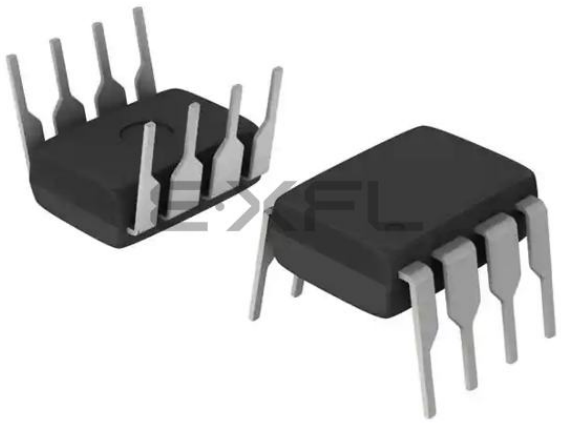


**Welcome to E-XFL.COM**

## 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"

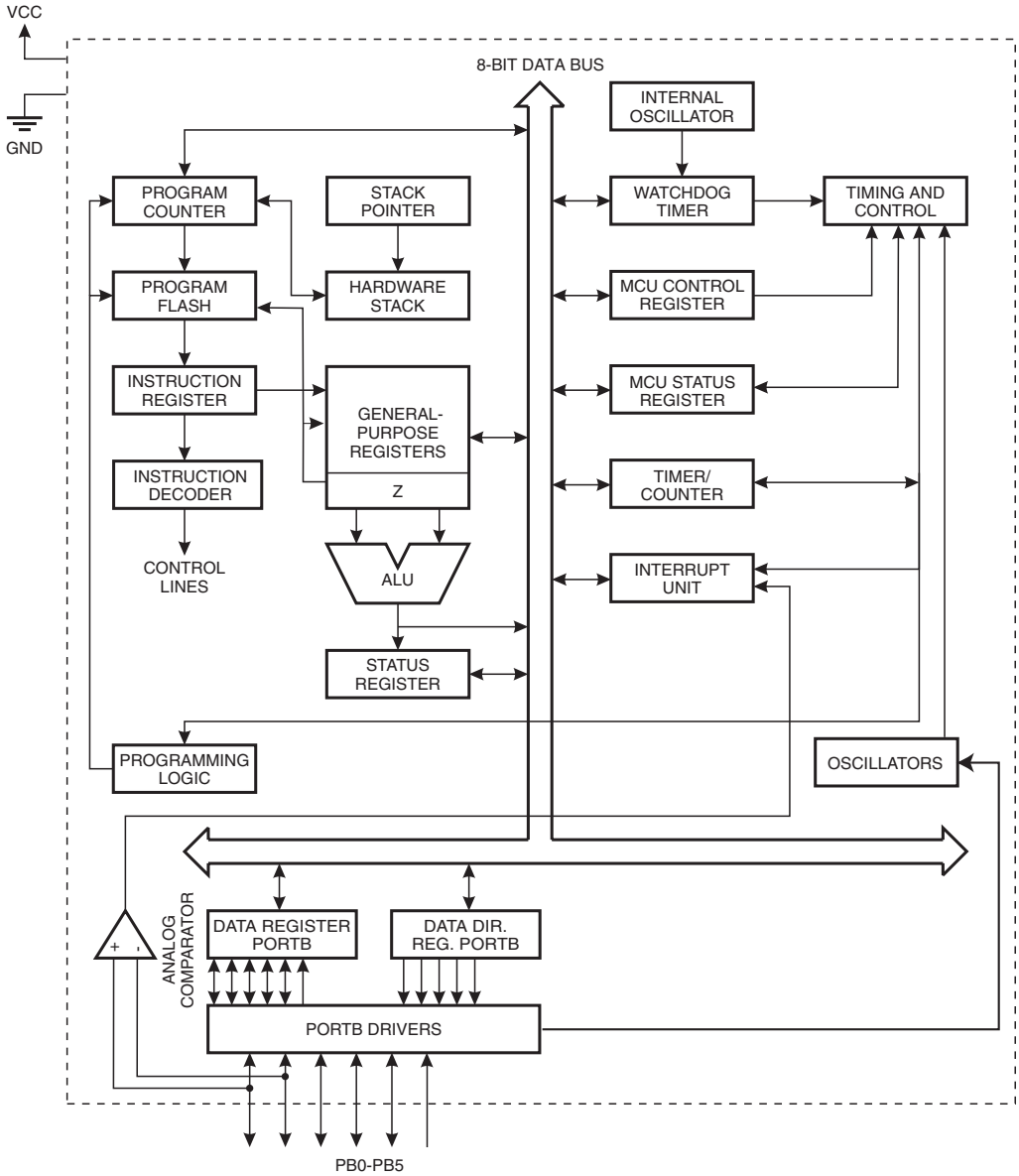
| Details | |
|---|---|
| Product Status | Obsolete |
| Core Processor | AVR |
| Core Size | 8-Bit |
| Speed | 6MHz |
| Connectivity | - |
| Peripherals | WDT |
| Number of I/O | 6 |
| Program Memory Size | 1KB (512 x 16) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | - |
| Voltage - Supply (Vcc/Vdd) | 4V ~ 5.5V |
| Data Converters | - |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| 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/attiny11-6pu |

## ATtiny11 Block Diagram

See Figure 1 on page 3. The ATtiny11 provides the following features: 1K bytes of Flash, up to five general-purpose I/O lines, one input line, 32 general-purpose working registers, an 8-bit timer/counter, internal and external interrupts, programmable Watchdog Timer with internal oscillator, and two software-selectable power-saving modes. The Idle Mode stops the CPU while allowing the timer/counters and interrupt system to continue functioning. The Power-down Mode saves the register contents but freezes the oscillator, disabling all other chip functions until the next interrupt or hardware reset. The wake-up or interrupt on pin change features enable the ATtiny11 to be highly responsive to external events, still featuring the lowest power consumption while in the power-down modes.

The device is manufactured using Atmel's high-density nonvolatile memory technology. By combining an RISC 8-bit CPU with Flash on a monolithic chip, the Atmel ATtiny11 is a powerful microcontroller that provides a highly-flexible and cost-effective solution to many embedded control applications.

