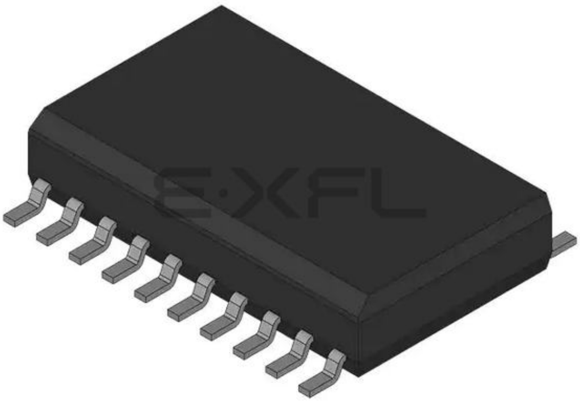


Details	
Product Status	Obsolete
Core Processor	AVR
Core Size	8-Bit
Speed	10MHz
Connectivity	USI
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	16
Program Memory Size	2KB (1K x 16)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 11x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	20-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/atmel/attiny261v-10sur

The fast-access Register File contains 32 x 8-bit general purpose working registers with a single clock cycle access time. This allows single-cycle Arithmetic Logic Unit (ALU) operation. In a typical ALU operation, two operands are output from the Register File, the operation is executed, and the result is stored back in the Register File – in one clock cycle.

Six of the 32 registers can be used as three 16-bit indirect address register pointers for Data Space addressing – enabling efficient address calculations. One of the these address pointers can also be used as an address pointer for look up tables in Flash Program memory. These added function registers are the 16-bit X-, Y-, and Z-register, described later in this section.

The ALU supports arithmetic and logic operations between registers or between a constant and a register. Single register operations can also be executed in the ALU. After an arithmetic operation, the Status Register is updated to reflect information about the result of the operation.

Program flow is provided by conditional and unconditional jump and call instructions, capable of directly addressing the whole address space. Most AVR instructions have a single 16-bit word format but 32-bit wide instructions also exist. The actual instruction set varies, as some devices only implement a part of the instruction set.

During interrupts and subroutine calls, the return address Program Counter (PC) is stored on the Stack. The Stack is effectively allocated in the general data SRAM, and consequently the Stack size is only limited by the total SRAM size and the usage of the SRAM. All user programs must initialize the SP in the Reset routine (before subroutines or interrupts are executed). The Stack Pointer (SP) is read/write accessible in the I/O space. The data SRAM can easily be accessed through the five different addressing modes supported in the AVR architecture.

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

A flexible interrupt module has its control registers in the I/O space with an additional Global Interrupt Enable bit in the Status Register. All interrupts have a separate Interrupt Vector in the Interrupt Vector table. The interrupts have priority in accordance with their Interrupt Vector position. The lower the Interrupt Vector address, the higher the priority.

The I/O memory space contains 64 addresses for CPU peripheral functions as Control Registers, SPI, and other I/O functions. The I/O memory can be accessed directly, or as the Data Space locations following those of the Register File, 0x20 - 0x5F.

4.2 ALU – Arithmetic Logic Unit

The high-performance AVR ALU operates in direct connection with all the 32 general purpose working registers. Within a single clock cycle, arithmetic operations between general purpose registers or between a register and an immediate are executed. The ALU operations are divided into three main categories – arithmetic, logical, and bit-functions. Some implementations of the architecture also provide a powerful multiplier supporting both signed/unsigned multiplication and fractional format. See the “Instruction Set” section for a detailed description.

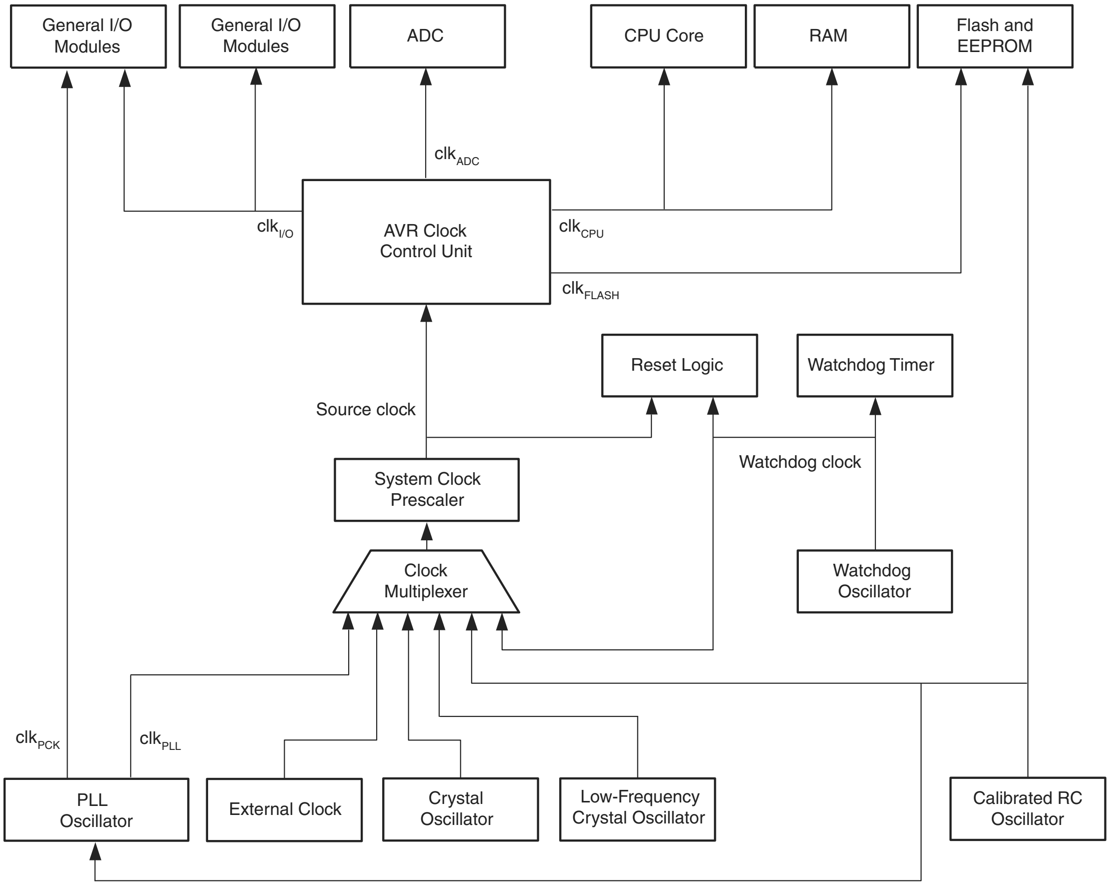
4.3 Status Register

The Status Register contains information about the result of the most recently executed arithmetic instruction. This information can be used for altering program flow in order to perform conditional operations. Note that the Status Register is updated after all ALU operations, as specified in the Instruction Set Reference. This will in many cases remove the need for using the dedicated compare instructions, resulting in faster and more compact code.

