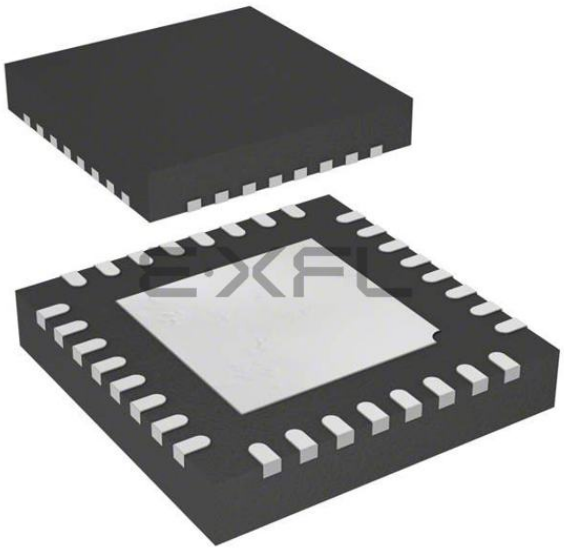


Welcome to E-XFL.COM

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	Active
Core Processor	AVR
Core Size	8-Bit
Speed	16MHz
Connectivity	SPI
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	19
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 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	32-VQFN Exposed Pad
Supplier Device Package	32-QFN (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at90pwm1-16mur

Table 5. Reset and Interrupt Vectors

Vector No.	Program Address	Source	Interrupt Definition
29	0x001C	INT3	External Interrupt Request 3
30	0x001D		
31	0x001E		
32	0x001F	SPM READY	Store Program Memory Ready

Notes: 1. When the BOOTRST Fuse is programmed, the device will jump to the Boot Loader address at reset, see “Boot Loader Support – Read-While-Write Self-Programming” on page 205.
2. When the IVSEL bit in MCUCR is set, Interrupt Vectors will be moved to the start of the Boot Flash Section. The address of each Interrupt Vector will then be the address in this table added to the start address of the Boot Flash Section.

Table 6 shows reset and Interrupt Vectors placement for the various combinations of BOOTRST and IVSEL settings. If the program never enables an interrupt source, the Interrupt Vectors are not used, and regular program code can be placed at these locations. This is also the case if the Reset Vector is in the Application section while the Interrupt Vectors are in the Boot section or vice versa.

Table 6. Reset and Interrupt Vectors Placement in AT90PWM1(1)

BOOTRST	IVSEL	Reset Address	Interrupt Vectors Start Address
1	0	0x000	0x001
1	1	0x000	Boot Reset Address + 0x001
0	0	Boot Reset Address	0x001
0	1	Boot Reset Address	Boot Reset Address + 0x001

Note: 1. The Boot Reset Address is shown in Table 70 on page 218. For the BOOTRST Fuse “1” means unprogrammed while “0” means programmed.

The most typical and general program setup for the Reset and Interrupt Vector Addresses in AT90PWM1 is:

```
Address  Labels Code                 Comments
0x000           rjmp   RESET         ; Reset Handler
0x001           rjmp   PSC2_CAPT     ; PSC2 Capture event Handler
0x002           rjmp   PSC2_EC       ; PSC2 End Cycle Handler
0x003           rjmp   PSC1_CAPT     ; PSC1 Capture event Handler
0x004           rjmp   PSC1_EC       ; PSC1 End Cycle Handler
0x005           rjmp   PSC0_CAPT     ; PSC0 Capture event Handler
0x006           rjmp   PSC0_EC       ; PSC0 End Cycle Handler
0x007           rjmp   ANA_COMP_0    ; Analog Comparator 0 Handler
0x008           rjmp   ANA_COMP_1    ; Analog Comparator 1 Handler
0x009           rjmp   ANA_COMP_2    ; Analog Comparator 2 Handler
0x00A           rjmp   EXT_INT0      ; IRQ0 Handler
0x00B           rjmp   TIM1_CAPT     ; Timer1 Capture Handler
0x00C           rjmp   TIM1_COMPA    ; Timer1 Compare A Handler
0x00D           rjmp   TIM1_COMPB    ; Timer1 Compare B Handler

0x00F           rjmp   TIM1_OVF      ; Timer1 Overflow Handler
```

IVSEL. If IVSEL is not written, interrupts remain disabled for four cycles. The I-bit in the Status Register is unaffected by the automatic disabling.

Note: If Interrupt Vectors are placed in the Boot Loader section and Boot Lock bit BLB02 is programmed, interrupts are disabled while executing from the Application section. If Interrupt Vectors are placed in the Application section and Boot Lock bit BLB12 is programed, interrupts are disabled while executing from the Boot Loader section. Refer to the section “Boot Loader Support – Read-While-Write Self-Programming” on page 205 for details on Boot Lock bits.

- **Bit 0 – IVCE: Interrupt Vector Change Enable**

The IVCE bit must be written to logic one to enable change of the IVSEL bit. IVCE is cleared by hardware four cycles after it is written or when IVSEL is written. Setting the IVCE bit will disable interrupts, as explained in the IVSEL description above. See Code Example below.

Assembly Code Example

```
Move_interrupts:
  ; Enable change of Interrupt Vectors
  ldi  r16, (1<<IVCE)
  out  MCUCR, r16
  ; Move interrupts to Boot Flash section
  ldi  r16, (1<<IVSEL)
  out  MCUCR, r16
  ret
```

C Code Example

```
void Move_interrupts(void)
{
  /* Enable change of Interrupt Vectors */
  MCUCR = (1<<IVCE);
  /* Move interrupts to Boot Flash section */
  MCUCR = (1<<IVSEL);
}
```

11.3.3 Alternate Functions of Port D

The Port D pins with alternate functions are shown in Table 12.

Table 12. Port D Pins Alternate Functions

Port Pin	Alternate Function
PD7	ACMP0 (Analog Comparator 0 Positive Input)
PD6	ADC3 (Analog Input Channel 3) ACMPM reference for analog comparators INT0
PD5	ADC2 (Analog Input Channel 2) ACMP2 (Analog Comparator 2 Positive Input)
PD4	ADC1 (Analog Input Channel 1) ICP1 (Timer 1 input capture) SCK_A (Programming & alternate SPI Clock)
PD3	OC0A (Timer 0 Output Compare A) SS (SPI Slave Select) MOSI_A (Programming & alternate SPI Master Out Slave In)
PD2	PSCIN2 (PSC 2 Digital Input) OC1A (Timer 1 Output Compare A) MISO_A (Programming & alternate Master In SPI Slave Out)
PD1	PSCIN0 (PSC 0 Digital Input) CLKO (System Clock Output)
PD0	PSCOUT00 output SS_A (Alternate SPI Slave Select)

The alternate pin configuration is as follows:

- **ACMP0 – Bit 7**

ACMP0, Analog Comparator 0 Positive Input. Configure the port pin as input with the internal pull-up switched off to avoid the digital port function from interfering with the function of the Analog Comparator.

