vs. Frequency

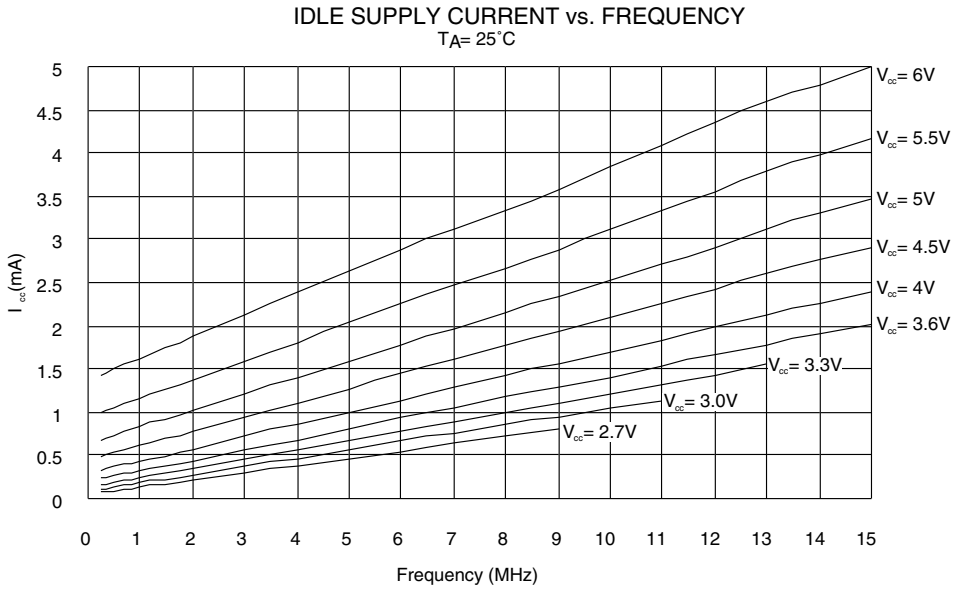


Figure 43. Idle Supply Current vs. V_{CC}

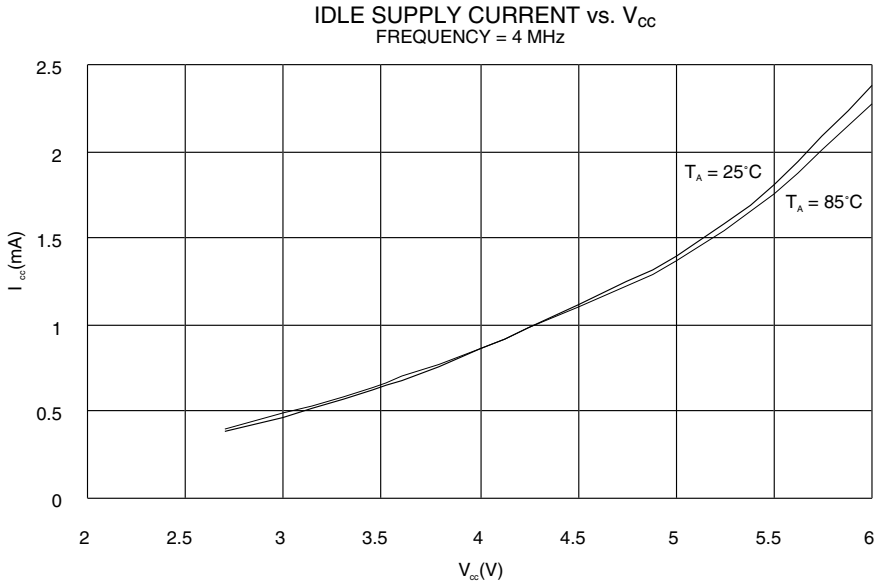