**Figure 1.** The ATtiny11 Block Diagram



# Memories

## I/O Memory

The I/O space definition of the ATtiny11/12 is shown in the following table:

**Table 5.** ATtiny11/12 I/O Space

| Address Hex | Name | Device | Function |
|---|---|---|---|
| $3F | SREG | ATtiny11/12 | Status Register |
| $3B | GIMSK | ATtiny11/12 | General Interrupt Mask Register |
| $3A | GIFR | ATtiny11/12 | General Interrupt Flag Register |
| $39 | TIMSK | ATtiny11/12 | Timer/Counter Interrupt Mask Register |
| $38 | TIFR | ATtiny11/12 | Timer/Counter Interrupt Flag Register |
| $35 | MCUCR | ATtiny11/12 | MCU Control Register |
| $34 | MCUSR | ATtiny11/12 | MCU Status Register |
| $33 | TCCR0 | ATtiny11/12 | Timer/Counter0 Control Register |
| $32 | TCNT0 | ATtiny11/12 | Timer/Counter0 (8-bit) |
| $31 | OSCCAL | ATtiny12 | Oscillator Calibration Register |
| $21 | WDTCR | ATtiny11/12 | Watchdog Timer Control Register |
| $1E | EEAR | ATtiny12 | EEPROM Address Register |
| $1D | EEDR | ATtiny12 | EEPROM Data Register |
| $1C | EECR | ATtiny12 | EEPROM Control Register |
| $18 | PORTB | ATtiny11/12 | Data Register, Port B |
| $17 | DDRB | ATtiny11/12 | Data Direction Register, Port B |
| $16 | PINB | ATtiny11/12 | Input Pins, Port B |
| $08 | ACSR | ATtiny11/12 | Analog Comparator Control and Status Register |

Note: Reserved and unused locations are not shown in the table.

All the different ATtiny11/12 I/O and peripherals are placed in the I/O space. The different I/O locations are accessed by the IN and OUT instructions transferring data between the 32 general-purpose working registers and the I/O space. I/O registers within the address range $00 - $1F are directly bit-accessible using the SBI and CBI instructions. In these registers, the value of single bits can be checked by using the SBIS and SBIC instructions. Refer to the Instruction Set Summary for more details.

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

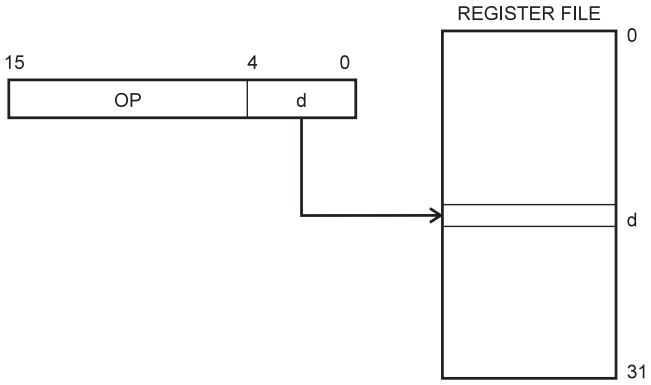
The different I/O and peripherals control registers are explained in the following sections.

## Program and Data Addressing Modes

The ATtiny11/12 AVR RISC Microcontroller supports powerful and efficient addressing modes. This section describes the different addressing modes supported in the ATtiny11/12. 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

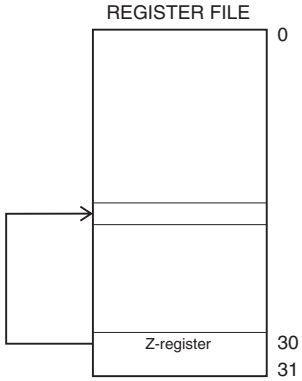
**Figure 8.** Direct Single-register Addressing




The operand is contained in register d (Rd).

### Register Indirect

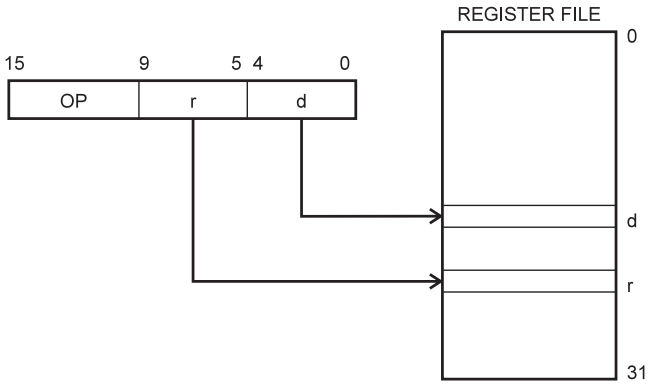
**Figure 9.** Indirect Register Addressing




The register accessed is the one pointed to by the Z-register (R31, R30).

### Register Direct, Two Registers Rd and Rr

**Figure 10.** Direct Register Addressing, Two Registers




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

## Sleep Modes

### Sleep Modes for the ATtiny11

To enter the sleep modes, the SE bit in MCUCR must be set (one) and a SLEEP instruction must be executed. The SM bit in the MCUCR register selects which sleep mode (Idle or Power-down) will be activated by the SLEEP instruction. If an enabled interrupt occurs while the MCU is in a sleep mode, the MCU awakes, executes the interrupt routine, and resumes execution from the instruction following SLEEP. On wake-up from Power Down Mode on pin change, two instruction cycles are executed before the pin change interrupt flag is updated. During these cycles, the prosessor executes intructions, but the interrupt condition is not readable, and the interrupt routine has not startet yet. The contents of the register file and I/O memory are unaltered. If a reset occurs during Sleep Mode, the MCU wakes up and executes from the Reset vector.

**Idle Mode**

When the SM bit is cleared (zero), the SLEEP instruction forces the MCU into the Idle Mode, stopping the CPU but allowing Timer/Counters, Watchdog and the interrupt system to continue operating. This enables the MCU to wake up from external triggered interrupts as well as internal ones like Timer Overflow interrupt and Watchdog Reset. If wake-up from the Analog Comparator interrupt is not required, the analog comparator can be powered down by setting the ACD-bit in the Analog Comparator Control and Status register – ACSR. This will reduce power consumption in Idle Mode. When the MCU wakes up from Idle mode, the CPU starts program execution immediately.

**Power-down Mode**

When the SM bit is set (one), the SLEEP instruction forces the MCU into the Power-down Mode. In this mode, the external oscillator is stopped, while the external interrupts and the Watchdog (if enabled) continue operating. Only an external reset, a watchdog reset (if enabled), an external level interrupt (INT0), or an pin change interrupt can wake up the MCU.

Note that if a level-triggered or pin change interrupt is used for wake-up from power-down, the changed level must be held for a time longer than the reset delay period of $t_{TOUT}$. Otherwise, the MCU will fail to wake up.

### Sleep Modes for the ATtiny12

To enter the sleep modes, the SE bit in MCUCR must be set (one) and a SLEEP instruction must be executed. The SM bit in the MCUCR register selects which sleep mode (Idle or Power-down) will be activated by the SLEEP instruction. If an enabled interrupt occurs while the MCU is in a sleep mode, the MCU awakes. The CPU is then halted for four cycles, it executes the interrupt routine, and resumes execution from the instruction following SLEEP. The contents of the register file and I/O memory are unaltered. If a reset occurs during sleep mode, the MCU wakes up and executes from the Reset vector.