- **ADC3/ACMPM/$\overline{\text{INT0}}$ – Bit 6**

ADC3, Analog to Digital Converter, input channel 3.

ACMPM, Analog Comparators Negative Input. Configure the port pin as input with the internal pull-up switched off to avoid the digital port function from interfering with the function of the Analog Comparator.

INT0, External Interrupt source 0. This pin can serve as an external interrupt source to the MCU.

- **ADC2/ACMP2 – Bit 5**

ADC2, Analog to Digital Converter, input channel 2.

ACMP2, Analog Comparator 1 Positive Input. Configure the port pin as input with the internal pull-up switched off to avoid the digital port function from interfering with the function of the Analog Comparator.

- **ADC1/ICP1/SCK_A – Bit 4**

ADC1, Analog to Digital Converter, input channel 1.

The definitions in Table 19 are also used extensively throughout the document.

Table 19. Definitions

BOTTOM	The counter reaches the BOTTOM when it becomes 0x00.
MAX	The counter reaches its MAXimum when it becomes 0xFF (decimal 255).
TOP	The counter reaches the TOP when it becomes equal to the highest value in the count sequence. The TOP value can be assigned to be the fixed value 0xFF (MAX) or the value stored in the OCR0A Register. The assignment is dependent on the mode of operation.

14.1.2 Registers

The Timer/Counter (TCNT0) and Output Compare Registers (OCR0A and OCR0B) are 8-bit registers. Interrupt request (abbreviated to Int.Req. in the figure) signals are all visible in the Timer Interrupt Flag Register (TIFR0). All interrupts are individually masked with the Timer Interrupt Mask Register (TIMSK0). TIFR0 and TIMSK0 are not shown in the figure.

The Timer/Counter can be clocked internally, via the prescaler, or by an external clock source on the T0 pin. The Clock Select logic block controls which clock source and edge the Timer/Counter uses to increment (or decrement) its value. The Timer/Counter is inactive when no clock source is selected. The output from the Clock Select logic is referred to as the timer clock (clk_{T0}).

The double buffered Output Compare Registers (OCR0A and OCR0B) are compared with the Timer/Counter value at all times. The result of the compare can be used by the Waveform Generator to generate a PWM or variable frequency output on the Output Compare pins (OC0A and OC0B). See “Using the Output Compare Unit” on page 106. for details. The compare match event will also set the Compare Flag (OCF0A or OCF0B) which can be used to generate an Output Compare interrupt request.

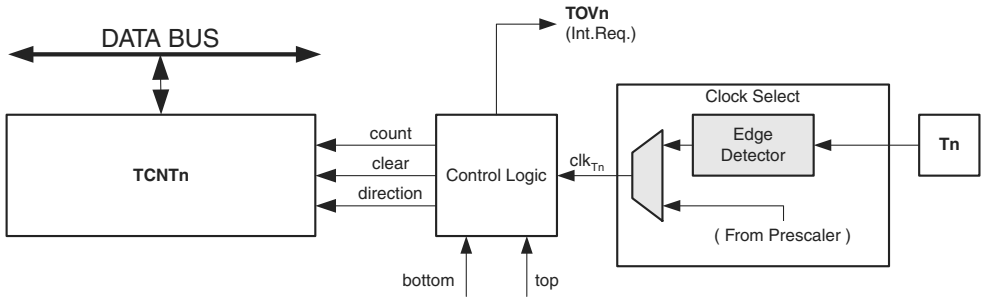
14.2 Timer/Counter Clock Sources

The Timer/Counter can be clocked by an internal or an external clock source. The clock source is selected by the Clock Select logic which is controlled by the Clock Select (CS02:0) bits located in the Timer/Counter Control Register (TCCR0B). For details on clock sources and prescaler, see “Timer/Counter0 and Timer/Counter1 Prescalers” on page 76.

14.3 Counter Unit

The main part of the 8-bit Timer/Counter is the programmable bi-directional counter unit. Figure 14-2 shows a block diagram of the counter and its surroundings.

Figure 14-2. Counter Unit Block Diagram



Signal description (internal signals):

These bits control the Output Compare pin (OC0B) behavior. If one or both of the COM0B1:0 bits are set, the OC0B output overrides the normal port functionality of the I/O pin it is connected to. However, note that the Data Direction Register (DDR) bit corresponding to the OC0B pin must be set in order to enable the output driver.

When OC0B is connected to the pin, the function of the COM0B1:0 bits depends on the WGM02:0 bit setting. Table 23 shows the COM0B1:0 bit functionality when the WGM02:0 bits are set to a normal or CTC mode (non-PWM).

Table 23. Compare Output Mode, non-PWM Mode

COM0B1	COM0B0	Description
0	0	Normal port operation, OC0B disconnected.
0	1	Toggle OC0B on Compare Match
1	0	Clear OC0B on Compare Match
1	1	Set OC0B on Compare Match

Table 24 shows the COM0B1:0 bit functionality when the WGM02:0 bits are set to fast PWM mode.

Table 24. Compare Output Mode, Fast PWM Mode(1)

COM0B1	COM0B0	Description
0	0	Normal port operation, OC0B disconnected.
0	1	Reserved
1	0	Clear OC0B on Compare Match, set OC0B at TOP
1	1	Set OC0B on Compare Match, clear OC0B at TOP

Note: 1. A special case occurs when OCR0B equals TOP and COM0B1 is set. In this case, the Compare Match is ignored, but the set or clear is done at TOP. See “Fast PWM Mode” on page 85 for more details.

Table 25 shows the COM0B1:0 bit functionality when the WGM02:0 bits are set to phase correct PWM mode.