Figure 44. Idle Supply Current vs. V_{CC}

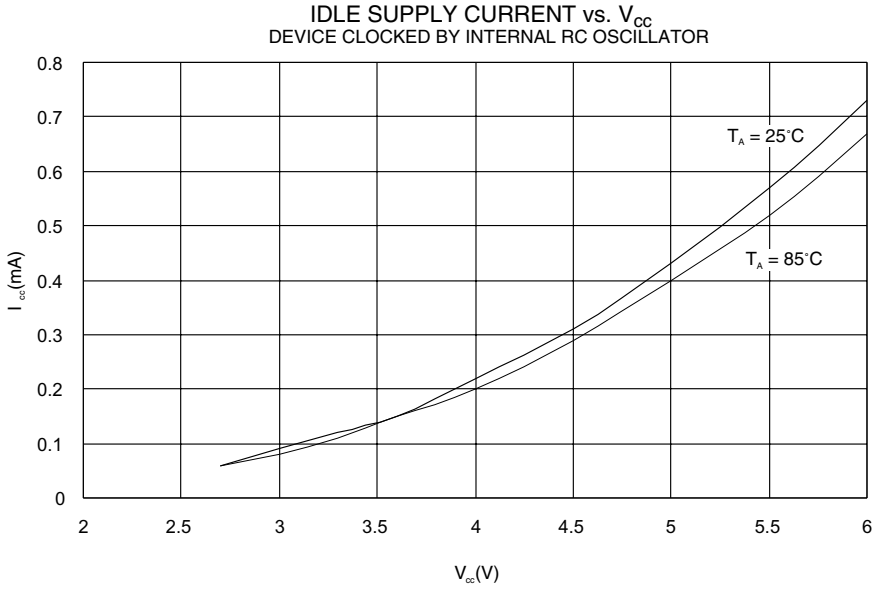
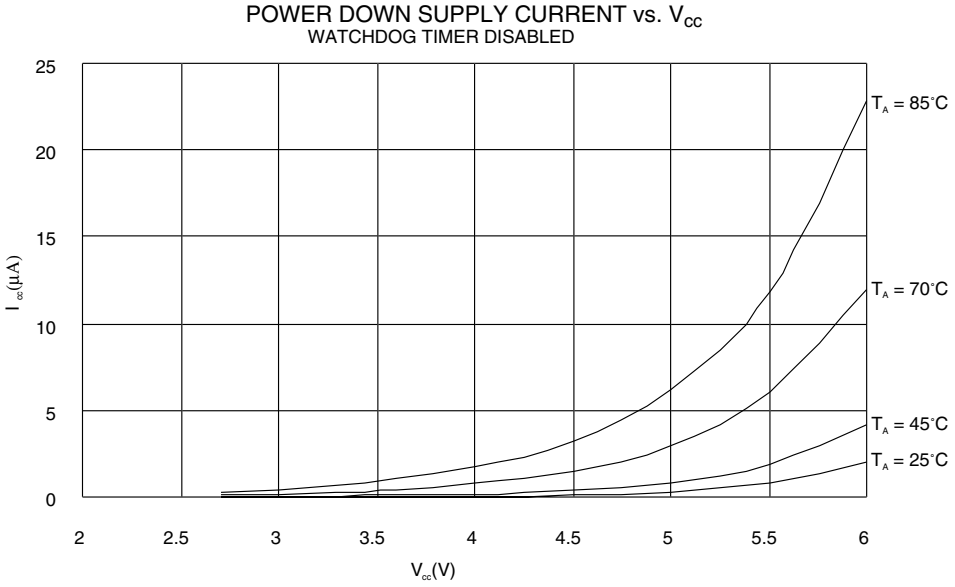