**Idle Mode**

When the SM bit is cleared (zero), the SLEEP instruction forces the MCU into the Idle Mode stopping the CPU but allowing Timer/Counters, Watchdog and the interrupt system to continue operating. This enables the MCU to wake up from external triggered interrupts as well as internal ones like Timer Overflow interrupt and Watchdog Reset. If wake-up from the Analog Comparator interrupt is not required, the analog comparator can be powered down by setting the ACD-bit in the Analog Comparator Control and Status Register – ACSR. This will reduce power consumption in Idle Mode.

**Power-down Mode**

When the SM bit is set (one), the SLEEP instruction forces the MCU into the Power-down Mode. In this mode, the external oscillator is stopped, while the external interrupts and the Watchdog (if enabled) continue operating. Only an external reset, a watchdog

### Power-on Reset for the ATtiny12

A Power-on Reset (POR) pulse is generated by an on-chip detection circuit. The detection level is nominally 1.4V. The POR is activated whenever $V_{CC}$ is below the detection level. The POR circuit can be used to trigger the start-up reset, as well as detect a failure in supply voltage.

The Power-on Reset (POR) circuit ensures that the device is reset from power-on. Reaching the Power-on Reset threshold voltage invokes a delay counter, which determines the delay for which the device is kept in Reset after $V_{CC}$ rise. The time-out period of the delay counter can be defined by the user through the CKSEL fuses. The different selections for the delay period are presented in Table 10. The Reset signal is activated again, without any delay, when the $V_{CC}$ decreases below detection level.

If the built-in start-up delay is sufficient, $\overline{\text{RESET}}$ can be connected to $V_{CC}$ directly or via an external pull-up resistor. See Figure 17. By holding the $\overline{\text{RESET}}$ pin low for a period after $V_{CC}$ has been applied, the Power-on Reset period can be extended. Refer to Figure 18 for a timing example on this.

**Figure 17.** MCU Start-up, $\overline{\text{RESET}}$ Tied to $V_{CC}$.

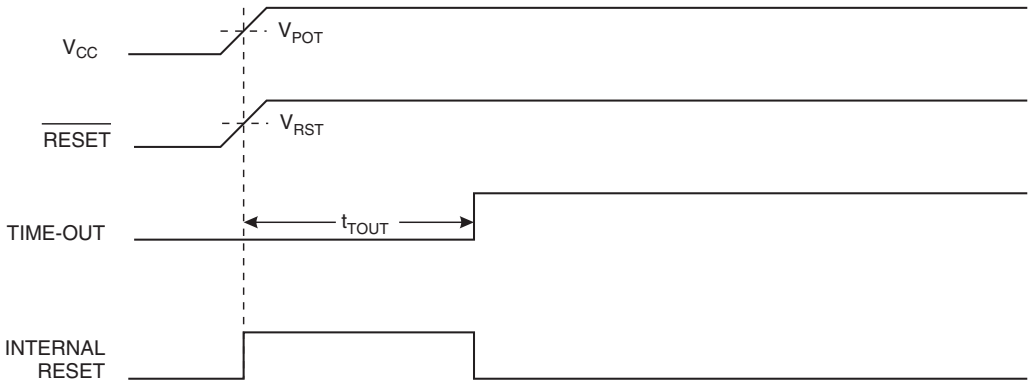
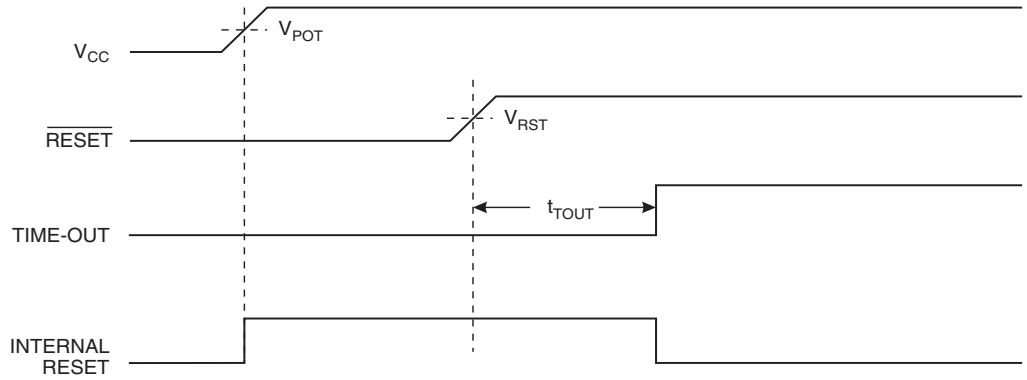



**Figure 18.** MCU Start-up, $\overline{\text{RESET}}$ Extended Externally



## General Interrupt Flag Register – GIFR

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| $3A | - | INTF0 | PCIF | - | - | - | - | - | GIFR |
| Read/Write | R | R/W | R/W | 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 ATtiny11/12 and always reads as zero.

- **Bit 6 - INTF0: External Interrupt Flag0**

When an edge on the INT0 pin triggers an interrupt request, the corresponding interrupt flag, INTF0 becomes set (one). If the I-bit in SREG and the corresponding interrupt enable bit, INT0 bit in GIMSK, are set (one), the MCU will jump to the interrupt vector. The flag is cleared when the interrupt routine is executed. Alternatively, the flag can be cleared by writing a logical one to it. The flag is always cleared when INT0 is configured as level interrupt.

- **Bit 5 - PCIF: Pin Change Interrupt Flag**

When an event on any input or I/O pin triggers an interrupt request, PCIF becomes set (one). If the I-bit in SREG and the PCIE bit in GIMSK are set (one), the MCU will jump to the interrupt vector at address $002. The flag is cleared when the interrupt routine is executed. Alternatively, the flag can be cleared by writing a logical one to it.

- **Bits 4..0 - Res: Reserved Bits**

These bits are reserved bits in the ATtiny11/12 and always read as zero.

## Timer/Counter Interrupt Mask Register – TIMSK

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| $39 | - | - | - | - | - | - | TOIE0 | - | TIMSK |
| Read/Write | R | R | R | R | R | R | R/W | R | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

- **Bit 7..2 - Res: Reserved Bits**

These bits are reserved bits in the ATtiny11/12 and always read as zero.

- **Bit 1 - TOIE0: Timer/Counter0 Overflow Interrupt Enable**

When the TOIE0 bit is set (one) and the I-bit in the Status Register is set (one), the Timer/Counter0 Overflow interrupt is enabled. The corresponding interrupt (at vector $003) is executed if an overflow in Timer/Counter0 occurs, i.e., when the Overflow Flag (Timer0) is set (one) in the Timer/Counter Interrupt Flag Register – TIFR.

- **Bit 0 - Res: Reserved Bit**

This bit is a reserved bit in the ATtiny11/12 and always reads as zero.