Table 25. Compare Output Mode, Phase Correct PWM Mode(1)

COM0B1	COM0B0	Description
0	0	Normal port operation, OC0B disconnected.
0	1	Reserved
1	0	Clear OC0B on Compare Match when up-counting. Set OC0B on Compare Match when down-counting.
1	1	Set OC0B on Compare Match when up-counting. Clear OC0B on Compare Match when down-counting.

Note: 1. A special case occurs when OCR0B equals TOP and COM0B1 is set. In this case, the Compare Match is ignored, but the set or clear is done at TOP. See “Phase Correct PWM Mode” on page 87 for more details.

- **Bits 3, 2 – Res: Reserved Bits**

These bits are reserved bits in the AT90PWM1 and will always read as zero.

- **Bits 1:0 – WGM01:0: Waveform Generation Mode**

The Input Capture Register can capture the Timer/Counter value at a given external (edge triggered) event on either the Input Capture pin (ICPn). The Input Capture unit includes a digital filtering unit (Noise Canceler) for reducing the chance of capturing noise spikes.

The TOP value, or maximum Timer/Counter value, can in some modes of operation be defined by either the OCRnA Register, the ICRn Register, or by a set of fixed values. When using OCRnA as TOP value in a PWM mode, the OCRnA Register can not be used for generating a PWM output. However, the TOP value will in this case be double buffered allowing the TOP value to be changed in run time. If a fixed TOP value is required, the ICRn Register can be used as an alternative, freeing the OCRnA to be used as PWM output.

15.1.2 Definitions

The following definitions are used extensively throughout the section:

Table 28.

BOTTOM	The counter reaches the *BOTTOM* when it becomes 0x0000.
MAX	The counter reaches its *MAX*imum when it becomes 0xFFFF (decimal 65535).
TOP	The counter reaches the *TOP* when it becomes equal to the highest value in the count sequence. The TOP value can be assigned to be one of the fixed values: 0x00FF, 0x01FF, or 0x03FF, or to the value stored in the OCRnA or ICRn Register. The assignment is dependent of the mode of operation.

15.2 Accessing 16-bit Registers

The TCNTn, OCRnx, and ICRn are 16-bit registers that can be accessed by the AVR CPU via the 8-bit data bus. The 16-bit register must be byte accessed using two read or write operations. Each 16-bit timer has a single 8-bit register for temporary storing of the high byte of the 16-bit access. The same temporary register is shared between all 16-bit registers within each 16-bit timer. Accessing the low byte triggers the 16-bit read or write operation. When the low byte of a 16-bit register is written by the CPU, the high byte stored in the temporary register, and the low byte written are both copied into the 16-bit register in the same clock cycle. When the low byte of a 16-bit register is read by the CPU, the high byte of the 16-bit register is copied into the temporary register in the same clock cycle as the low byte is read.

Not all 16-bit accesses uses the temporary register for the high byte. Reading the OCRnx 16-bit registers does not involve using the temporary register.

To do a 16-bit write, the high byte must be written before the low byte. For a 16-bit read, the low byte must be read before the high byte.

The following code examples show how to access the 16-bit Timer Registers assuming that no interrupts updates the temporary register. The same principle can be used directly for accessing the OCRnx and ICRn Registers. Note that when using “C”, the compiler handles the 16-bit access.

- **Bit 3:0 – PRFMnx3:0: PSC n Fault Mode**

These four bits define the mode of operation of the Fault or Retrigger functions.

(see PSC Functional Specification for more explanations)

Table 49. Level Sensitivity and Fault Mode Operation

PRFMnx3:0	Description
0000b	No action, PSC Input is ignored
0001b	PSC Input Mode 1: Stop signal, Jump to Opposite Dead-Time and Wait
0010b	PSC Input Mode 2: Stop signal, Execute Opposite Dead-Time and Wait
0011b	PSC Input Mode 3: Stop signal, Execute Opposite while Fault active
0100b	PSC Input Mode 4: Deactivate outputs without changing timing.
0101b	PSC Input Mode 5: Stop signal and Insert Dead-Time
0110b	PSC Input Mode 6: Stop signal, Jump to Opposite Dead-Time and Wait.
0111b	PSC Input Mode 7: Halt PSC and Wait for Software Action
1000b	PSC Input Mode 8: Edge Retrigger PSC
1001b	PSC Input Mode 9: Fixed Frequency Edge Retrigger PSC
1010b	Reserved (do not use)
1011b	
1100b	
1101b	
1110b	PSC Input Mode 14: Fixed Frequency Edge Retrigger PSC and Disactivate Output
1111b	Reserved (do not use)

16.25.15 PSC 0 Input Capture Register – PICR0H and PICR0L

Bit	7	6	5	4	3	2	1	0	
	PCST0	–	–	–	PICR0[11:8]				PICR0H
	PICR0[7:0]								PICR0L
Read/Write	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	

16.25.16 PSC 2 Input Capture Register – PICR2H and PICR2L

Bit	7	6	5	4	3	2	1	0	
	PCST2	–	–	–	PICR2[11:8]				PICR2H
	PICR2[7:0]								PICR2L
Read/Write	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	

17.2.6 SPI Data Register – SPDR

Bit	7	6	5	4	3	2	1	0	
	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0	SPDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	X	X	X	X	X	X	X	X	Undefined

- **Bits 7:0 - SPD7:0: SPI Data**

The SPI Data Register is a read/write register used for data transfer between the Register File and the SPI Shift Register. Writing to the register initiates data transmission. Reading the register causes the Shift Register Receive buffer to be read.