Figure 45. Power-down Supply Current vs. V_{CC}



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

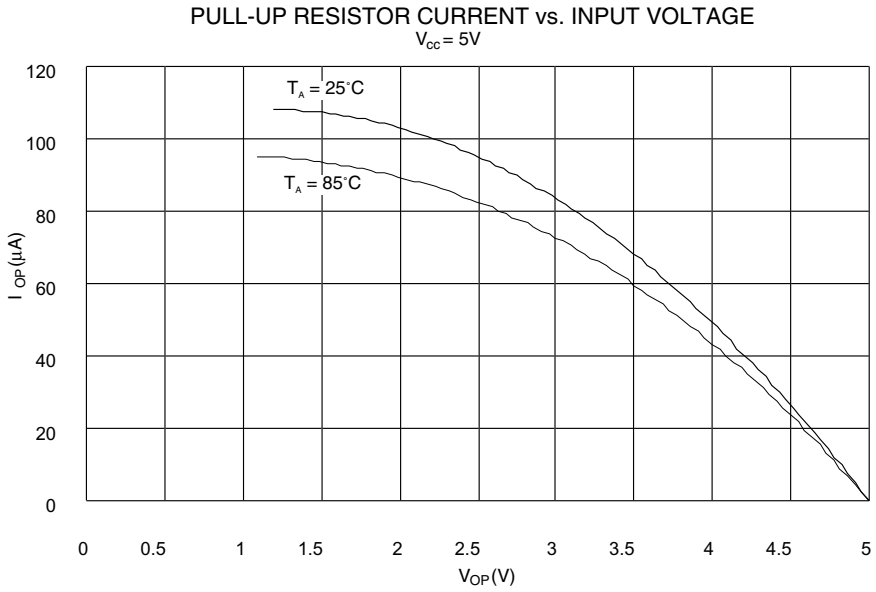


Figure 49. Pull-up Resistor Current vs. Input Voltage

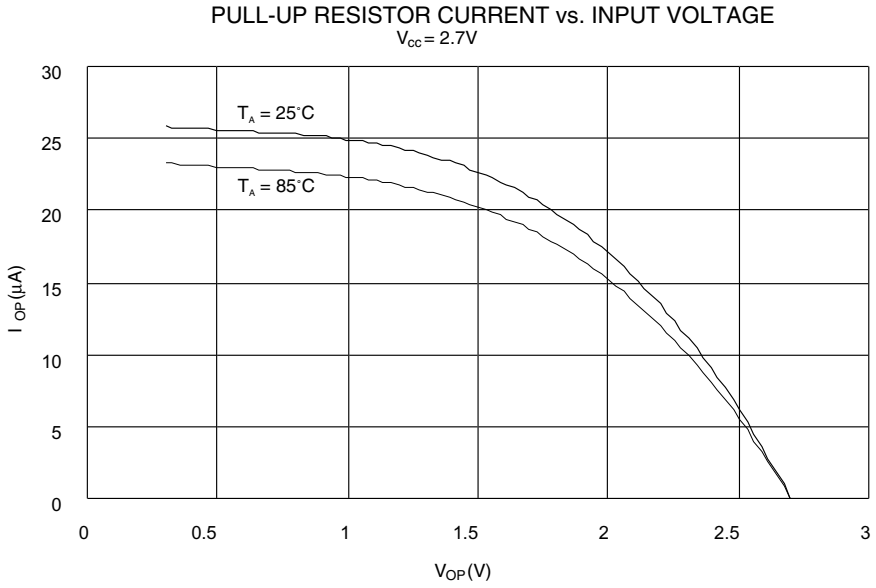


Figure 52. I/O Pin Sink Current vs. Output Voltage