6. Clock System

Figure 6-1 presents the principal clock systems and their distribution in ATtiny261/461/861. All of the clocks need not be active at a given time. In order to reduce power consumption, the clocks to modules not being used can be halted by using different sleep modes, as described in "Power Management and Sleep Modes" on page 36.

Figure 6-1. Clock Distribution



6.1 Clock Subsystems

The clock subsystems are detailed in the sections below.

6.1.1 CPU Clock – clk_{CPU}

The CPU clock is routed to parts of the system concerned with operation of the AVR core. Examples of such modules are the General Purpose Register File, the Status Register and the Data memory holding the Stack Pointer. Halting the CPU clock inhibits the core from performing general operations and calculations.

6.1.2 I/O Clock – $clk_{I/O}$

The I/O clock is used by the majority of the I/O modules, like Timer/Counter. The I/O clock is also used by the External Interrupt module, but note that some external interrupts are detected by asynchronous logic, allowing such interrupts to be detected even if the I/O clock is halted.

Refer to “DIDR0 – Digital Input Disable Register 0” on page 162 or “DIDR1 – Digital Input Disable Register 1” on page 162 for details.

7.4 Register Description

7.4.1 MCUCR – MCU Control Register

The MCU Control Register contains control bits for power management.

Bit	7	6	5	4	3	2	1	0	
0x35 (0x55)	**–**	**PUD**	**SE**	**SM1**	**SM0**	**—**	**ISC01**	**ISC00**	**MCUCR**
Read/Write	R	R/W	R/W	R/W	R/W	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 5 – SE: Sleep Enable**

The SE bit must be written to logic one to make the MCU enter the sleep mode when the SLEEP instruction is executed. To avoid the MCU entering the sleep mode unless it is the programmer’s purpose, it is recommended to write the Sleep Enable (SE) bit to one just before the execution of the SLEEP instruction and to clear it immediately after waking up.

- **Bits 4, 3 – SM1:0: Sleep Mode Select Bits 2:0**

These bits select between the three available sleep modes as shown in Table 7-2.

Table 7-2. Sleep Mode Select

SM1	SM0	Sleep Mode
0	0	Idle
0	1	ADC Noise Reduction
1	0	Power-down
1	1	Standby

- **Bit 2 – Res: Reserved Bit**

This bit is reserved and will always read zero.

7.4.2 PRR – Power Reduction Register

The Power Reduction Register provides a method to reduce power consumption by allowing peripheral clock signals to be disabled.

Bit	7	6	5	4	3	2	1	0	
0x36 (0x56)	**–**	**-**	**-**	**-**	**PRTIM1**	**PRTIM0**	**PRUSI**	**PRADC**	**PRR**
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bits 7, 6, 5, 4 – Res: Reserved Bits**

These bits are reserved and will always read zero.

- **Bit 3 – PRTIM1: Power Reduction Timer/Counter1**

Writing a logic one to this bit shuts down the Timer/Counter1 module. When the Timer/Counter1 is enabled, operation will continue like before the shutdown.

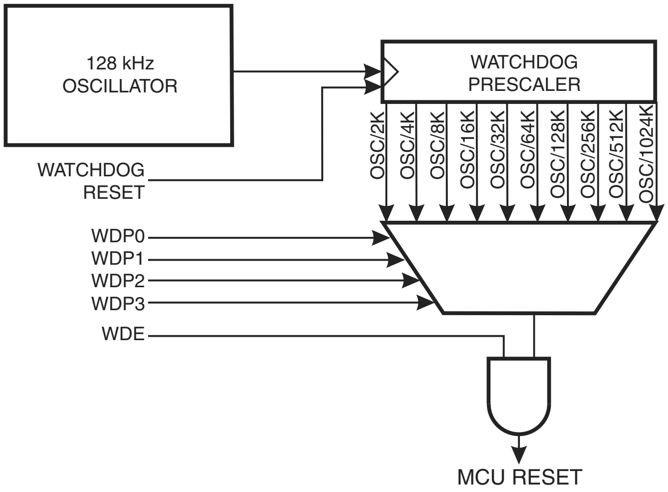
The Wathdog Timer can also be configured to generate an interrupt instead of a reset. This can be very helpful when using the Watchdog to wake-up from Power-down.

To prevent unintentional disabling of the Watchdog or unintentional change of time-out period, two different safety levels are selected by the fuse WDTON as shown in Table 8-1 Refer to "Timed Sequences for Changing the Configuration of the Watchdog Timer" on page 45 for details.

Table 8-1. WDT Configuration as a Function of the Fuse Settings of WDTON

WDTON	Safety Level	WDT Initial State	How to Disable the WDT	How to Change Time-out
Unprogrammed	1	Disabled	Timed sequence	No limitations
Programmed	2	Enabled	Always enabled	Timed sequence

Figure 8-7. Watchdog Timer



8.3.1 Timed Sequences for Changing the Configuration of the Watchdog Timer

The sequence for changing configuration differs slightly between the two safety levels. Separate procedures are described for each level.

8.3.1.1 Safety Level 1

In this mode, the Watchdog Timer is initially disabled, but can be enabled by writing the WDE bit to one without any restriction. A timed sequence is needed when disabling an enabled Watchdog Timer. To disable an enabled Watchdog Timer, the following procedure must be followed:

1. In the same operation, write a logic one to WDCE and WDE. A logic one must be written to WDE regardless of the previous value of the WDE bit.
2. Within the next four clock cycles, in the same operation, write the WDE and WDP bits as desired, but with the WDCE bit cleared.

8.3.1.2 Safety Level 2

In this mode, the Watchdog Timer is always enabled, and the WDE bit will always read as one. A timed sequence is needed when changing the Watchdog Time-out period. To change the Watchdog Time-out, the following procedure must be followed:

rupt will be generated. The start-up time is defined by the SUT and CKSEL Fuses as described in "Clock System" on page 24.

If the low level on the interrupt pin is removed before the device has woken up then program execution will not be diverted to the interrupt service routine but continue from the instruction following the SLEEP command.

9.3 Register Description

9.3.1 MCUCR – MCU Control Register

The MCU Register contains control bits for interrupt sense control.