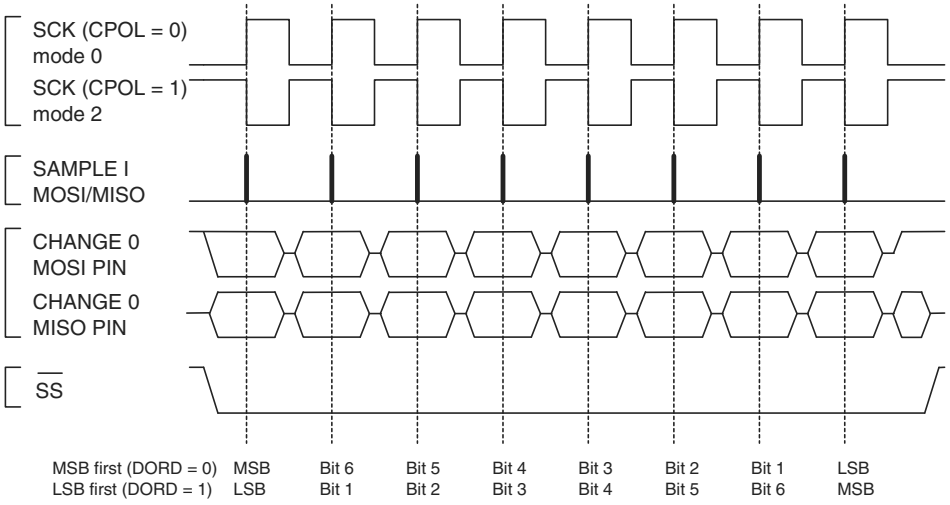
17.3 Data Modes

There are four combinations of SCK phase and polarity with respect to serial data, which are determined by control bits CPHA and CPOL. The SPI data transfer formats are shown in Figure 17-3 and Figure 17-4. Data bits are shifted out and latched in on opposite edges of the SCK signal, ensuring sufficient time for data signals to stabilize. This is clearly seen by summarizing Table 52 and Table 53, as done below:

Table 55. 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 17-3. SPI Transfer Format with CPHA = 0



This bit is cleared by hardware when the corresponding interrupt vector is executed in case the AC0IE in AC0CON register is set. Anyway, this bit is cleared by writing a logical one on it. This bit can also be used to synchronize ADC or DAC conversions.

- **Bit 2– AC2O: Analog Comparator 2 Output Bit**

AC2O bit is directly the output of the Analog comparator 2.
Set when the output of the comparator is high.
Cleared when the output comparator is low.

- **Bit 0– AC0O: Analog Comparator 0 Output Bit**

AC0O bit is directly the output of the Analog comparator 0.
Set when the output of the comparator is high.
Cleared when the output comparator is low.

18.2.4 Digital Input Disable Register 0 – DIDR0

Bit	7	6	5	4	3	2	1	0	
	ADC7D	ADC6D	ADC5D	ADC4D	ADC3D ACMPM	ADC2D ACMP2D	ADC1D	ADC0D	DIDR0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 3:2 – ACMPM and ACMP2D: ACMPM and ACMP2 Digital Input Disable**

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

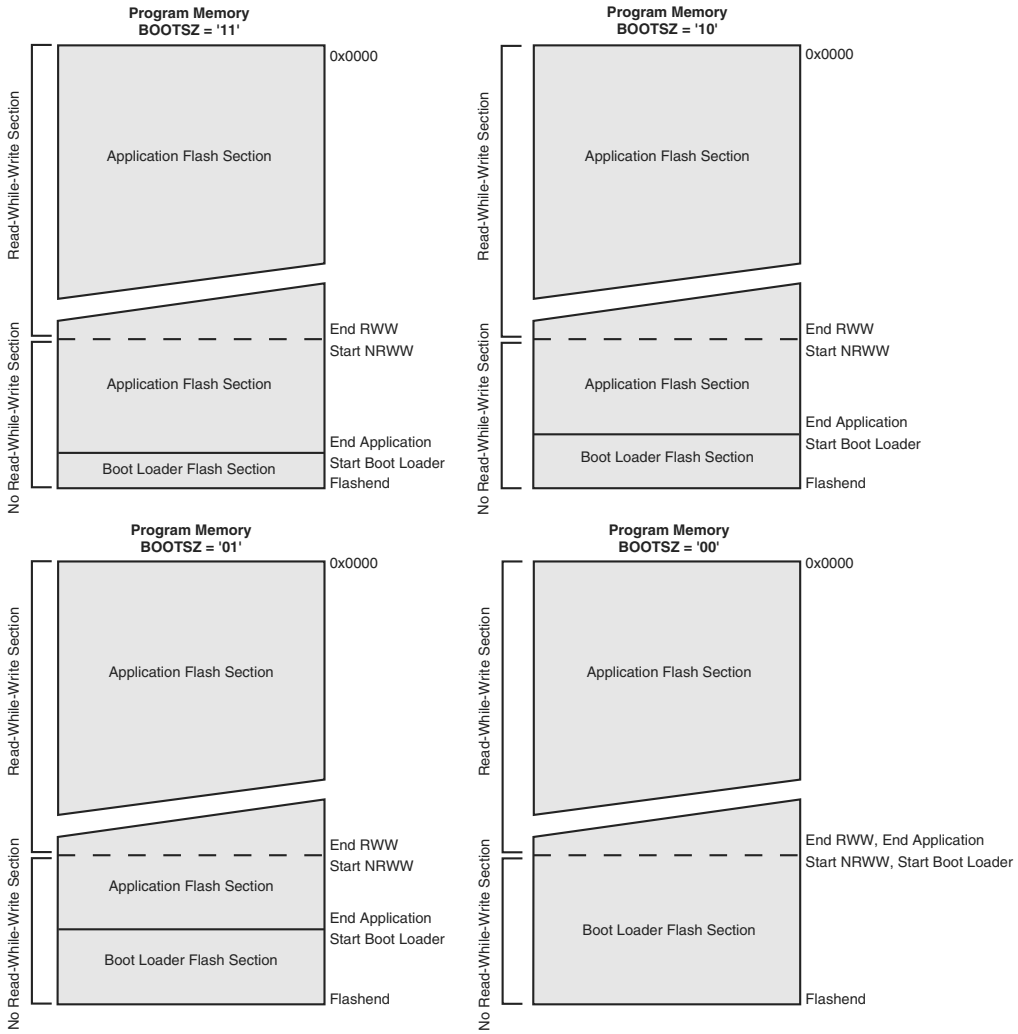
18.2.5 Digital Input Disable Register 1– DIDR1

Bit	7	6	5	4	3	2	1	0	
	-	-	ACMP0D	AMP0PD	AMP0ND				DIDR1
Read/Write	-	-	R/W	R/W	R/W				
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 5, 2: ACMP0D and ACMP1 Digital Input Disable**

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

Figure 21-2. Memory Sections



Note: 1. The parameters in the figure above are given in Table 70 on page 218.