## I/O Port B

All AVR ports have true read-modify-write functionality when used as general digital I/O ports. This means that the direction of one port pin can be changed without unintentionally changing the direction of any other pin with the SBI and CBI instructions. The same applies for changing drive value (if configured as output) or enabling/disabling of pull-up resistors (if configured as input).

Port B is a 6-bit bi-directional I/O port.

Three I/O memory address locations are allocated for Port B, one each for the Data Register – PORTB, $18, Data Direction Register – DDRB, $17, and the Port B Input Pins – PINB, $16. The Port B Input Pins address is read only, while the Data Register and the Data Direction Register are read/write.

Ports PB5..3 have special functions as described in the section “Pin Descriptions” on page 5. If PB5 is not configured as external reset, it is input with no pull-up. On ATtiny12, it can also output a logical zero, acting as an open-drain output. Note that, since PB5 only has one possible output value, the output functionality of this pin is controlled by the DDRB register alone. If PB4 and/or PB3 are not used for clock function, they are I/O pins. All I/O pins have individually selectable pull-ups.

The Port B output buffers on PB0 to PB4 can sink 20 mA and thus drive LED displays directly. On ATtiny12, PB5 can sink 12 mA. When pins PB0 to PB4 are used as inputs and are externally pulled low, they will source current ($I_{IL}$) if the internal pull-ups are activated.

The Port B pins with alternate functions are shown in Table 16:

**Table 16.** Port B Pins Alternate Functions

| Port Pin | Alternate Functions | Device |
|---|---|---|
| PB0 | AIN0 (Analog Comparator Positive Input) | ATtiny11/12 |
| | MOSI (Data Input Line for Memory Downloading) | ATtiny12 |
| PB1 | INT0 (External Interrupt0 Input) | ATtiny11/12 |
| | AIN1 (Analog Comparator Negative Input) | ATtiny11/12 |
| | MISO (Data Output Line for Memory Downloading) | ATtiny12 |
| PB2 | T0 (Timer/Counter0 External Counter Input) | ATtiny11/12 |
| | SCK (Serial Clock Input for Serial Programming) | ATtiny12 |
| PB3 | XTAL1 (Oscillator Input) | ATtiny11/12 |
| PB4 | XTAL2 (Oscillator Output) | ATtiny11/12 |
| PB5 | $\overline{\text{RESET}}$ (External Reset Pin) | ATtiny11/12 |

When the pins PB2..0 are used for the alternate function, the DDRB and PORTB register has to be set according to the alternate function description. When PB5..3 are used for alternate functions, the values in the corresponding DDRB and PORTB bits are ignored.

**Alternate Functions of Port B**

All port B pins are connected to a pin change detector that can trigger the pin change interrupt. See “Pin Change Interrupt” on page 32 for details. In addition, Port B has the following alternate functions:

- **$\overline{\text{RESET}}$ - Port B, Bit 5**

When the RSTDISBL fuse is unprogrammed, this pin serves as external reset. When the RSTDISBL fuse is programmed, this pin is a general input pin. In ATtiny12, it is also an open-drain output pin.

- **XTAL2 - Port B, Bit 4**

XTAL2, oscillator output. When this pin is not used for clock purposes, it is a general I/O pin. Refer to section “Pin Descriptions” on page 5 for details.

- **XTAL1 - Port B, Bit 3**

XTAL1, oscillator or clock input. When this pin is not used for clock purposes, it is a general I/O pin. Refer to section “Pin Descriptions” on page 5 for details.

- **T0/SCK - Port B, Bit 2**

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. In ATtiny12 and serial programming mode, this pin serves as the serial clock input, SCK.

- **INT0/AIN1/MISO - Port B, Bit 1**

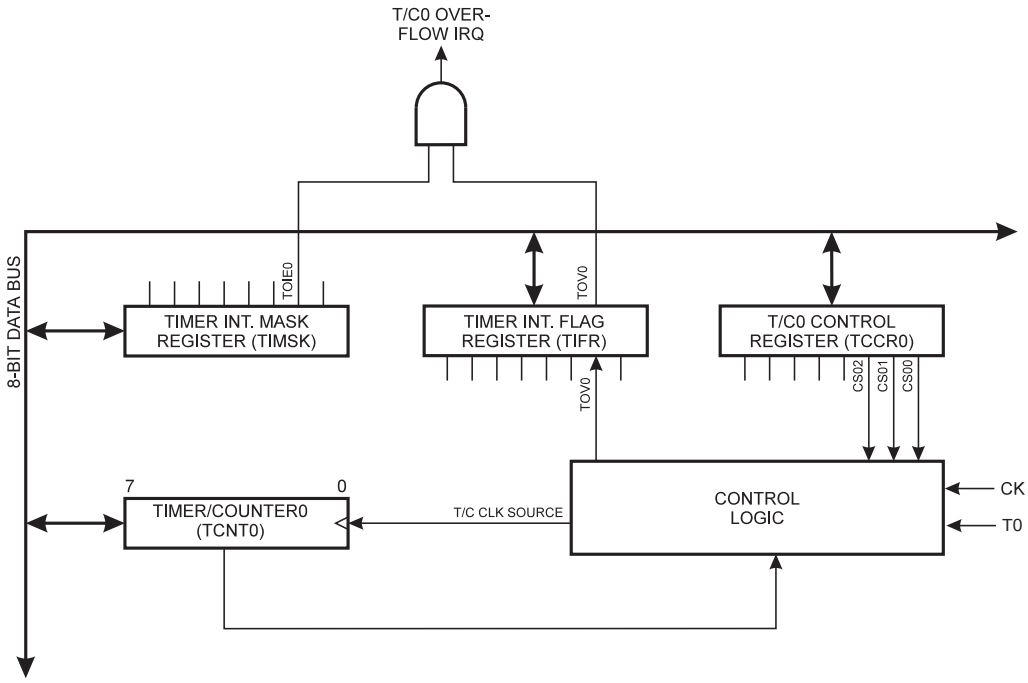
This pin can serve as the external interrupt0 input. See the interrupt description for details on how to enable this interrupt. Note that activity on this pin will trigger the interrupt even if the pin is configured as an output. This pin also serves as the negative input of the on-chip Analog Comparator. In ATtiny12 and serial programming mode, this pin serves as the serial data input, MISO.

- **AIN0/MOSI - Port B, Bit 0**

This pin also serves as the positive input of the on-chip Analog Comparator. In ATtiny12 and serial programming mode, this pin serves as the serial data output, MOSI.

During Power-down Mode, the schmitt triggers of the digital inputs are disconnected on the Analog Comparator input pins. This allows an analog voltage close to $V_{CC}/2$ to be present during power-down without causing excessive power consumption.