Bit	7	6	5	4	3	2	1	0	
0x35 (0x55)	**–**	**PUD**	**SE**	**SM1**	**SM0**	**–**	**ISC01**	**ISC00**	**MCUCR**
Read/Write	R	R/W	R/W	R/W	R/W	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

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

The External Interrupt 0 is activated by the external pin INT0 or INT1 if the SREG I-flag and the corresponding interrupt mask are set. The level and edges on the external INT0 or INT1 pin that activate the interrupt are defined in Table 9-2. The value on the INT0 or INT1 pin is sampled before detecting edges. If edge or toggle interrupt is selected, pulses that last longer than one clock period will generate an interrupt. Shorter pulses are not guaranteed to generate an interrupt. If low level interrupt is selected, the low level must be held until the completion of the currently executing instruction to generate an interrupt.

Table 9-2. Interrupt 0 Sense Control

ISC01	ISC00	Description
0	0	The low level of INT0 or INT1 generates an interrupt request.
0	1	Any logical change on INT0 or INT1 generates an interrupt request.
1	0	The falling edge of INT0 or INT1 generates an interrupt request.
1	1	The rising edge of INT0 or INT1 generates an interrupt request.

9.3.2 GIMSK – General Interrupt Mask Register

Bit	7	6	5	4	3	2	1	0	
0x3B (0x5B)	**INT1**	**INT0**	**PCIE1**	**PCIE0**	**–**	**–**	**–**	**–**	**GIMSK**
Read/Write	R/W	R/W	R/W	R/w	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – INT1: External Interrupt Request 1 Enable**

When the INT1 bit is set (one) and the I-bit in the Status Register (SREG) is set (one), the external pin interrupt is enabled. The Interrupt Sense Control0 bits 1/0 (ISC01 and ISC00) in the MCU Control Register (MCUCR) define whether the external interrupt is activated on rising and/or falling edge of the INT1 pin or level sensed. Activity on the pin will cause an interrupt request even if INT1 is configured as an output. The corresponding interrupt of External Interrupt Request 1 is executed from the INT1 Interrupt Vector.

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

When the INT0 bit is set (one) and the I-bit in the Status Register (SREG) is set (one), the external pin interrupt is enabled. The Interrupt Sense Control0 bits 1/0 (ISC01 and ISC00) in the MCU

- **Port A, Bit 5 – ADC4/AIN2/PCINT5**
  - ADC4: Analog to Digital Converter, Channel 4.
  - AIN2: Analog Comparator 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.
  - PCINT5: Pin Change Interrupt source 5.

- **Port A, Bit 4 – ADC3/ICP0/PCINT4**
  - ADC3: Analog to Digital Converter, Channel 3.
  - ICP0: Timer/Counter0 Input Capture Pin.
  - PCINT4: Pin Change Interrupt source 4.

- **Port A, Bit 3 – AREF/PCINT3**
  - AREF: External analog reference for ADC. Pullup and output driver are disabled on PA3 when the pin is used as an external reference or internal voltage reference with external capacitor at the AREF pin.
  - PCINT3: Pin Change Interrupt source 3.

- **Port A, Bit 2 – ADC2/INT1/USCK/SCL/PCINT2**
  - ADC2: Analog to Digital Converter, Channel 2.
  - INT1: The PA2 pin can serve as an External Interrupt source 1.
  - USCK: Three-wire mode Universal Serial Interface Clock.
  - SCL: Two-wire mode Serial Clock for USI Two-wire mode.
  - PCINT2: Pin Change Interrupt source 2.

- **Port A, Bit 1 – ADC1/DO/PCINT1**
  - ADC1: Analog to Digital Converter, Channel 1.
  - DO: Three-wire mode Universal Serial Interface Data output. Three-wire mode Data output overrides PORTA1 value and it is driven to the port when data direction bit DDA1 is set. PORTA1 still enables the pull-up, if the direction is input and PORTA1 is set.
  - PCINT1: Pin Change Interrupt source 1.

- **Port A, Bit 0 – ADC0/DI/SDA/PCINT0**
  - ADC0: Analog to Digital Converter, Channel 0.
  - DI: Data Input in USI Three-wire mode. USI Three-wire mode does not override normal port functions, so pin must be configure as an input for DI function.
  - SDA: Two-wire mode Serial Interface Data.
  - PCINT0: Pin Change Interrupt source 0.

11.10.4 TCNT0H – Timer/Counter0 Register High Byte

Bit	7	6	5	4	3	2	1	0	
0x14 (0x34)	**TCNT0H[7:0]**								**TCNT0H**
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	

When 16-bit mode is selected (the TCW0 bit is set to one) the Timer/Counter Register TCNT0H combined to the Timer/Counter Register TCNT0L gives direct access, both for read and write operations, to the Timer/Counter unit 16-bit counter. To ensure that both the high and low bytes are read and written simultaneously when the CPU accesses these registers, the access is performed using an 8-bit temporary high byte register (TEMP). This temporary register is shared by all the other 16-bit registers. See “Accessing Registers in 16-bit Mode” on page 80

11.10.5 OCR0A – Timer/Counter0 Output Compare Register A

Bit	7	6	5	4	3	2	1	0	
0x13 (0x33)	**OCR0A[7:0]**								**OCR0A**
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 Output Compare Register A contains an 8-bit value that is continuously compared with the counter value (TCNT0L). A match can be used to generate an Output Compare interrupt.

In 16-bit mode the OCR0A register contains the low byte of the 16-bit Output Compare Register. To ensure that both the high and the low bytes are written simultaneously when the CPU writes to these registers, the access is performed using an 8-bit temporary high byte register (TEMP). This temporary register is shared by all the other 16-bit registers. See “Accessing Registers in 16-bit Mode” on page 80.

11.10.6 OCR0B – Timer/Counter0 Output Compare Register B

Bit	7	6	5	4	3	2	1	0	
0x12 (0x32)	**OCR0B[7:0]**								**OCR0B**
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 Output Compare Register B contains an 8-bit value that is continuously compared with the counter value (TCNT0L in 8-bit mode and TCNTH in 16-bit mode). A match can be used to generate an Output Compare interrupt.

In 16-bit mode the OCR0B register contains the high byte of the 16-bit Output Compare Register. To ensure that both the high and the low bytes are written simultaneously when the CPU writes to these registers, the access is performed using an 8-bit temporary high byte register (TEMP). This temporary register is shared by all the other 16-bit registers. See “Accessing Registers in 16-bit Mode” on page 80.