21.4 Boot Loader Lock Bits

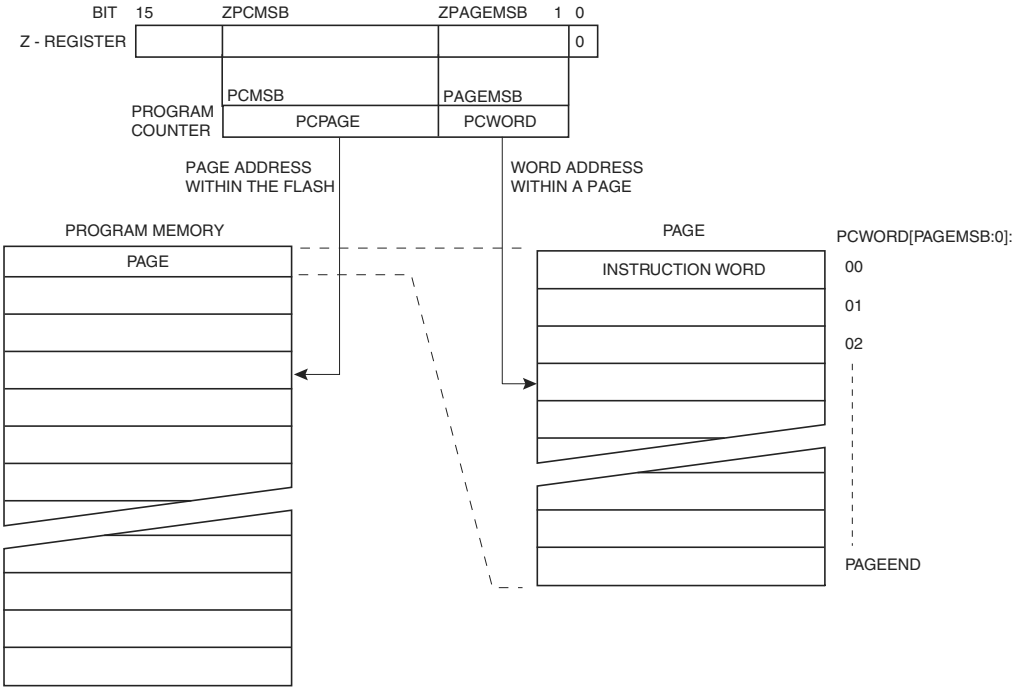
If no Boot Loader capability is needed, the entire Flash is available for application code. The Boot Loader has two separate sets of Boot Lock bits which can be set independently. This gives the user a unique flexibility to select different levels of protection.

The user can select:

- To protect the entire Flash from a software update by the MCU.
- To protect only the Boot Loader Flash section from a software update by the MCU.
- To protect only the Application Flash section from a software update by the MCU.
- Allow software update in the entire Flash.

See Table 66 and Table 67 for further details. The Boot Lock bits can be set in software and in Serial or Parallel Programming mode, but they can be cleared by a Chip Erase command only. The general Write Lock (Lock Bit mode 2) does not control the programming of the Flash memory by SPM instruction. Similarly, the general Read/Write Lock (Lock Bit mode 1) does not control reading nor writing by LPM/SPM, if it is attempted.

Figure 21-3. Addressing the Flash During SPM[1]



Note: 1. The different variables used in Figure 21-3 are listed in Table 72 on page 219.

21.7 Self-Programming the Flash

The program memory is updated in a page by page fashion. Before programming a page with the data stored in the temporary page buffer, the page must be erased. The temporary page buffer is filled one word at a time using SPM and the buffer can be filled either before the Page Erase command or between a Page Erase and a Page Write operation:

Alternative 1, fill the buffer before a Page Erase

- Fill temporary page buffer
- Perform a Page Erase
- Perform a Page Write

Alternative 2, fill the buffer after Page Erase

- Perform a Page Erase
- Fill temporary page buffer
- Perform a Page Write

If only a part of the page needs to be changed, the rest of the page must be stored (for example in the temporary page buffer) before the erase, and then be rewritten. When using alternative 1, the Boot Loader provides an effective Read-Modify-Write feature which allows the user software to first read the page, do the necessary changes, and then write back the modified data. If alternative 2 is used, it is not possible to read the old data while loading since the page is already erased. The temporary page buffer can be accessed in a random sequence. It is essential that the page address used in both the Page Erase and Page Write operation is addressing the same page. See “Simple Assembly Code Example for a Boot Loader” on page 216 for an assembly code example.

1. A: Load Command “0000 1000”.
2. B: Load Address Low Byte, 0x00.
3. Set $\overline{OE}$ to “0”, and BS1 to “1”. The Calibration byte can now be read at DATA.
4. Set $\overline{OE}$ to “1”.

22.8.15 Parallel Programming Characteristics

Figure 22-7. Parallel Programming Timing, Including some General Timing Requirements

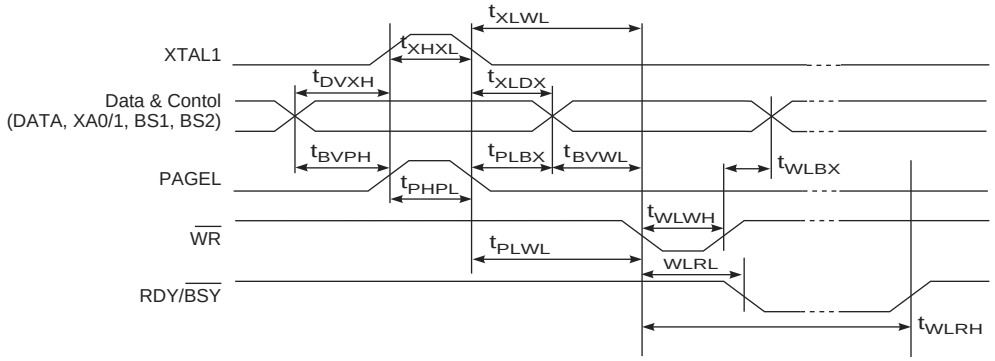
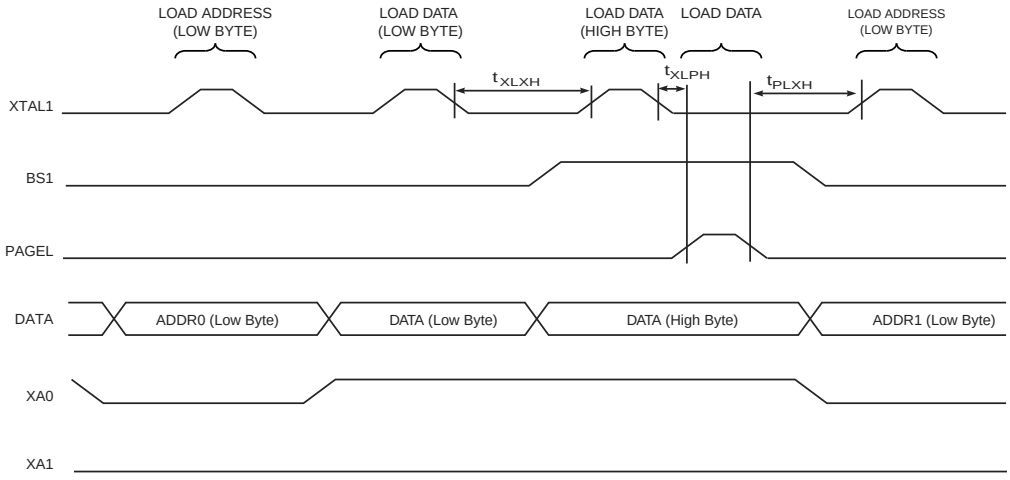