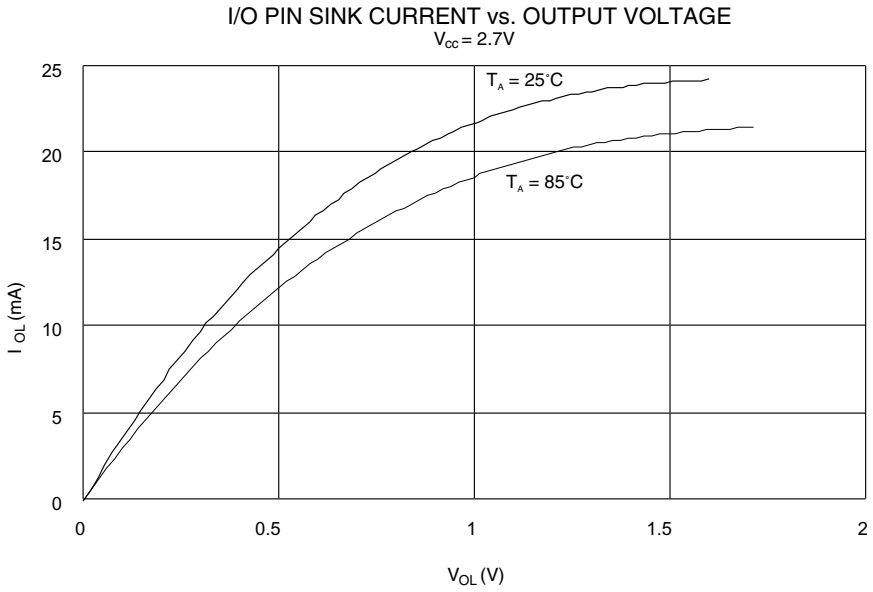


Figure 53. I/O Pin Source Current vs. Output voltage

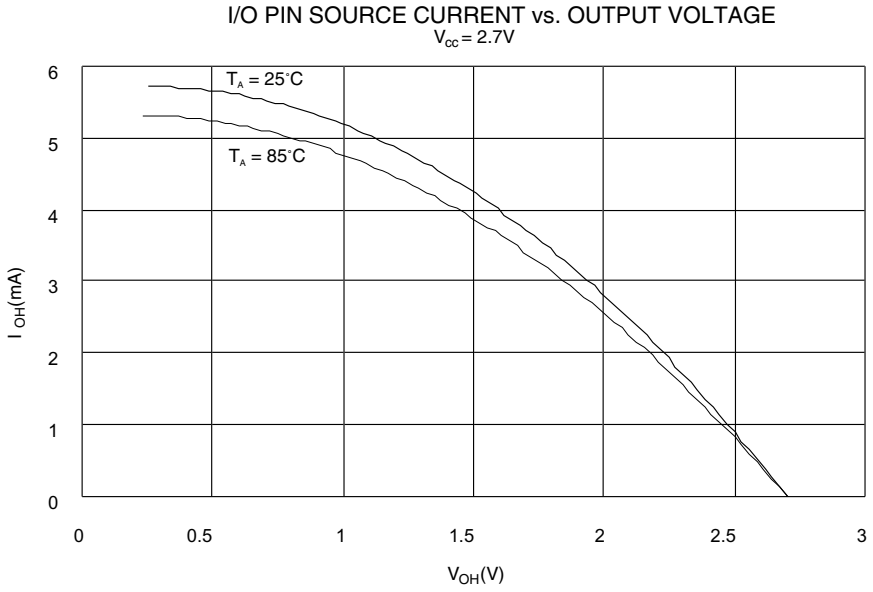


Figure 54. I/O Pin Input Threshold Voltage vs. V_{CC}

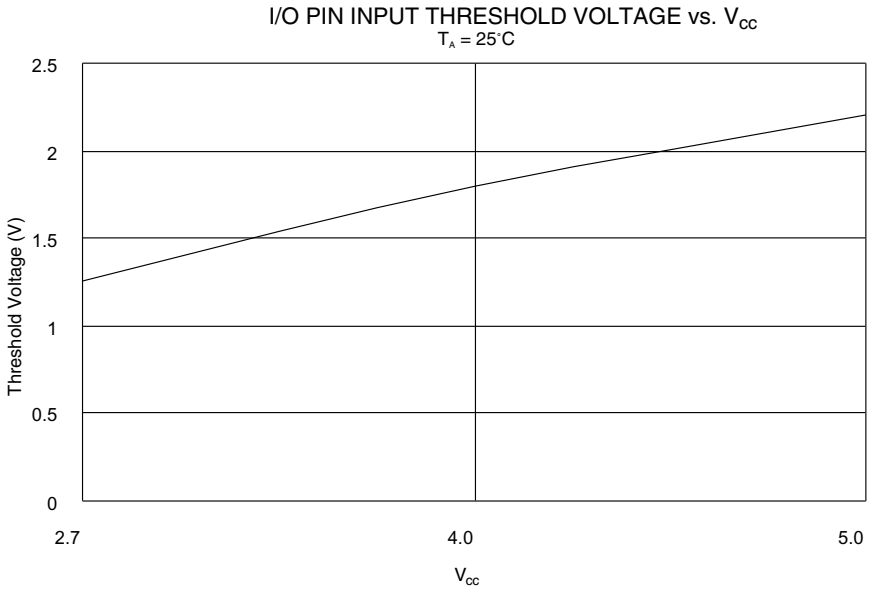
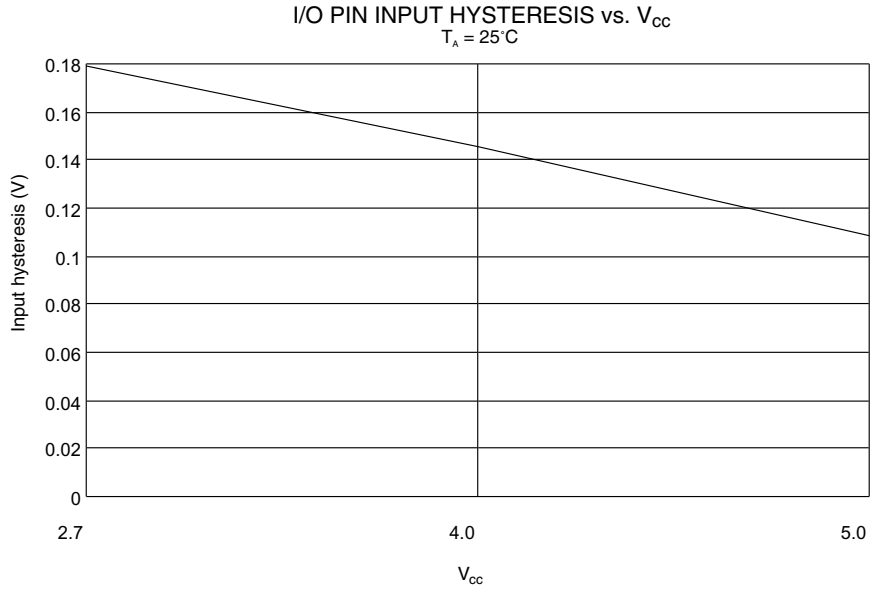