11.10.7 TIMSK – Timer/Counter0 Interrupt Mask Register

Bit	7	6	5	4	3	2	1	0	
0x39 (0x59)	**OCIE1D**	**OCIE1A**	**OCIE1B**	**OCIE0A**	**OCIE0B**	**TOIE1**	**TOIE0**	**TICIE0**	**TIMSK**
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 4 – OCIE0A: Timer/Counter0 Output Compare Match A Interrupt Enable**

When the OCIE0A bit is written to one, and the I-bit in the Status Register is set, the Timer/Counter0 Compare Match A interrupt is enabled. The corresponding interrupt is executed

- **Bits 5,4 – COM1B1, COM1B0: Comparator B Output Mode, Bits 1 and 0**

These bits control the behaviour of the Waveform Output (OCW1B) and the connection of the Output Compare pin (OC1B). If one or both of the COM1B1:0 bits are set, the OC1B output overrides the normal port functionality of the I/O pin it is connected to. The complementary $\overline{\text{OC1B}}$ output is connected only in PWM modes when the COM1B1:0 bits are set to “01”. Note that the Data Direction Register (DDR) bit corresponding to the OC1B pin must be set in order to enable the output driver.

The function of the COM1B1:0 bits depends on the PWM1B and WGM11:10 bit settings. Table 12-13 shows the COM1B1:0 bit functionality when the PWM1B bit is set to Normal Mode (non-PWM).

Table 12-13. Compare Output Mode, Normal Mode (non-PWM)

COM1B1:0	OCW1B Behaviour	OC1B Pin	$\overline{\text{OC1B}}$ Pin
00	Normal port operation.	Disconnected	Disconnected
01	Toggle on Compare Match.	Connected	Disconnected
10	Clear on Compare Match.	Connected	Disconnected
11	Set on Compare Match.	Connected	Disconnected

Table 12-14 shows the COM1B1:0 bit functionality when the PWM1B and WGM11:10 bits are set to Fast PWM Mode.

Table 12-14. Compare Output Mode, Fast PWM Mode

COM1B1:0	OCW1B Behaviour	OC1B Pin	$\overline{\text{OC1B}}$ Pin
00	Normal port operation.	Disconnected	Disconnected
01	Cleared on Compare Match. Set when TCNT1 = 0x000.	Connected	Connected
10	Cleared on Compare Match. Set when TCNT1 = 0x000.	Connected	Disconnected
11	Set on Compare Match. Cleared when TCNT1 = 0x000.	Connected	Disconnected

Table 12-15 shows the COM1B1:0 bit functionality when the PWM1B and WGM11:10 bits are set to Phase and Frequency Correct PWM Mode.

Table 12-15. Compare Output Mode, Phase and Frequency Correct PWM Mode

COM1B1:0	OCW1B Behaviour	OC1B Pin	$\overline{\text{OC1B}}$ Pin
00	Normal port operation.	Disconnected	Disconnected
01	Cleared on Compare Match when up-counting. Set on Compare Match when down-counting.	Connected	Connected
10	Cleared on Compare Match when up-counting. Set on Compare Match when down-counting.	Connected	Disconnected
11	Set on Compare Match when up-counting. Cleared on Compare Match when down-counting.	Connected	Disconnected

Bits COM1B1 and COM1B0 are shadowed in TCCR1C. Writing to bits COM1B1 and COM1B0 will also change bits COM1B1S and COM1B0S in TCCR1C. Similary, changes written to bits

13. USI – Universal Serial Interface

13.1 Features

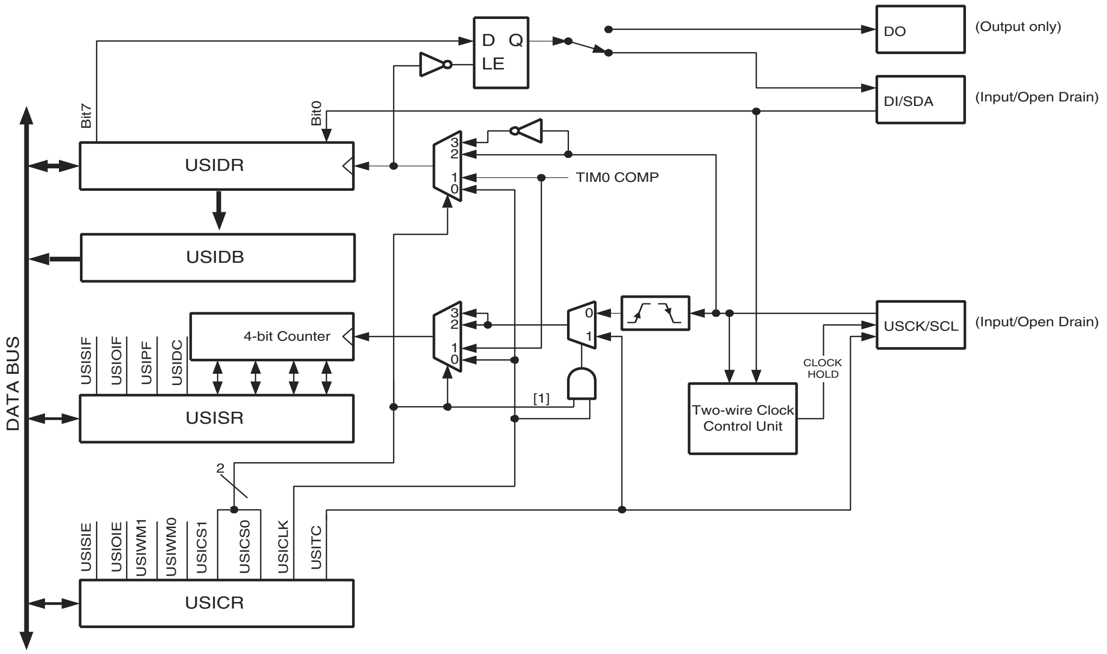
- **Two-wire Synchronous Data Transfer (Master or Slave)**
- **Three-wire Synchronous Data Transfer (Master or Slave)**
- **Data Received Interrupt**
- **Wakeup from Idle Mode**
- **In Two-wire Mode: Wake-up from All Sleep Modes, Including Power-down Mode**
- **Two-wire Start Condition Detector with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

13.2 Overview

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

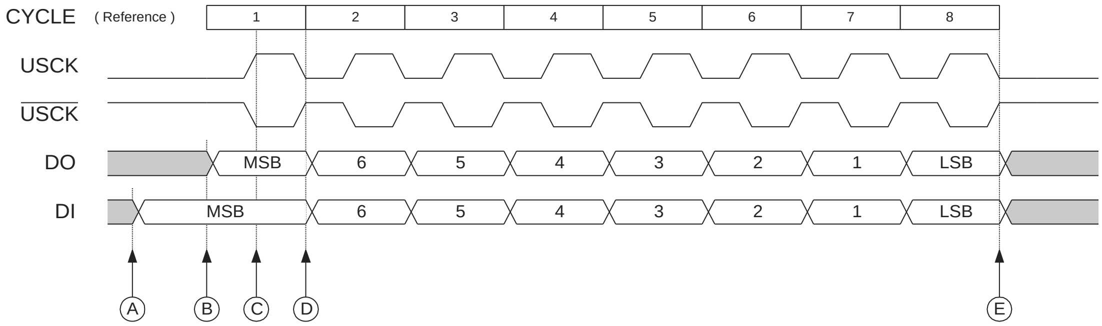
A simplified block diagram of the USI is shown in Figure 13-1 For actual placement of I/O pins refer to “Pinout ATtiny261/461/861 and ATtiny261V/461V/861V” on page 2. Device-specific I/O Register and bit locations are listed in the “Register Descriptions” on page 132.