Figure 22-8. Parallel Programming Timing, Loading Sequence with Timing Requirements(1)



Note: 1. The timing requirements shown in Figure 22-7 (i.e., t_{DVXH}, t_{XHXL}, and t_{XLDX}) also apply to loading operation.

Table 86. Parallel Programming Characteristics, V_{CC} = 5V ± 10% (Continued)

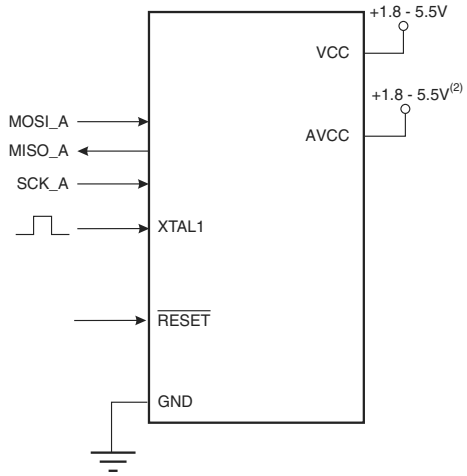
Symbol	Parameter	Min	Typ	Max	Units
t_{BVDV}	BS1 Valid to DATA valid	0		250	ns
t_{OLDV}	$\overline{OE}$ Low to DATA Valid			250	ns
t_{OHDZ}	$\overline{OE}$ High to DATA Tri-stated			250	ns

Notes: 1. t_{WLRH} is valid for the Write Flash, Write EEPROM, Write Fuse bits and Write Lock bits commands.
2. t_{WLRH_CE} is valid for the Chip Erase command.

22.9 Serial Downloading

Both the Flash and EEPROM memory arrays can be programmed using the serial SPI bus while $\overline{RESET}$ is pulled to GND. The serial interface consists of pins SCK, MOSI (input) and MISO (output). After $\overline{RESET}$ is set low, the Programming Enable instruction needs to be executed first before program/erase operations can be executed. NOTE, in Table 85 on page 225, the pin mapping for SPI programming is listed. Not all parts use the SPI pins dedicated for the internal SPI interface.

Figure 22-10. Serial Programming and Verify(1)



Notes: 1. If the device is clocked by the internal Oscillator, it is no need to connect a clock source to the XTAL1 pin.
2. V_{CC} - 0.3V < AVCC < V_{CC} + 0.3V, however, AVCC should always be within 1.8 - 5.5V

When programming the EEPROM, an auto-erase cycle is built into the self-timed programming operation (in the Serial mode ONLY) and there is no need to first execute the Chip Erase instruction. The Chip Erase operation turns the content of every memory location in both the Program and EEPROM arrays into 0xFF.

Depending on CKSEL Fuses, a valid clock must be present. The minimum low and high periods for the serial clock (SCK) input are defined as follows:

Low:> 2 CPU clock cycles for f_{ck} < 12 MHz, 3 CPU clock cycles for f_{ck} >= 12 MHz

High:> 2 CPU clock cycles for f_{ck} < 12 MHz, 3 CPU clock cycles for f_{ck} >= 12 MHz

22.9.1 Serial Programming Algorithm

When writing serial data to the AT90PWM1, data is clocked on the rising edge of SCK.

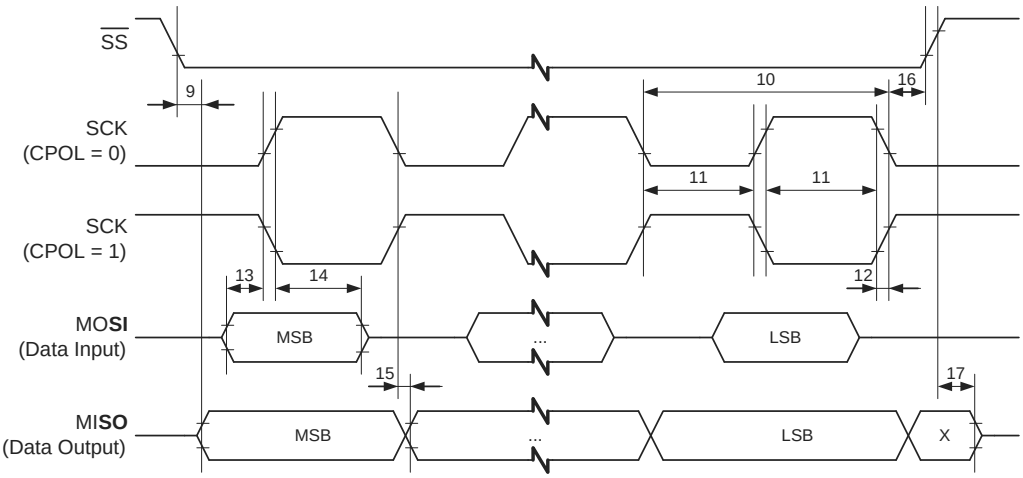
23. Electrical Characteristics(1)

23.1 Absolute Maximum Ratings*

Operating Temperature..... -40°C to +105°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 ..... 6.0V 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.

Note: 1. Electrical Characteristics for this product have not yet been finalized. Please consider all values listed herein as preliminary and non-contractual.