**Figure 23.** Timer/Counter0 Block Diagram



## Register Description

### Timer/Counter0 Control Register – TCCR0

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| $33 | - | - | - | - | - | 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 ATtiny11/12 and always read as zero.

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

The Clock Select0 bits 2,1 and 0 define the prescaling source of Timer0.

**Table 18.** 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 |

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 SW control of the counting.

### Timer Counter 0 – TCNT0

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| $32 | 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 | |

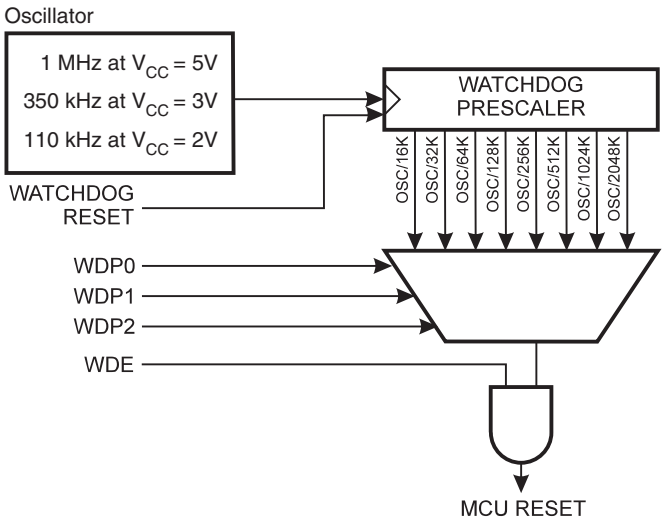
The Timer/Counter0 is implemented as an up-counter with read and write access. If the Timer/Counter0 is written and a clock source is present, the Timer/Counter0 continues counting in the timer clock cycle following the write operation.

## 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 19. 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 ATtiny11/12 resets and executes from the reset vector. For timing details on the Watchdog reset, refer to page 28.

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 24.** Watchdog Timer




## Register Description

### Watchdog Timer Control Register – WDTCR

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| $21 | - | - | - | WDTOE | WDE | WDP2 | WDP1 | WDP0 | WDTCR |
| Read/Write | R | R | R | R/W | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

- **Bits 7..5 - Res: Reserved Bits**

These bits are reserved bits in the ATtiny11/12 and will always read as zero.

- **Bit 4 - WDTOE: Watchdog Turn-off Enable**

This bit must be set (one) when the WDE bit is cleared. Otherwise, the watchdog will not be disabled. Once set, hardware will clear this bit to zero after four clock cycles. Refer to the description of the WDE bit for a watchdog disable procedure.

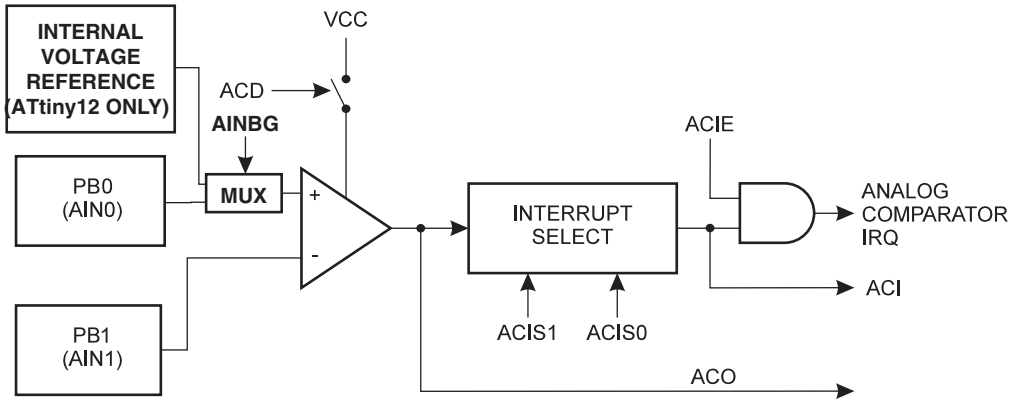
- **Bit 3 - WDE: Watchdog Enable**

When the WDE is set (one) the Watchdog Timer is enabled, and if the WDE is cleared (zero) the Watchdog Timer function is disabled. WDE can be cleared only when the WDTOE bit is set(one). To disable an enabled watchdog timer, the following procedure must be followed:

## Analog Comparator

The Analog Comparator compares the input values on the positive input PB0 (AIN0) and negative input PB1 (AIN1). When the voltage on the positive input PB0 (AIN0) is higher than the voltage on the negative input PB1 (AIN1), the Analog Comparator Output (ACO) is set (one). The comparator's output can trigger a separate interrupt, exclusive to the Analog Comparator. The user can select interrupt triggering on comparator output rise, fall or toggle. A block diagram of the comparator and its surrounding logic is shown in Figure 25.

**Figure 25.** Analog Comparator Block Diagram.




## Register Description

### Analog Comparator Control and Status Register – ACSR

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---|---|---|---|---|---|---|---|---|---|
| $08 | ACD | (AINBG) | ACO | ACI | ACIE | - | ACIS1 | ACIS0 | ACSR |
| Read/Write | R/W | R(/W) | R | R/W | R/W | R | R/W | R/W | |
| Initial Value | 0 | 0 | X | 0 | 0 | 0 | 0 | 0 | |

Note: AINBG is only available in ATtiny12.

- **Bit 7 - ACD: Analog Comparator Disable**

When this bit is set (one), the power to the Analog Comparator is switched off. This bit can be set at any time to turn off the Analog Comparator. When changing the ACD bit, the Analog Comparator Interrupt must be disabled by clearing the ACIE bit in ACSR. Otherwise an interrupt can occur when the bit is changed.

- **Bit 6 - AINBG: Analog Comparator Bandgap Select in ATtiny12**

In ATtiny12, when this bit is set, a fixed bandgap voltage of 1.22 ± 0.05V replaces the normal input to the positive input (AIN0) of the comparator. When this bit is cleared, the normal input pin PB0 is applied to the positive input of the comparator.

- **Bit 6- Res: Reserved Bit in ATtiny11**

This bit is a reserved bit in the ATtiny11 and will always read as zero.

- **Bit 5 - ACO: Analog Comparator Output**

ACO is directly connected to the comparator output.

# Memory Programming

## Program (and Data) Memory Lock Bits

The ATtiny11/12 MCU provides two lock bits which can be left unprogrammed (“1”) or can be programmed (“0”) to obtain the additional features listed in Table 21. The lock bits can only be erased with the Chip Erase command.

**Table 21.** Lock Bit Protection Modes

| Memory Lock Bits | | | |
|---|---|---|---|
| **Mode** | **LB1** | **LB2** | **Protection Type** |
| 1 | 1 | 1 | No memory lock features enabled. |
| 2 | 0 | 1 | Further programming of the Flash (and EEPROM for ATtiny12) is disabled.(1) |
| 3 | 0 | 0 | Same as mode 2, and verify is also disabled. |

Note: 1. In the High-voltage Serial Programming mode, further programming of the fuse bits are also disabled. Program the fuse bits before programming the lock bits.