Figure 13-1. Universal Serial Interface, Block Diagram



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

Figure 13-3. Three-wire Mode, Timing Diagram



The Three-wire mode timing is shown in Figure 13-3. At the top of the figure is a USCK cycle reference. One bit is shifted into the USI Data Register (USIDR) for each of these cycles. The USCK timing is shown for both external clock modes. In External Clock mode 0 (USICS0 = 0), DI is sampled at positive edges, and DO is changed (Data Register is shifted by one) at negative edges. In external clock mode 1 (USICS0 = 1) the opposite edges with respect to mode 0 are used. In other words, data is sampled at negative and output is changed at positive edges. The USI clock modes corresponds to the SPI data mode 0 and 1.

Referring to the timing diagram (Figure 13-3), a bus transfer involves the following steps:

1. The slave and master devices set up their data outputs and, depending on the protocol used, enable their output drivers (mark A and B). The output is set up by writing the data to be transmitted to the USI Data Register. The output is enabled by setting the corresponding bit in the Data Direction Register of Port A. Note that there is not a preferred order of points A and B in the figure, but both must be at least one half USCK cycle before point C, where the data is sampled. This is in order to ensure that the data setup requirement is satisfied. The 4-bit counter is reset to zero.
2. The master software generates a clock pulse by toggling the USCK line twice (C and D). The bit values on the data input (DI) pins are sampled by the USI on the first edge (C), and the data output is changed on the opposite edge (D). The 4-bit counter will count both edges.
3. Step 2. is repeated eight times for a complete register (byte) transfer.
4. After eight clock pulses (i.e., 16 clock edges) the counter will overflow and indicate that the transfer has been completed. The data bytes transferred must now be processed before a new transfer can be initiated. The overflow interrupt will wake up the processor if it is set to Idle mode. Depending on the protocol used the slave device can now set its output to high impedance.

13.3.2 SPI Master Operation Example

The following code demonstrates how to use the USI module as a SPI Master:

```
SPITransfer:
    sts    USIDR,r16
    ldi    r16,(1<<USIOIF)
    sts    USISR,r16
    ldi    r16,(1<<USIWM0)|(1<<USICS1)|(1<<USICLK)|(1<<USITC)
```

13.5.2 USIBR – USI Buffer Register

Bit	7	6	5	4	3	2	1	0	
0x10 (0x30)	**MSB**							**LSB**	**USIBR**
Read/Write	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	

The content of the Serial Register is loaded to the USI Buffer Register when the trasfer is completed, and instead of accessing the USI Data Register (the Serial Register) the USI Data Buffer can be accessed when the CPU reads the received data. This gives the CPU time to handle other program tasks too as the controlling of the USI is not so timing critical. The USI flags as set same as when reading the USIDR register.

13.5.3 USISR – USI Status Register

Bit	7	6	5	4	3	2	1	0	
0x0E (0x2E)	**USISIF**	**USIOIF**	**USIPF**	**USIDC**	**USICNT3**	**USICNT2**	**USICNT1**	**USICNT0**	**USISR**
Read/Write	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

The Status Register contains Interrupt Flags, line Status Flags and the counter value.

- **Bit 7 – USISIF: Start Condition Interrupt Flag**

When Two-wire mode is selected, the USISIF Flag is set (to one) when a start condition is detected. When output disable mode or Three-wire mode is selected and (USICSx = 0b11 & USICLK = 0) or (USICS = 0b10 & USICLK = 0), any edge on the SCK pin sets the flag.

An interrupt will be generated when the flag is set while the USISIE bit in USICR and the Global Interrupt Enable Flag are set. The flag will only be cleared by writing a logical one to the USISIF bit. Clearing this bit will release the start detection hold of USCL in Two-wire mode.

A start condition interrupt will wakeup the processor from all sleep modes.

- **Bit 6 – USIOIF: Counter Overflow Interrupt Flag**

This flag is set (one) when the 4-bit counter overflows (i.e., at the transition from 15 to 0). An interrupt will be generated when the flag is set while the USIOIE bit in USICR and the Global Interrupt Enable Flag are set. The flag will only be cleared if a one is written to the USIOIF bit. Clearing this bit will release the counter overflow hold of SCL in Two-wire mode.

A counter overflow interrupt will wakeup the processor from Idle sleep mode.

- **Bit 5 – USIPF: Stop Condition Flag**

When Two-wire mode is selected, the USIPF Flag is set (one) when a stop condition is detected. The flag is cleared by writing a one to this bit. Note that this is not an Interrupt Flag. This signal is useful when implementing Two-wire bus master arbitration.

- **Bit 4 – USIDC: Data Output Collision**

This bit is logical one when bit 7 in the USI Data Register differs from the physical pin value. The flag is only valid when Two-wire mode is used. This signal is useful when implementing Two-wire bus master arbitration.

- **Bits 3:0 – USICNT3:0: Counter Value**

These bits reflect the current 4-bit counter value. The 4-bit counter value can directly be read or written by the CPU.

The 4-bit counter increments by one for each clock generated either by the external clock edge detector, by a Timer/Counter0 Compare Match, or by software using USICLK or USITC strobe bits. The clock source depends of the setting of the USICS1:0 bits. For external clock operation a special feature is added that allows the clock to be generated by writing to the USITC strobe bit. This feature is enabled by write a one to the USICLK bit while setting an external clock source (USICS1 = 1).

Note that even when no wire mode is selected (USIWM1:0 = 0) the external clock input (USCK/SCL) are can still be used by the counter.

13.5.4 USICR – USI Control Register

Bit	7	6	5	4	3	2	1	0	
0x0D (0x2D)	**USISIE**	**USIOIE**	**USIWM1**	**USIWM0**	**USICS1**	**USICS0**	**USICLK**	**USITC**	**USICR**
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	W	W	
Initial Value	0	0	0	0	0	0	0	0	

The Control Register includes interrupt enable control, wire mode setting, Clock Select setting, and clock strobe.

