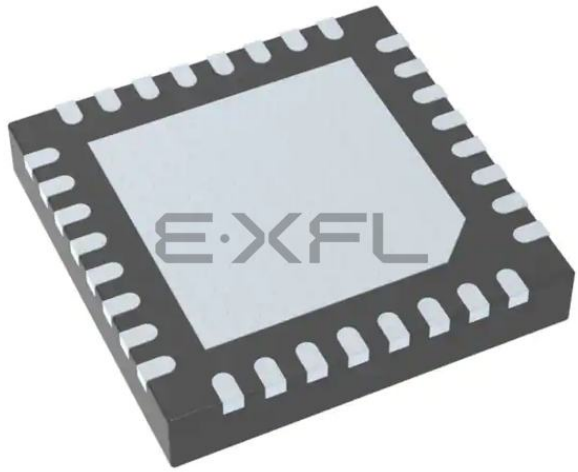


Welcome to E-XFL.COM

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	20MHz
Connectivity	USI
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, 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 ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	32-VFQFN Exposed Pad
Supplier Device Package	32-QFN (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/attiny261a-mf

5.3.1 EEPROM Read/Write Access

The EEPROM Access Registers are accessible in the I/O space.

The write access times for the EEPROM are given in Table 5-1 on page 22. A self-timing function, however, lets the user software detect when the next byte can be written. If the user code contains instructions that write the EEPROM, some precautions must be taken. In heavily filtered power supplies, V_{CC} is likely to rise or fall slowly on Power-up/down. This causes the device for some period of time to run at a voltage lower than specified as minimum for the clock frequency used. See "Preventing EEPROM Corruption" on page 19 for details on how to avoid problems in these situations.

In order to prevent unintentional EEPROM writes, a specific write procedure must be followed. Refer to "Atomic Byte Programming" on page 17 and "Split Byte Programming" on page 17 for details on this.

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

5.3.2 Atomic Byte Programming

Using Atomic Byte Programming is the simplest mode. When writing a byte to the EEPROM, the user must write the address into the EEARL Register and data into EEDR Register. If the EEPMn bits are zero, writing EEPE (within four cycles after EEMPE is written) will trigger the erase/write operation. Both the erase and write cycle are done in one operation and the total programming time is given in Table 5-1 on page 22. The EEPE bit remains set until the erase and write operations are completed. While the device is busy with programming, it is not possible to do any other EEPROM operations.

5.3.3 Split Byte Programming

It is possible to split the erase and write cycle in two different operations. This may be useful if the system requires short access time for some limited period of time (typically if the power supply voltage falls). In order to take advantage of this method, it is required that the locations to be written have been erased before the write operation. But since the erase and write operations are split, it is possible to do the erase operations when the system allows doing time-critical operations (typically after Power-up).

5.3.4 Erase

To erase a byte, the address must be written to EEAR. If the EEPMn bits are 0b01, writing the EEPE within four cycles after EEMPE is written will trigger the erase operation only (programming time is given in Table 5-1 on page 22). The EEPE bit remains set until the erase operation completes. While the device is busy programming, it is not possible to do any other EEPROM operations.

5.3.5 Write

To write a location, the user must write the address into EEAR and the data into EEDR. If the EEPMn bits are 0b10, writing the EEPE (within four cycles after EEMPE is written) will trigger the write operation only (programming time is given in Table 5-1 on page 22). The EEPE bit remains set until the write operation completes. If the location to be written has not been erased before write, the data that is stored must be considered as lost. While the device is busy with programming, it is not possible to do any other EEPROM operations.

When the PLL output is selected as clock source, the start-up times are determined by SUT fuse bits as shown in Table 6-5.

Table 6-5. Start-up Times for the PLLCK

SUT[1:0]	Start-up Time from Power Down	Additional Delay from Power-On-Reset (V_{CC} = 5.0V)	Recommended usage
00	14CK + 1K (1024) + 4 ms	4 ms	BOD enabled
01	14CK + 16K (16384) + 4 ms	4 ms	Fast rising power
10	14CK + 1K (1024) + 64 ms	4 ms	Slowly rising power
11	14CK + 16K (16384) + 64 ms	4 ms	Slowly rising power

6.2.3 Calibrated Internal 8 MHz Oscillator

By default, the Internal Oscillator provides an approximately 8 MHz clock signal. Though voltage and temperature dependent, this clock can be very accurately calibrated by the user. See Table 19-2 on page 187 and "Internal Oscillators" on page 222 for more details. The device is shipped with the CKDIV8 Fuse programmed. See "System Clock Prescaler" on page 31 for more details.