## Fuse Bits in ATtiny11

The ATtiny11 has five fuse bits, FSTRT, RSTDISBL and CKSEL2..0.

- FSTRT: See Table 8, “Start-up Times for the ATtiny11 ($V_{CC}$ = 2.7V),” on page 23 for which value to use. Default value is unprogrammed (“1”).
- When RSTDISBL is programmed (“0”), the external reset function of pin PB5 is disabled.(1) Default value is unprogrammed (“1”).
- CKSEL2..0: See Table 3, “Device Clocking Options Select,” on page 10, for which combination of CKSEL2..0 to use. Default value is “100”, internal RC oscillator.

The status of the fuse bits is not affected by Chip Erase.

Note: 1. If the RSTDISBL Fuse is programmed, then the programming hardware should apply +12V to PB5 while the ATtiny11 is in Power-on Reset. If not, the part can fail to enter programming mode caused by drive contention on PB0.

## Fuse Bits in ATtiny12

The ATtiny12 has eight fuse bits, BODLEVEL, BODEN, SPIEN, RSTDISBL and CKSEL3..0. All the fuse bits are programmable in both High-voltage and Low-voltage Serial programming modes. Changing the fuses does not have any effect while in programming mode.

- The BODLEVEL Fuse selects the Brown-out Detection level and changes the start-up times. See “Brown-out Detection (ATtiny12)” on page 27. See Table 10, “ATtiny12 Clock Options and Start-up Times,” on page 25. Default value is programmed (“0”).
- When the BODEN Fuse is programmed (“0”), the Brown-out Detector is enabled. See “Brown-out Detection (ATtiny12)” on page 27. Default value is unprogrammed (“1”).
- When the SPIEN Fuse bit is programmed (“0”), Low-Voltage Serial Program and Data Downloading is enabled. Default value is programmed (“0”). Unprogramming this fuse while in the Low-Voltage Serial Programming mode will disable future in-system downloading attempts.
- When the RSTDISBL Fuse is programmed (“0”), the external reset function of pin PB5 is disabled.(1) Default value is unprogrammed (“1”). Programming this fuse while in the Low-Voltage Serial Programming mode will disable future in-system downloading attempts.

## High-voltage Serial Programming Algorithm

To program and verify the ATtiny11/12 in the High-voltage Serial Programming mode, the following sequence is recommended (See instruction formats in Table 23):

1. Power-up sequence: Apply 4.5 - 5.5V between $V_{CC}$ and GND. Set PB5 and PB0 to “0” and wait at least 100 ns. Toggle PB3 at least four times with minimum 100 ns pulse-width. Set PB3 to “0”. Wait at least 100 ns. Apply 12V to PB5 and wait at least 100 ns before changing PB0. Wait 8 µs before giving any instructions.
2. The Flash array is programmed one byte at a time by supplying first the address, then the low and high data byte. The write instruction is self-timed, wait until the PB2 (RDY/$\overline{\text{BSY}}$) pin goes high.
3. The EEPROM array (ATtiny12 only) is programmed one byte at a time by supplying first the address, then the data byte. The write instruction is self-timed, wait until the PB2 (RDY/$\overline{\text{BSY}}$) pin goes high.
4. Any memory location can be verified by using the Read instruction which returns the contents at the selected address at serial output PB2.
5. Power-off sequence:Set PB3 to “0”.
   Set PB5 to “1”.
   Turn $V_{CC}$ power off.

When writing or reading serial data to the ATtiny11/12, data is clocked on the rising edge of the serial clock, see Figure 27, Figure 28 and Table 24 for details.

**Figure 42.** Power-down Supply Current vs. $V_{CC}$

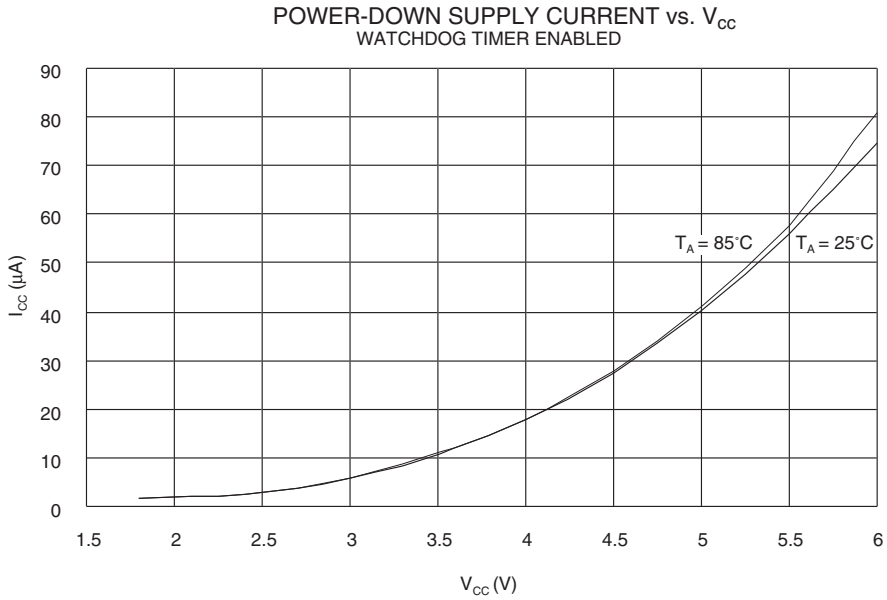
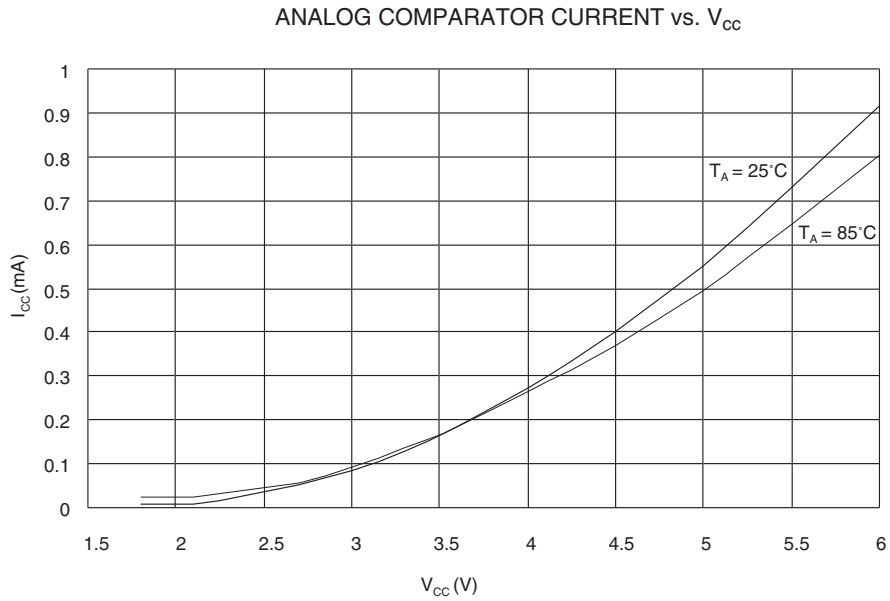



**Figure 43.** Analog Comparator Current vs. $V_{CC}$



Analog comparator offset voltage is measured as absolute offset.