- **Bit 7 – USISIE: Start Condition Interrupt Enable**

Setting this bit to one enables the Start Condition detector interrupt. If there is a pending interrupt when the USISIE and the Global Interrupt Enable Flag is set to one, this will immediately be executed. Refer to the USISIF bit description on page 133 for further details.

- **Bit 6 – USIOIE: Counter Overflow Interrupt Enable**

Setting this bit to one enables the Counter Overflow interrupt. If there is a pending interrupt when the USIOIE and the Global Interrupt Enable Flag is set to one, this will immediately be executed. Refer to the USIOIF bit description on page 133 for further details.

- **Bits 5:4 – USIWM1:0: Wire Mode**

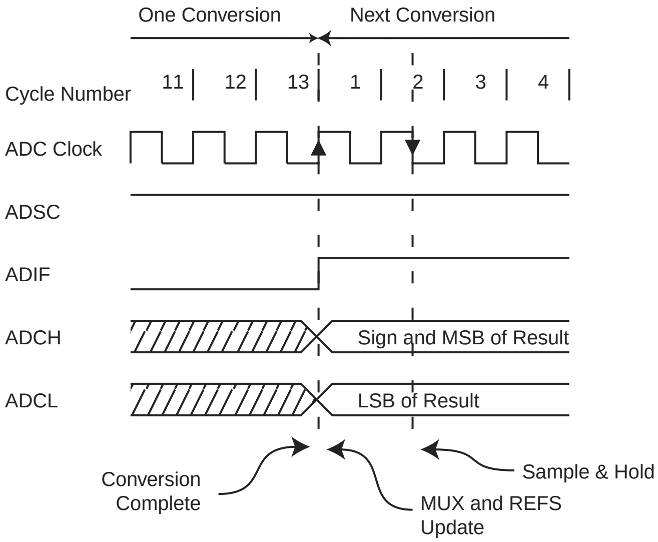
These bits set the type of wire mode to be used, as shown in Table 13-1 on page 134.

Basically, only the function of the outputs are affected by these bits. Data and clock inputs are not affected by the mode selected and will always have the same function. The counter and USI Data Register can therefore be clocked externally, and data input sampled, even when outputs are disabled.

Table 13-1. Relationship between USIWM1:0 and USI Operation

USIWM1	USIWM0	Description
0	0	Outputs, clock hold, and start detector disabled. Port pins operate as normal.
0	1	Three-wire mode. Uses DO, DI, and USCK pins. The *Data Output* (DO) pin overrides the corresponding bit in the PORTA register. However, the corresponding DDRA bit still controls the data direction. When the port pin is set as input the pin pull-up is controlled by the PORTA bit. The *Data Input* (DI) and *Serial Clock* (USCK) pins do not affect the normal port operation. When operating as master, clock pulses are software generated by toggling the PORTA register, while the data direction is set to output. The USITC bit in the USICR Register can be used for this purpose.

Figure 15-7. ADC Timing Diagram, Free Running Conversion



For a summary of conversion times, see Table 15-1.

Table 15-1. ADC Conversion Time

Condition	Sample & Hold (Cycles from Start of Conversion)	Total Conversion Time (Cycles)
First conversion	13.5	25
Normal conversions	1.5	13
Auto Triggered conversions	2	13.5

15.6 Changing Channel or Reference Selection

The MUX5:0 and REFS2:0 bits in the ADCSRB and ADMUX registers are single buffered through a temporary register to which the CPU has random access. This ensures that the channels and reference selection only takes place at a safe point during the conversion. The channel and reference selection is continuously updated until a conversion is started. Once the conversion starts, the channel and reference selection is locked to ensure a sufficient sampling time for the ADC. Continuous updating resumes in the last ADC clock cycle before the conversion completes (ADIF in ADCSRA is set). Note that the conversion starts on the following rising ADC clock edge after ADSC is written. The user is thus advised not to write new channel or reference selection values to ADMUX until one ADC clock cycle after ADSC is written.

If Auto Triggering is used, the exact time of the triggering event can be indeterministic. Special care must be taken when updating the ADMUX Register, in order to control which conversion will be affected by the new settings.

If both ADATE and ADEN is written to one, an interrupt event can occur at any time. If the ADMUX Register is changed in this period, the user cannot tell if the next conversion is based on the old or the new settings.

ADMUX can be safely updated in the following ways:

gain stage enables offset measurements. Selecting the single-ended channel ADC11 enables the temperature sensor. Refer to Table 15-4 for details.

Table 15-4. Input Channel Selections

MUX5:0	Single-Ended Input	Differential Input		Gain
		Positive	Negative	
000000	ADC0 (PA0)	NA	NA	NA
000001	ADC1 (PA1)			
000010	ADC2 (PA2)			
000011	ADC3 (PA4)			
000100	ADC4 (PA5)			
000101	ADC5 (PA6)			
000110	ADC6 (PA7)			
000111	ADC7 (PB4)			
001000	ADC8 (PB5)			
001001	ADC9 (PB6)			
001010	ADC10 (PB7)			
001011	NA	ADC0 (PA0)	ADC1 (PA1)	20x
001100		ADC0 (PA0)	ADC1 (PA1)	1x
001101		ADC1 (PA1)	ADC1 (PA1)	20x
001110		ADC2 (PA2)	ADC1 (PA1)	20x
001111		ADC2 (PA2)	ADC1 (PA1)	1x
010000	N/A	ADC2 (PA2)	ADC3 (PA4)	1x
010001		ADC3 (PA4)	ADC3 (PA4)	20x
010010		ADC4 (PA5)	ADC3 (PA4)	20x
010011		ADC4 (PA5)	ADC3 (PA4)	1x
010100	NA	ADC4 (PA5)	ADC5 (PA6)	20x
010101		ADC4 (PA5)	ADC5 (PA6)	1x
010110		ADC5 (PA6)	ADC5 (PA6)	20x
010111		ADC6 (PA7)	ADC5 (PA6)	20x
011000		ADC6 (PA7)	ADC5 (PA6)	1x
011001	NA	ADC8 (PB5)	ADC9 (PB6)	20x
011010		ADC8 (PB5)	ADC9 (PB6)	1x
011011		ADC9 (PB6)	ADC9 (PB6)	20x
011100		ADC10 (PB7)	ADC9 (PB6)	20x
011101		ADC10 (PB7)	ADC9 (PB6)	1x
011110	1.1V	N/A	N/A	N/A
011111	0V			

Table 15-5. ADC Prescaler Selections (Continued)

ADPS2	ADPS1	ADPS0	Division Factor
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

15.13.3 ADCL and ADCH – The ADC Data Register

15.13.3.1 ADLAR = 0

Bit	15	14	13	12	11	10	9	8	
0x05 (0x25)	–	–	–	–	–	–	**ADC9**	**ADC8**	**ADCH**
0x04 (0x24)	**ADC7**	**ADC6**	**ADC5**	**ADC4**	**ADC3**	**ADC2**	**ADC1**	**ADC0**	**ADCL**
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R	R	R	R	
	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

15.13.3.2 ADLAR = 1

Bit	15	14	13	12	11	10	9	8	
0x05 (0x25)	**ADC9**	**ADC8**	**ADC7**	**ADC6**	**ADC5**	**ADC4**	**ADC3**	**ADC2**	**ADCH**
0x04 (0x24)	**ADC1**	**ADC0**	–	–	–	–	–	–	**ADCL**
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R	R	R	R	
	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

When an ADC conversion is complete, the result is found in these two registers.