Figure 23-4. SPI Interface Timing Requirements (Slave Mode)



23.8 Parallel Programming Characteristics

Figure 23-5. Parallel Programming Timing, Including some General Timing Requirements

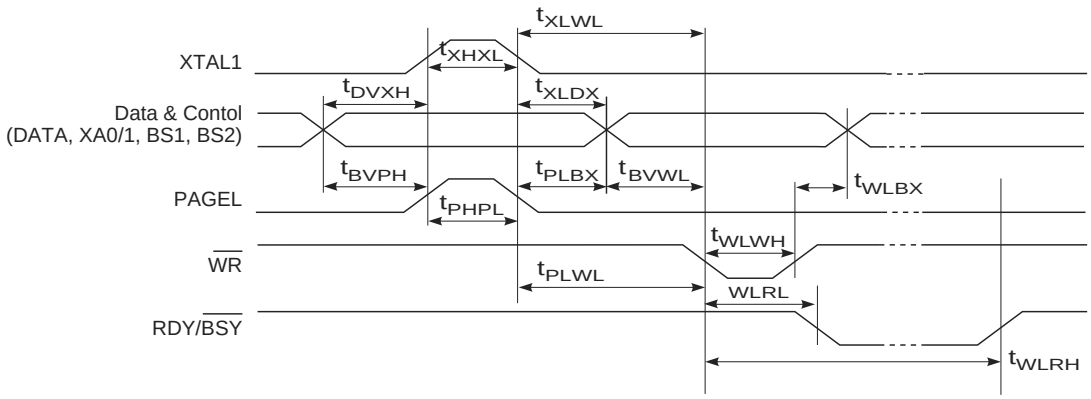
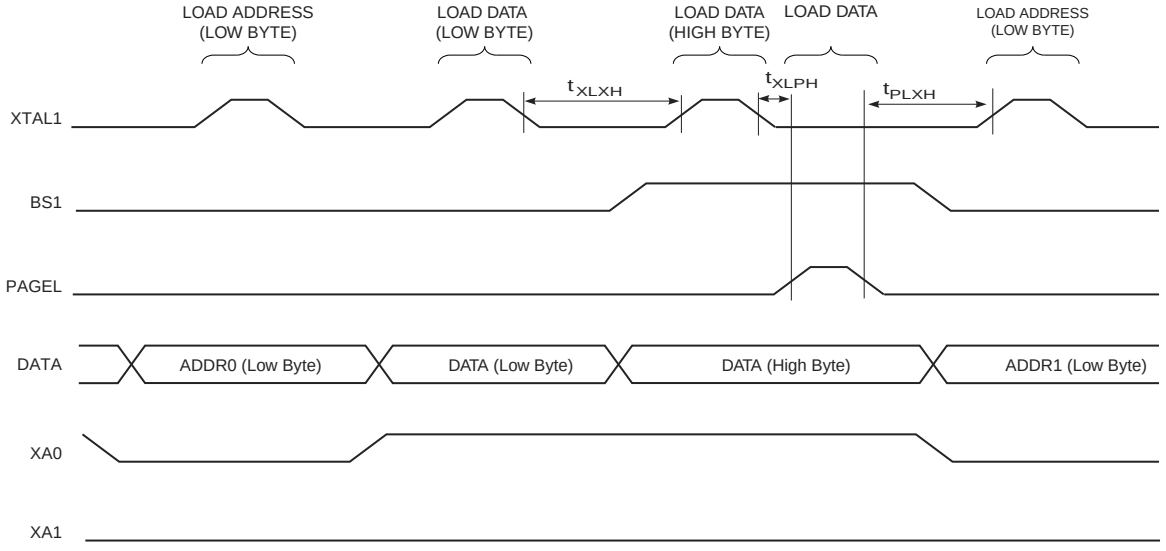


Figure 23-6. Parallel Programming Timing, Loading Sequence with Timing Requirements(1)



Note: 1. The timing requirements shown in Figure 23-5 (i.e., t_{DVXH}, t_{XHXL}, and t_{XLDX}) also apply to loading operation.

Figure 24-25. XTAL1 Input Threshold Voltage vs. V_{CC} (XTAL1 Pin Read As '1')

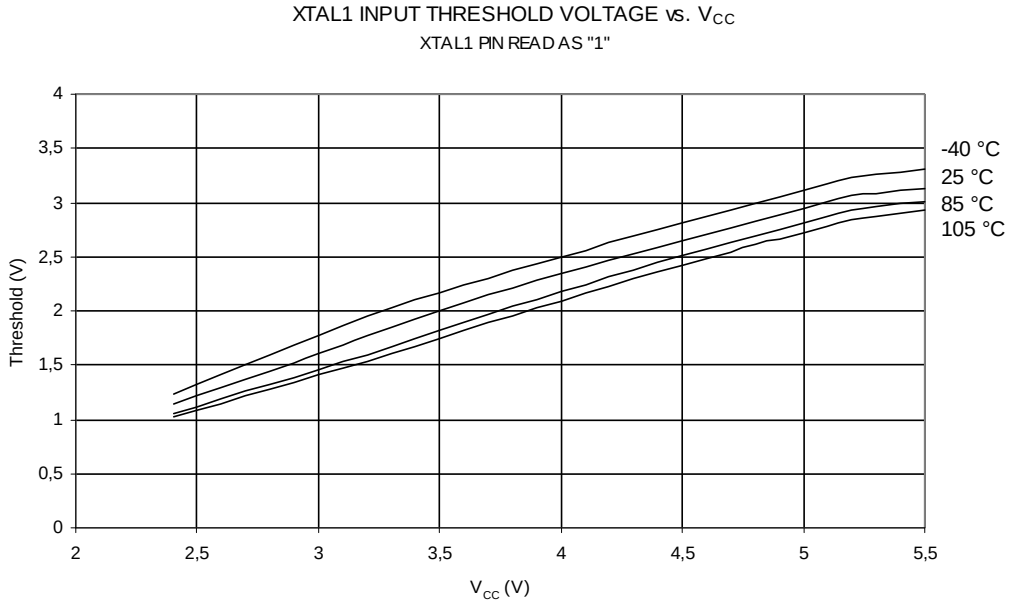
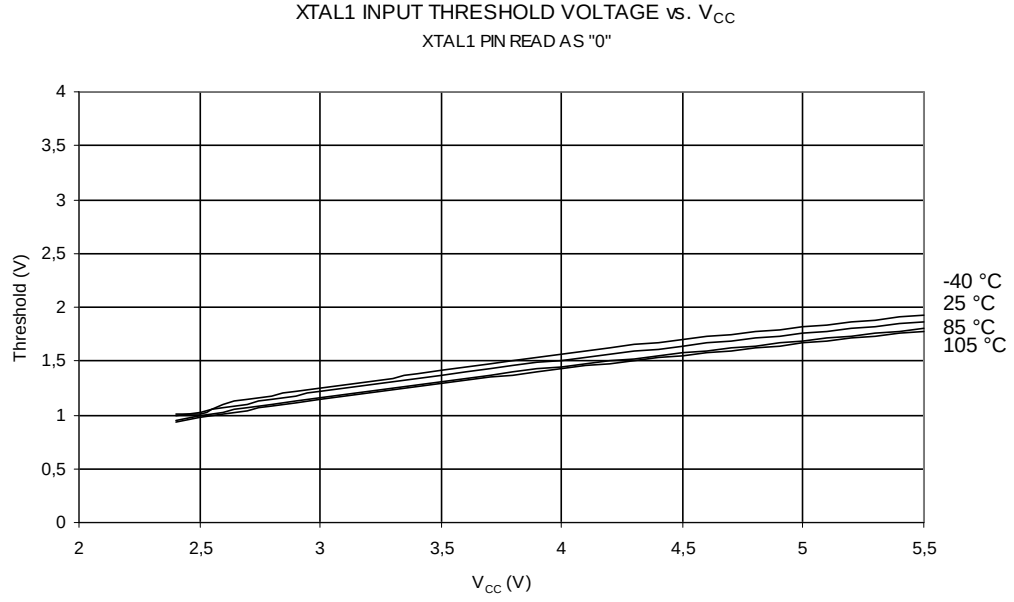