**Figure 44.** Analog Comparator Offset Voltage vs. Common Mode Voltage

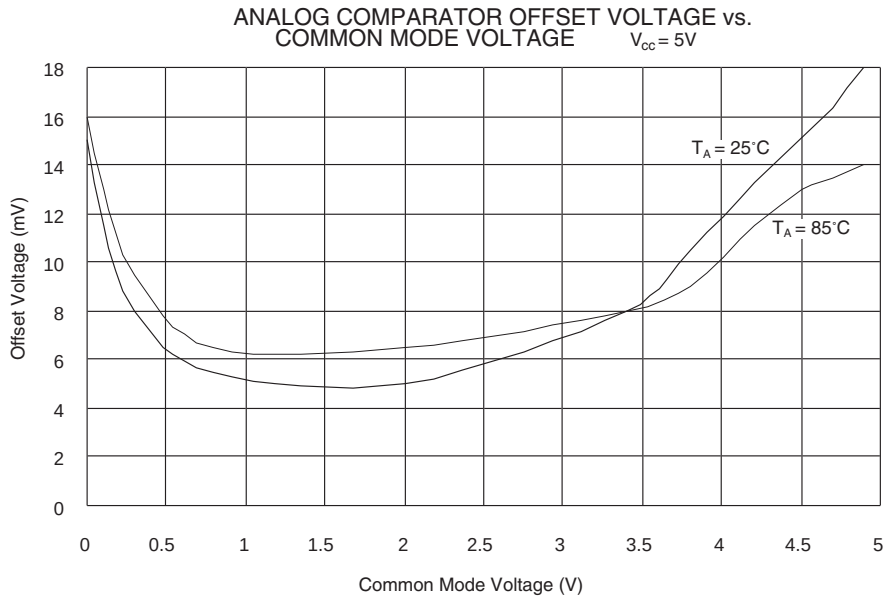



**Figure 45.** Analog Comparator Offset Voltage vs. Common Mode Voltage

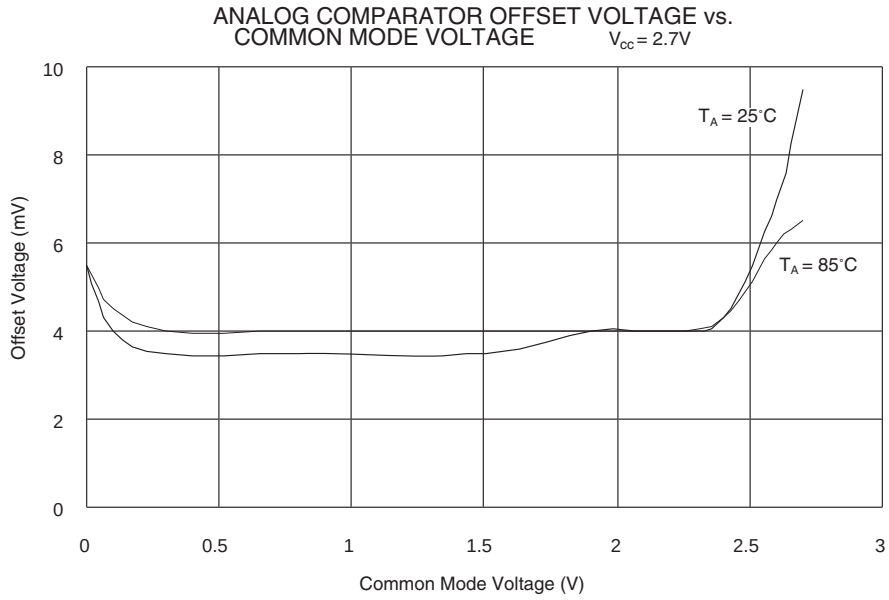


**Figure 69.** I/O Pin Sink Current vs. Output Voltage ($V_{CC}$ = 2.7V)

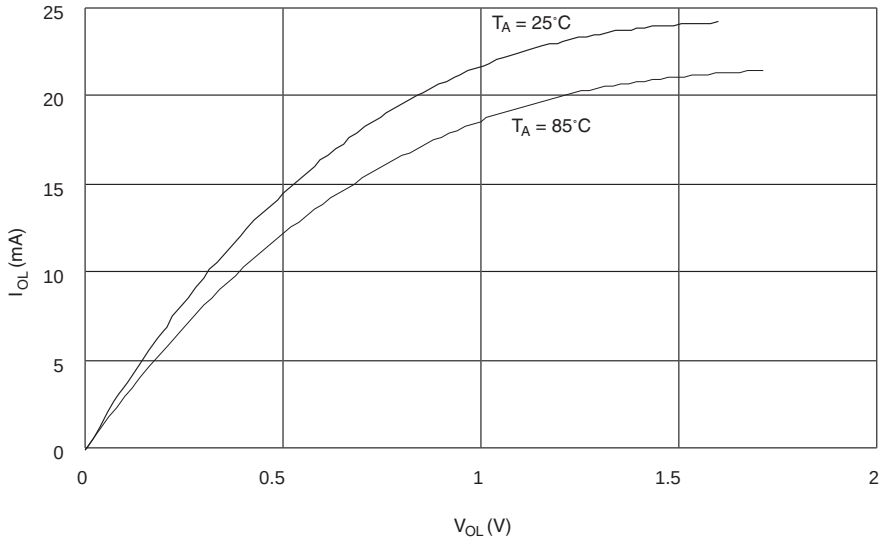



**Figure 70.** I/O Pin Source Current vs. Output Voltage ($V_{CC}$ = 2.7V)

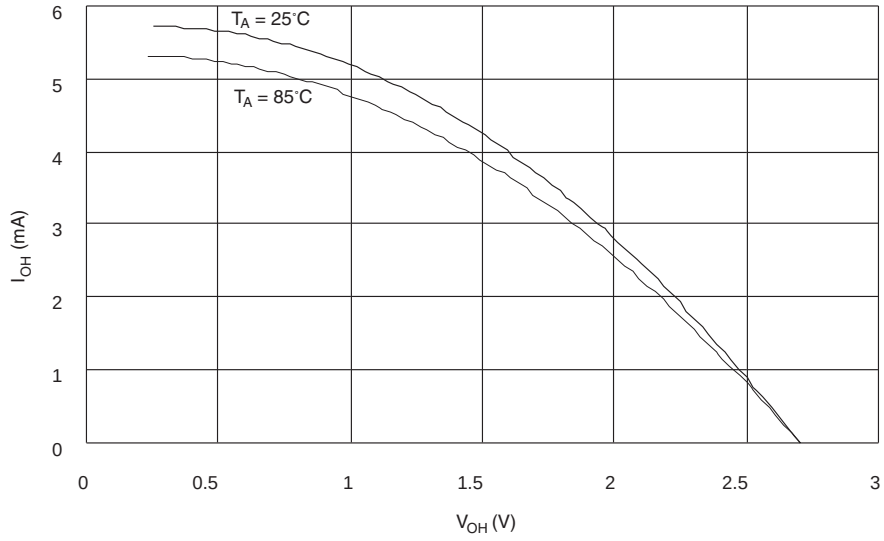


**Figure 71.** I/O Pin Input Threshold Voltage vs. $V_{CC}$ ($T_A$ = 25°C)

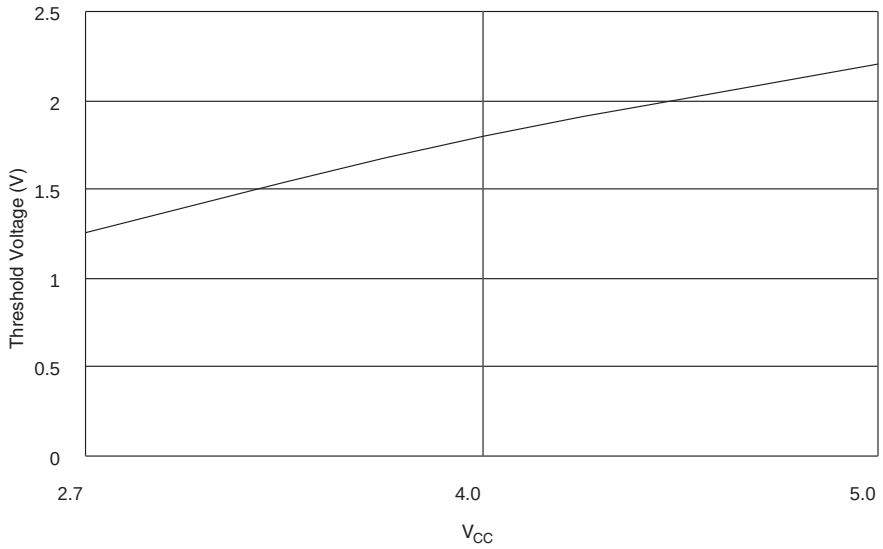
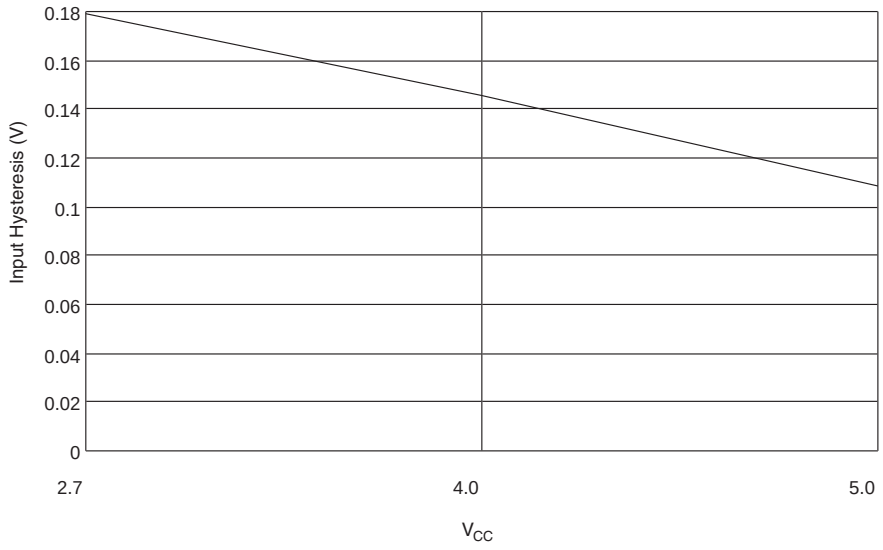



**Figure 72.** I/O Pin Input Hysteresis vs. $V_{CC}$ ($T_A$ = 25°C)



## Instruction Set Summary

| Mnemonics | 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 |
| 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 |
| 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 • (FFh - 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 |
| RCALL | k | Relative Subroutine Call | PC ← PC + k + 1 | 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 |
| 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 |
| SBRS | Rr, b | Skip if Bit in Register is Set | if (Rr(b)=1) PC ← PC + 2 or 3 | None | 1/2 |