Figure 55. I/O Pin Input Hysteresis vs. V_{CC}



AT90S2323/2343 Register Summary

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
$3F ($5F)	SREG	I	T	H	S	V	N	Z	C	page 18
$3E ($5E)	Reserved									
$3D ($5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	page 19
$3C ($5C)	Reserved									
$3B ($5B)	GIMSK	-	INT0	-	-	-	-	-	-	page 24
$3A ($5A)	GIFR	-	INTF0							page 25
$39 ($59)	TIMSK	-	-	-	-	-	-	TOIE0	-	page 25
$38 ($58)	TIFR	-	-	-	-	-	-	TOV0	-	page 25
$37 ($57)	Reserved									
$36 ($56)	Reserved									
$35 ($55)	MCUCR	-	-	SE	SM	-	-	ISC01	ISC00	page 26
$34 ($54)	MCUSR	-	-	-	-	-	-	EXTRF	PORF	page 23
$33 ($53)	TCCR0	-	-	-	-	-	CS02	CS01	CS00	page 29
$32 ($52)	TCNT0	Timer/Counter0 (8 Bits)								page 30
$31 ($51)	Reserved									
$30 ($50)	Reserved									
$2F ($4F)	Reserved									
$2E ($4E)	Reserved									
$2D ($4D)	Reserved									
$2C ($4C)	Reserved									
$2B ($4B)	Reserved									
$2A ($4A)	Reserved									
$29 ($49)	Reserved									
$28 ($48)	Reserved									
$27 ($47)	Reserved									
$26 ($46)	Reserved									
$25 ($45)	Reserved									
$24 ($44)	Reserved									
$23 ($43)	Reserved									
$22 ($42)	Reserved									
$21 ($41)	WDTCR	-	-	-	WDTOE	WDE	WDP2	WDP1	WDP0	page 31
$20 ($40)	Reserved									
$1F ($3F)	Reserved									
$1E ($3E)	EEAR	-	EEPROM Address Register							page 32
$1D ($3D)	EEDR	EEPROM Data Register								page 32
$1C ($3C)	EECR	-	-	-	-	-	EEMWE	EEWE	EERE	page 33
$1B ($3B)	Reserved									
$1A ($3A)	Reserved									
$19 ($39)	Reserved									
$18 ($38)	PORTB	-	-	-	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	page 35
$17 ($37)	DDRB	-	-	-	DDB4	DDB3	DDB2	DDB1	DDB0	page 35
$16 ($36)	PINB	-	-	-	PINB4	PINB3	PINB2	PINB1	PINB0	page 36
$15 ($35)	Reserved									
…	Reserved									
$00 ($20)	Reserved									

Note: 1. For compatibility with future devices, reserved bits should be written to zero if accessed. Reserved I/O memory addresses should never be written.