Figure 24-26. XTAL1 Input Threshold Voltage vs. V_{CC} (XTAL1 Pin Read As '0')



24.7 BOD Thresholds and Analog Comparator Offset

Figure 24-29. BOD Thresholds vs. Temperature (BODLEVEL Is 4.3V)

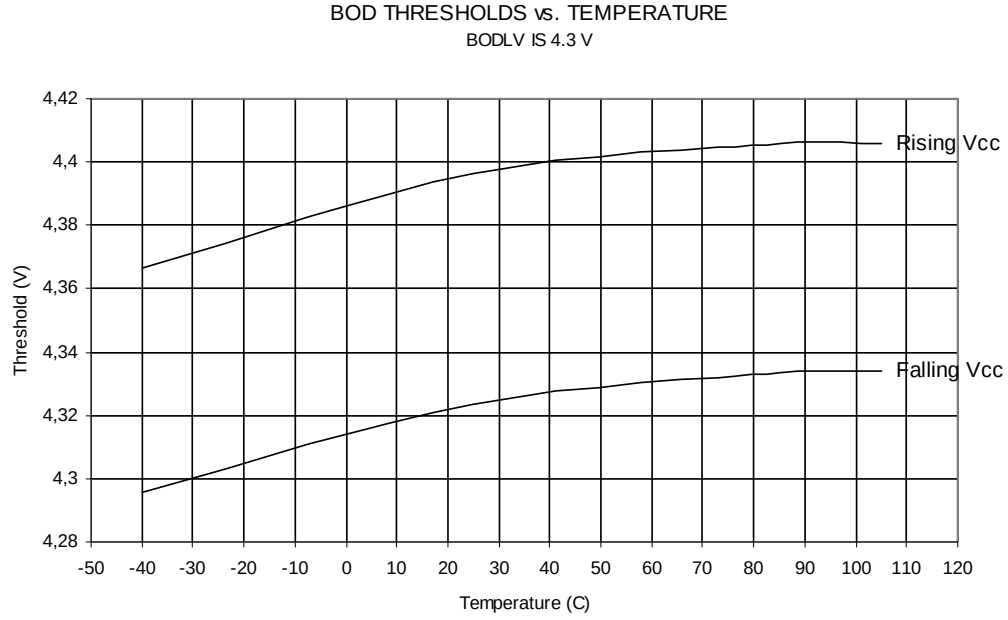


Figure 24-30. BOD Thresholds vs. Temperature (BODLEVEL Is 2.7V)

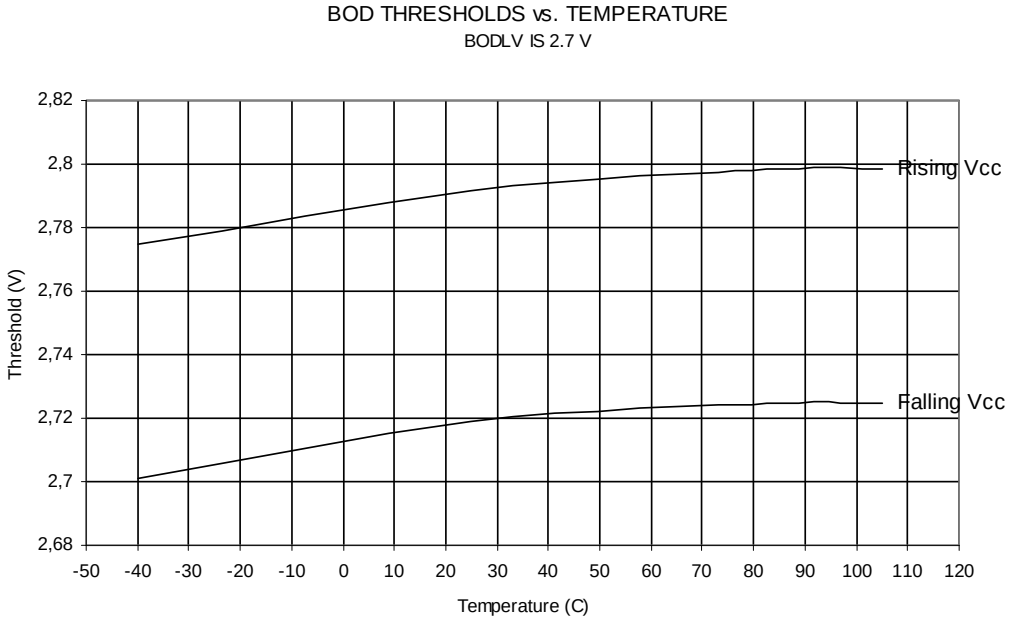


Table of Contents