When ADCL is read, the ADC Data Register is not updated until ADCH is read. Consequently, if the result is left adjusted and no more than 8-bit precision is required, it is sufficient to read ADCH. Otherwise, ADCL must be read first, then ADCH.

The ADLAR bit in ADMUX, and the MUXn bits in ADMUX affect the way the result is read from the registers. If ADLAR is set, the result is left adjusted. If ADLAR is cleared (default), the result is right adjusted.

- **ADC9:0: ADC Conversion Result**

These bits represent the result from the conversion, as detailed in "ADC Conversion Result" on page 153.

18.2 Fuse Bytes

The ATtiny261/461/861 have three fuse bytes. Table 18-3, Table 18-4 and Table 18-5 describe briefly the functionality of all the fuses and how they are mapped into the fuse bytes. Note that the fuses are read as logical zero, “0”, if they are programmed.

Table 18-3. Fuse Extended Byte

Fuse High Byte	Bit No	Description	Default Value
	7	-	1 (unprogrammed)
	6	-	1 (unprogrammed)
	5	-	1 (unprogrammed)
	4	-	1 (unprogrammed)
	3	-	1 (unprogrammed)
	2	-	1 (unprogrammed)
	1	-	1 (unprogrammed)
SELFPRGEN (1)	0	Self-Programming Enable	1 (unprogrammed)

Notes: 1. Enables SPM instruction. See “Self-Programming the Flash” on page 165.

Table 18-4. Fuse High Byte

Fuse High Byte	Bit No	Description	Default Value
RSTDISBL (1)	7	External Reset disable	1 (unprogrammed)
DWEN (2)	6	DebugWIRE Enable	1 (unprogrammed)
SPIEN (3))	6	Enable Serial Program and Data Downloading	0 (programmed, SPI prog. enabled)
WDTON (4)	4	Watchdog Timer always on	1 (unprogrammed)
EESAVE	3	EEPROM memory is preserved through the Chip Erase	1 (unprogrammed, EEPROM not preserved)
BODLEVEL2 (5)	2	Brown-out Detector trigger level	1 (unprogrammed)
BODLEVEL1 (5)	1	Brown-out Detector trigger level	1 (unprogrammed)
BODLEVEL0 (5)	0	Brown-out Detector trigger level	1 (unprogrammed)

Notes:
1. See “Alternate Functions of Port B” on page 66 for description of RSTDISBL and DWEN Fuses. After programming the RSTDISBL fuse, parallel programming must be used to change fuses and allow further programming.
2. DWEN must be unprogrammed when Lock Bit security is required. See “Program And Data Memory Lock Bits” on page 170.
3. The SPIEN Fuse is not accessible in SPI programming mode.
4. Programming this fues will disable the Watchdog Timer Interrupt. See “WDTCR – Watchdog Timer Control Register” on page 47 for details.
5. See Table 19-6 on page 191 for BODLEVEL Fuse decoding.

19. Electrical Characteristics

19.1 Absolute Maximum Ratings*

Operating Temperature.................................. -55°C to +125°C

Storage Temperature .... -65°C to +150°C

Voltage on any Pin except $\overline{\text{RESET}}$
with respect to Ground ....-0.5V to V_{CC}+0.5V

Voltage on $\overline{\text{RESET}}$ with respect to Ground......-0.5V 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.

19.2 DC Characteristics

Table 19-1. DC Characteristics. T_A = -40°C to +85°C, V_{CC} = 1.8V to 5.5V (unless otherwise noted).

Symbol	Parameter	Condition	Min	Typ [(1)]	Max	Units
V_{IL}	Input Low-voltage	Except XTAL1 and $\overline{\text{RESET}}$ pins	-0.5		$0.2V_{CC}$(3)	V
		XTAL1 pin, External Clock Selected	-0.5		$0.1V_{CC}$(3)	V
		$\overline{\text{RESET}}$ pin	-0.5		$0.2V_{CC}$(3)	V
		$\overline{\text{RESET}}$ pin as I/O	-0.5		$0.2V_{CC}$(3)	V
V_{IH}	Input High-voltage	Except XTAL1 and $\overline{\text{RESET}}$ pins	$0.7V_{CC}$(2)		V_{CC} +0.5	V
		XTAL1 pin, External Clock Selected	$0.8V_{CC}$(2)		V_{CC} +0.5	V
		$\overline{\text{RESET}}$ pin	$0.9V_{CC}$(2)		V_{CC} +0.5	V
		$\overline{\text{RESET}}$ pin as I/O	$0.7V_{CC}$(2)		V_{CC} +0.5	V
V_{OL}	Output Low Voltage (4) (Except Reset pin) (6)	I_{OL} = 10 mA, V_{CC} = 5V I_{OL} = 5 mA, V_{CC} = 3V			0.6 0.5	V V
V_{OH}	Output High-voltage (5) (Except Reset pin) (6)	I_{OH} = -10 mA, V_{CC} = 5V I_{OH} = -5 mA, V_{CC} = 3V	4.3 2.5			V V
I_{IL}	Input Leakage Current I/O Pin	Vcc = 5.5V, pin low (absolute value)		< 0.05	1	µA
I_{IH}	Input Leakage Current I/O Pin	Vcc = 5.5V, pin high (absolute value)		< 0.05	1	µA
R_{RST}	Reset Pull-up Resistor		30		60	kΩ
R_{PU}	I/O Pin Pull-up Resistor		20		50	kΩ

Figure 20-20. I/O Pin Output Voltage vs. Sink Current (V_{CC} = 5V)