2. Some of the status flags are cleared by writing a logical “1” to them. Note that the CBI and SBI instructions will operate on all bits in the I/O register, writing a one back into any flag read as set, thus clearing the flag. The CBI and SBI instructions work with registers $00 to $1F only.

Instruction Set Summary

Mnemonic	Operands	Description	Operation	Flags	# Clocks
ARITHMETIC AND LOGIC INSTRUCTIONS					
ADD	Rd, Rr	Add Two Registers	Rd ← Rd + Rr	Z,C,N,V,H	1
ADC	Rd, Rr	Add with Carry Two Registers	Rd ← Rd + Rr + C	Z,C,N,V,H	1
ADIW	Rdl, K	Add Immediate to Word	Rdh:Rdl ← Rdh:Rdl + K	Z,C,N,V,S	2
SUB	Rd, Rr	Subtract Two Registers	Rd ← Rd – Rr	Z,C,N,V,H	1
SUBI	Rd, K	Subtract Constant from Register	Rd ← Rd – K	Z,C,N,V,H	1
SBIW	Rdl, K	Subtract Immediate from Word	Rdh:Rdl ← Rdh:Rdl – K	Z,C,N,V,S	2
SBC	Rd, Rr	Subtract with Carry Two Registers	Rd ← Rd – Rr – C	Z,C,N,V,H	1
SBCI	Rd, K	Subtract with Carry Constant from Reg.	Rd ← Rd – K – C	Z,C,N,V,H	1
AND	Rd, Rr	Logical AND Registers	Rd ← Rd • Rr	Z,N,V	1
ANDI	Rd, K	Logical AND Register and Constant	Rd ← Rd • K	Z,N,V	1
OR	Rd, Rr	Logical OR Registers	Rd ← Rd v Rr	Z,N,V	1
ORI	Rd, K	Logical OR Register and Constant	Rd ← Rd v K	Z,N,V	1
EOR	Rd, Rr	Exclusive OR Registers	Rd ← Rd ⊕ Rr	Z,N,V	1
COM	Rd	One's Complement	Rd ← $FF – Rd	Z,C,N,V	1
NEG	Rd	Two's Complement	Rd ← $00 – Rd	Z,C,N,V,H	1
SBR	Rd, K	Set Bit(s) in Register	Rd ← Rd v K	Z,N,V	1
CBR	Rd, K	Clear Bit(s) in Register	Rd ← Rd • ($FF – K)	Z,N,V	1
INC	Rd	Increment	Rd ← Rd + 1	Z,N,V	1
DEC	Rd	Decrement	Rd ← Rd – 1	Z,N,V	1
TST	Rd	Test for Zero or Minus	Rd ← Rd • Rd	Z,N,V	1
CLR	Rd	Clear Register	Rd ← Rd ⊕ Rd	Z,N,V	1
SER	Rd	Set Register	Rd ← $FF	None	1
BRANCH INSTRUCTIONS					
RJMP	k	Relative Jump	PC ← PC + k + 1	None	2
IJMP		Indirect Jump to (Z)	PC ← Z	None	2
RCALL	k	Relative Subroutine Call	PC ← PC + k + 1	None	3
ICALL		Indirect Call to (Z)	PC ← Z	None	3
RET		Subroutine Return	PC ← STACK	None	4
RETI		Interrupt Return	PC ← STACK	I	4
CPSE	Rd, Rr	Compare, Skip if Equal	if (Rd = Rr) PC ← PC + 2 or 3	None	1/2/3
CP	Rd, Rr	Compare	Rd – Rr	Z,N,V,C,H	1
CPC	Rd, Rr	Compare with Carry	Rd – Rr – C	Z,N,V,C,H	1
CPI	Rd, K	Compare Register with Immediate	Rd – K	Z,N,V,C,H	1
SBRC	Rr, b	Skip if Bit in Register Cleared	if (Rr(b) = 0) PC ← PC + 2 or 3	None	1/2/3
SBRS	Rr, b	Skip if Bit in Register is Set	if (Rr(b) = 1) PC ← PC + 2 or 3	None	1/2/3
SBIC	P, b	Skip if Bit in I/O Register Cleared	if (P(b) = 0) PC ← PC + 2 or 3	None	1/2/3
SBIS	P, b	Skip if Bit in I/O Register is Set	if (R(b) = 1) PC ← PC + 2 or 3	None	1/2/3
BRBS	s, k	Branch if Status Flag Set	if (SREG(s) = 1) then PC ← PC + k + 1	None	1/2
BRBC	s, k	Branch if Status Flag Cleared	if (SREG(s) = 0) then PC ← PC + k + 1	None	1/2
BREQ	k	Branch if Equal	if (Z = 1) then PC ← PC + k + 1	None	1/2
BRNE	k	Branch if Not Equal	if (Z = 0) then PC ← PC + k + 1	None	1/2
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