This clock may be selected as the system clock by programming the CKSEL Fuses as shown in Table 6-6. If selected, it will operate with no external components. During reset, hardware loads the pre-programmed calibration value into the OSCCAL Register and thereby automatically calibrates the internal oscillator. The accuracy of this calibration is shown as Factory calibration in Table 19-2 on page 187.

Table 6-6. Internal Calibrated Oscillator Operating Modes

CKSEL[3:0]	Nominal Frequency
0010 (1)	8.0 MHz (2)

Notes: 1. The device is shipped with this option selected.

2. If the oscillator frequency exceeds the specification of the device (depends on V_{CC}), the CKDIV8 Fuse can be programmed to divide the internal frequency by 8.

When this oscillator is selected, start-up times are determined by SUT fuses as shown in Table 6-7.

Table 6-7. Start-up Times for the Internal Calibrated Oscillator Clock Selection

SUT[1:0]	Start-up Time from Power-down	Additional Delay from Reset (V_{CC} = 5.0V)	Recommended Usage
00	6 CK	14CK(1)	BOD enabled
01	6 CK	14CK + 4 ms	Fast rising power
10(2)	6 CK	14CK + 64 ms	Slowly rising power
11	Reserved		

Note: 1. If the RSTDISBL fuse is programmed, this start-up time will be increased to 14CK + 4 ms to ensure programming mode can be entered.

2. The device is shipped with this option selected.

Table 10-4 and Table 10-5 relate the alternate functions of Port A to the overriding signals shown in Figure 10-5 on page 60.

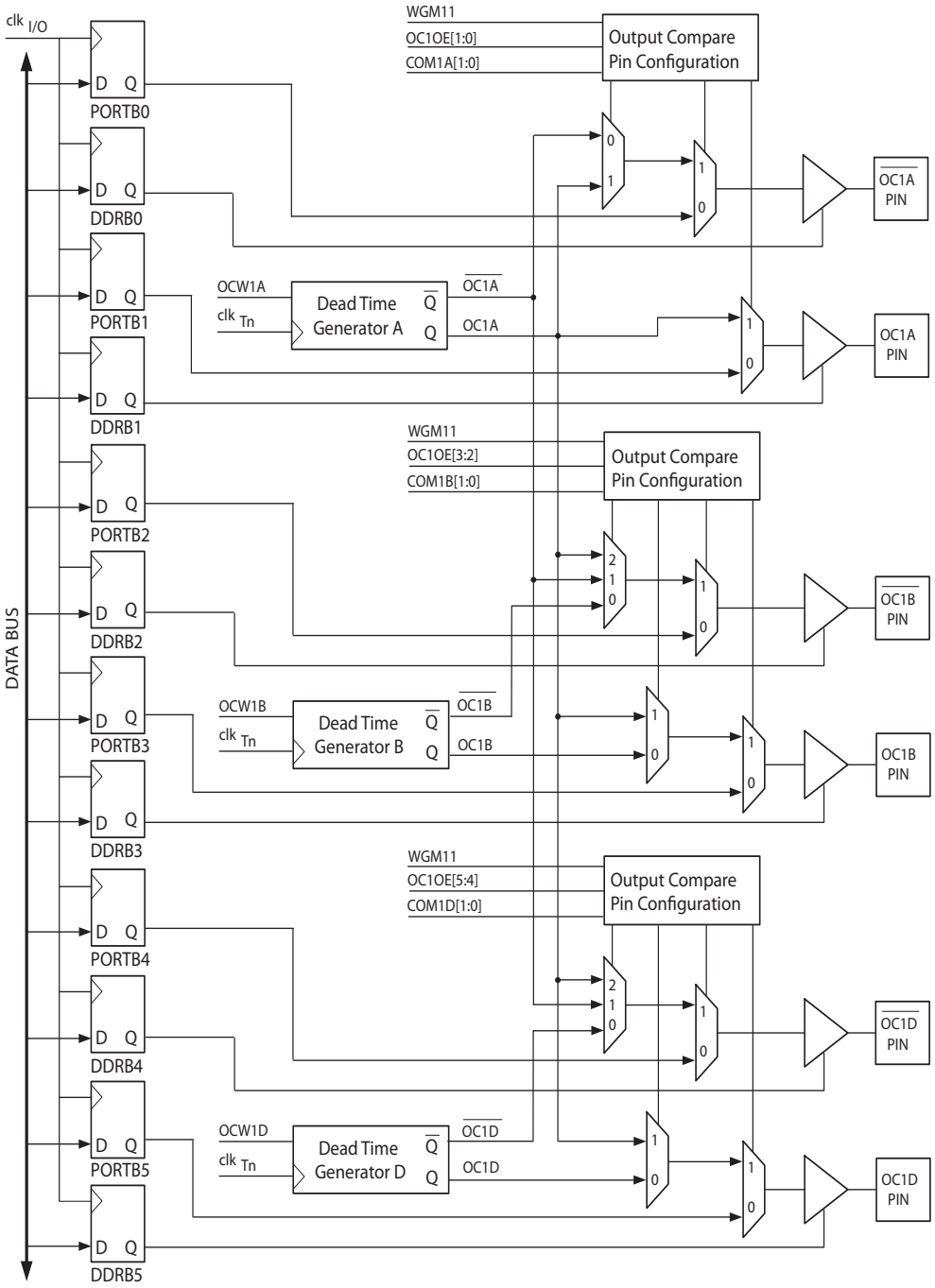
Table 10-4. Overriding Signals for Alternate Functions in PA[7:4]