| SBIC | P, b | Skip if Bit in I/O Register Cleared | if (P(b)=0) PC ← PC + 2 or 3 | None | 1/2 |
| SBIS | P, b | Skip if Bit in I/O Register is Set | if (P(b)=1) PC ← PC + 2 or 3 | None | 1/2 |
| 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 |

## ATtiny12

| Power Supply | Speed (MHz) | Ordering Code | Package | Operation Range |
|---|---|---|---|---|
| 1.8 - 5.5V | 1.2 | ATtiny12V-1PC<br>ATtiny12V-1SC | 8P3<br>8S2 | Commercial<br>(0°C to 70°C) |
| | | ATtiny12V-1PI<br>ATtiny12V-1PU[2]<br>ATtiny12V-1SI<br>ATtiny12V-1SU[2] | 8P3<br>8P3<br>8S2<br>8S2 | Industrial<br>(-40°C to 85°C) |
| 2.7 - 5.5V | 4 | ATtiny12L-4PC<br>ATtiny12L-4SC | 8P3<br>8S2 | Commercial<br>(0°C to 70°C) |
| | | ATtiny12L-4PI<br>ATtiny12L-4PU[2]<br>ATtiny12L-4SI<br>ATtiny12L-4SU[2] | 8P3<br>8P3<br>8S2<br>8S2 | Industrial<br>(-40°C to 85°C) |
| 4.0 - 5.5V | 8 | ATtiny12-8PC<br>ATtiny12-8SC | 8P3<br>8S2 | Commercial<br>(0°C to 70°C) |
| | | ATtiny12-8PI<br>ATtiny12-8PU[2]<br>ATtiny12-8SI<br>ATtiny12-8SU[2] | 8P3<br>8P3<br>8S2<br>8S2 | Industrial<br>(-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. Pb-free packaging alternative, complies to the European Directive for Restriction of Hazardous Substances (RoHS directive). Also Halide free and fully Green.

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