Ordering Information

Power Supply	Speed (MHz)	Ordering Code	Package	Operation Range
2.7 - 6.0V	4	AT90LS2323-4PC AT90LS2323-4SC	8P3 8S2	Commercial (0°C to 70°C)
		AT90LS2323-4PI AT90LS2323-4SI	8P3 8S2	Industrial (-40°C to 85°C)
4.0 - 6.0V	10	AT90S2323-10PC AT90S2323-10SC	8P3 8S2	Commercial (0°C to 70°C)
		AT90S2323-10PI AT90S2323-10SI	8P3 8S2	Industrial (-40°C to 85°C)
2.7 - 6.0V	1	AT90LS2343-1PC AT90LS2343-1SC	8P3 8S2	Commercial (0°C to 70°C)
		AT90LS2343-1PI AT90LS2343-1SI	8P3 8S2	Industrial (-40°C to 85°C)
2.7 - 6.0V	4	AT90LS2343-4PC AT90LS2343-4SC	8P3 8S2	Commercial (0°C to 70°C)
		AT90LS2343-4PI AT90LS2343-4SI	8P3 8S2	Industrial (-40°C to 85°C)
4.0 - 6.0V	10	AT90S2343-10PC AT90S2343-10SC	8P3 8S2	Commercial (0°C to 70°C)
		AT90S2343-10PI AT90S2343-10SI	8P3 8S2	Industrial (-40°C to 85°C)