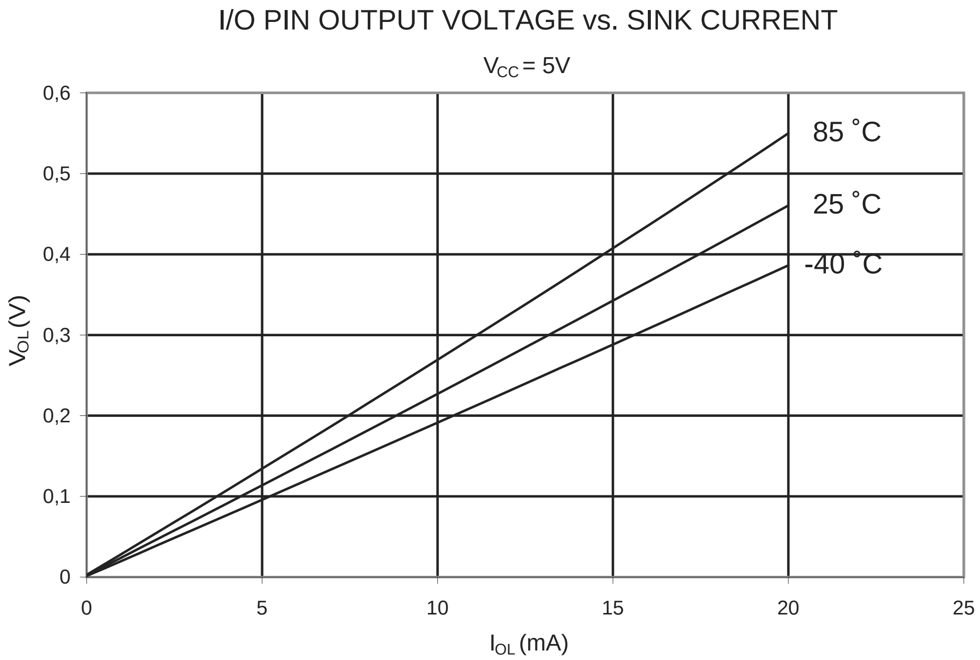


Figure 20-21. I/O Pin Output Voltage vs. Source Current (V_{CC} = 3V)

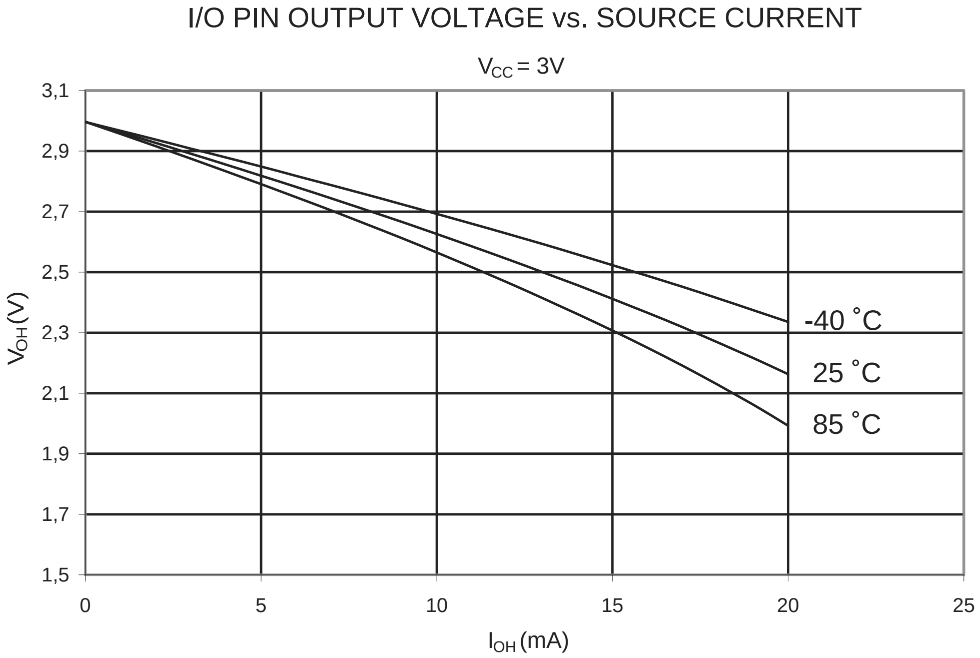
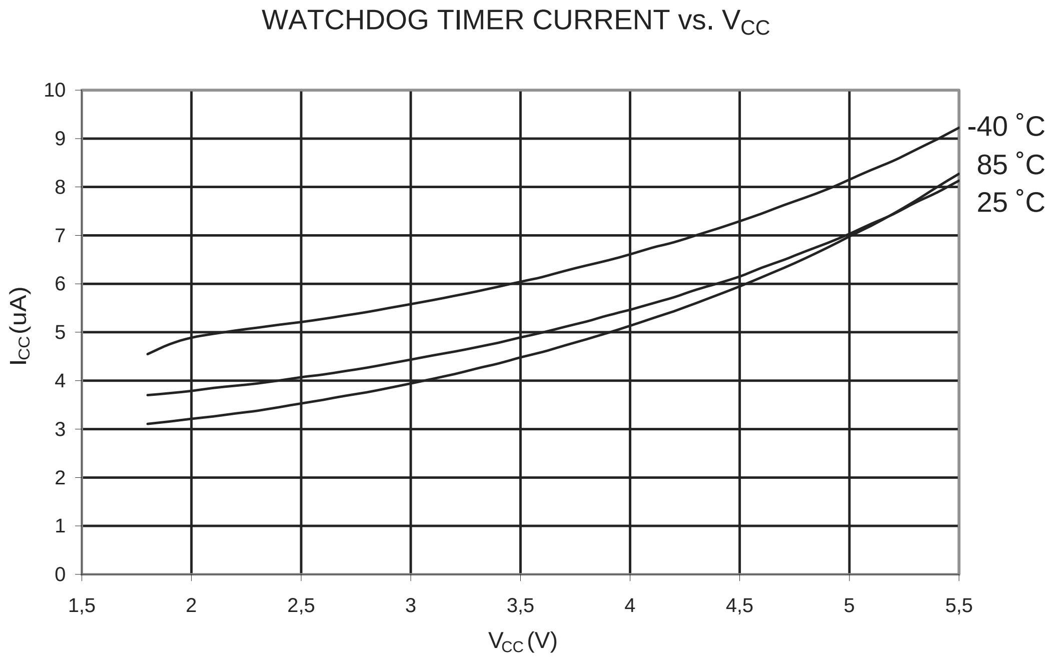


Figure 20-46. Watchdog Timer Current vs. V_{CC}



20.11 Current Consumption in Reset and Reset Pulsewidth

Figure 20-47. Reset Supply Current vs. Low Frequency (0.1 - 1.0 MHz, Excluding Current Through the Reset Pull-up)