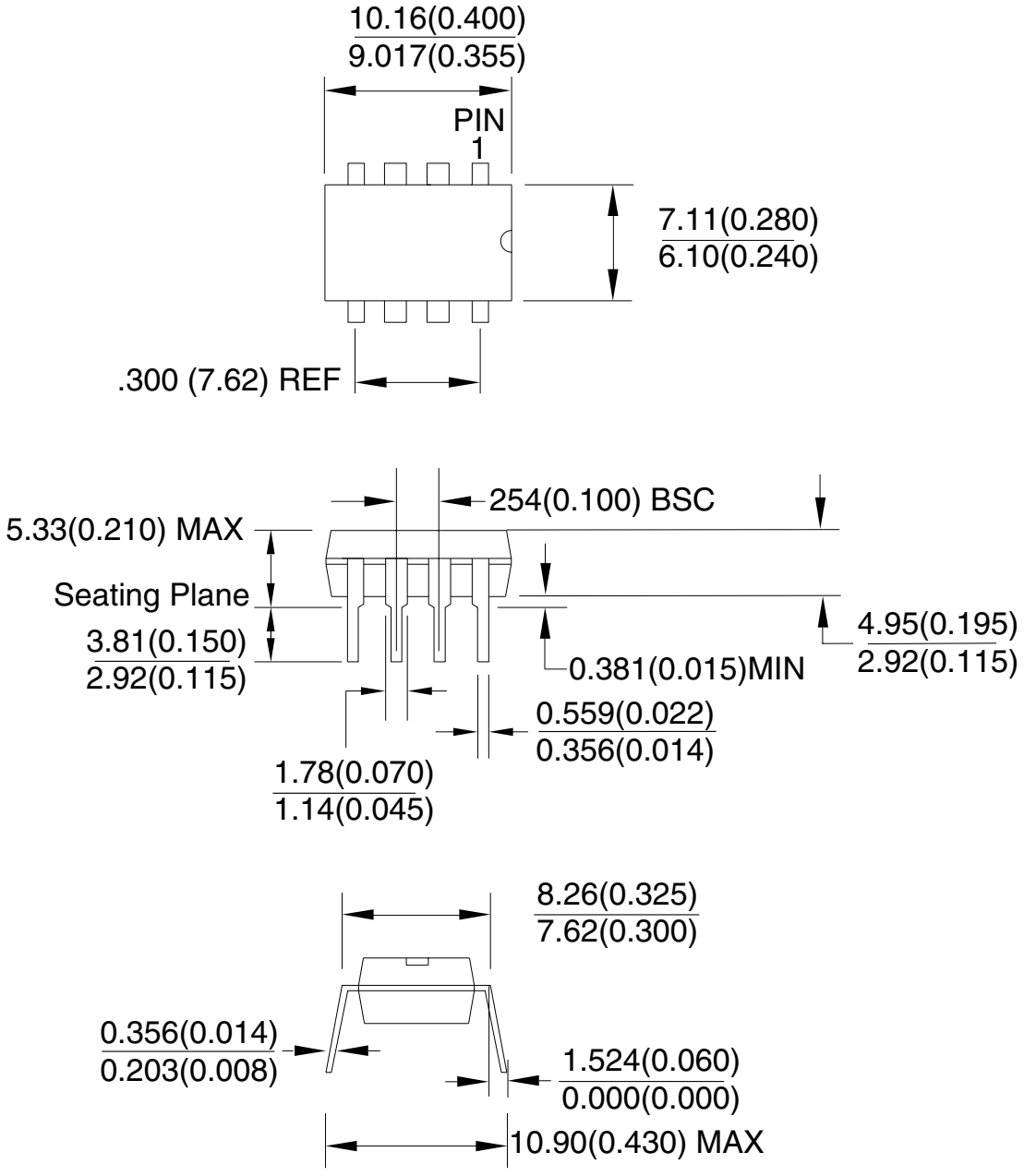
Notes: 1. The speed grade refers to maximum clock rate when using an external crystal or external clock drive. The internal RC oscillator has the same nominal clock frequency for all speed grades.
2. In AT90LS2343-1xx, the internal RC oscillator is selected as default MCU clock source (RCEN fuse is programmed) when the device is shipped from Atmel. In AT90LS2343-4xx and AT90S2343-10xx, the default MCU clock source is the clock input pin (RCEN fuse is unprogrammed). The fuse settings can be changed by high voltage serial programming.

Package Type	
8P3	8-lead, 0.300" Wide, Plastic Dual Inline Package (PDIP)
8S2	8-lead, 0.200" Wide, Plastic Gull Wing Small Outline Package (EIAJ SOIC)

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

Atmel Headquarters

Corporate Headquarters
2325 Orchard Parkway
San Jose, CA 95131
TEL (408) 441-0311
FAX (408) 487-2600

Europe
Atmel SarL
Route des Arsenaux 41
Casa Postale 80
CH-1705 Fribourg
Switzerland
TEL (41) 26-426-5555
FAX (41) 26-426-5500

Asia
Atmel Asia, Ltd.
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimhatsui
East Kowloon
Hong Kong
TEL (852) 2721-9778
FAX (852) 2722-1369

Japan
Atmel Japan K.K.
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
TEL (81) 3-3523-3551
FAX (81) 3-3523-7581

Atmel Product Operations

Atmel Colorado Springs
1150 E. Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906
TEL (719) 576-3300
FAX (719) 540-1759

Atmel Grenoble
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
TEL (33) 4-7658-3000
FAX (33) 4-7658-3480

Atmel Heilbronn
Theresienstrasse 2
POB 3535
D-74025 Heilbronn, Germany
TEL (49) 71 31 67 25 94
FAX (49) 71 31 67 24 23

Atmel Nantes
La Chantrerie
BP 70602
44306 Nantes Cedex 3, France
TEL (33) 0 2 40 18 18 18
FAX (33) 0 2 40 18 19 60

Atmel Rousset
Zone Industrielle
13106 Rousset Cedex, France
TEL (33) 4-4253-6000
FAX (33) 4-4253-6001

Atmel Smart Card ICs
Scottish Enterprise Technology Park
East Kilbride, Scotland G75 0QR
TEL (44) 1355-357-000
FAX (44) 1355-242-743

---

e-mail
literature@atmel.com

Web Site
http://www.atmel.com

BBS
1-(408) 436-4309

1004D–09/01/xM