Signal Name	PA7/ADC6/AIN0/ PCINT7	PA6/ADC5/AIN1/ PCINT6	PA5/ADC4/AIN2/ PCINT5	PA4/ADC3/ICP0/ PCINT4
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	0	0	0
DDOV	0	0	0	0
PVOE	0	0	0	0
PVOV	0	0	0	0
PTOE	0	0	0	0
DIEOE	PCINT7 • PCIE + ADC6D	PCINT6 • PCIE + ADC5D	PCINT5 • PCIE + ADC4D	PCINT4 • PCIE + ADC3D
DIEOV	$\overline{\text{ADC6D}}$	$\overline{\text{ADC5D}}$	$\overline{\text{ADC4D}}$	$\overline{\text{ADC3D}}$
DI	PCINT7	PCINT6	PCINT5	ICP0/PCINT4
AIO	ADC6, AIN0	ADC5, AIN1	ADC4, AIN2	ADC3

Table 10-5. Overriding Signals for Alternate Functions in PA[3:0]

Signal Name	PA3/AREF/ PCINT3	PA2/ADC2/INT1/ USCK/SCL/PCINT2	PA1/ADC1/DO/ PCINT1	PA0/ADC0/DI/SDA/ PCINT0
PUOE	0	0	0	0
PUOV	0	0	0	0
DDOE	0	USI_TWO_WIRE • USIPOS	0	USI_TWO_WIRE • USIPOS
DDOV	0	(USI_SCL_HOLD + $\overline{\text{PORTB2}}$) • DDB2 • USIPOS	0	($\overline{\text{SDA}}$ + $\overline{\text{PORTB0}}$) • DDRB0 • USIPOS
PVOE	0	USI_TWO_WIRE • DDRB2	USI_THREE_WIRE • USIPOS	USI_TWO_WIRE • DDRB0 • USIPOS
PVOV	0	0	DO • USIPOS	0
PTOE	0	USI_PTOE • USIPOS	0	0
DIEOE	PCINT3 • PCIE	PCINT2 • PCIE + INT1 + ADC2D + USISIE • USIPOS	PCINT1 • PCIE + ADC1D	PCINT0 • PCIE + ADC0D + USISIE • USIPOS
DIEOV	0	$\overline{\text{ADC2D}}$	ADC1D	ADC0D
DI	PCINT3	USCK/SCL/INT1/ PCINT2	PCINT1	DI/SDA/PCINT0
AIO	AREF	ADC2	ADC1	ADC0

Figure 12-10. Compare Match Output Unit, Schematic



The general I/O port function is overridden by the Output Compare (OC1x / $\overline{OC1x}$) from the Dead Time Generator if either of the COM1x[1:0] bits are set. However, the OC1x pin direction (input or output) is still controlled by the Data Direction Register (DDR) for the port pin. The Data Direction Register bit for the OC1x and $\overline{OC1x}$ pins (DDR_OC1x and DDR_$\overline{OC1x}$) must be set as output before the OC1x and $\overline{OC1x}$ values are visible on the pin. The port override function is independent of the Output Compare mode.

The design of the Output Compare Pin Configuration logic allows initialization of the OC1x state before the output is enabled. Note that some COM1x[1:0] bit settings are reserved for certain modes of operation. For Output Compare Pin Configurations refer to Table 12-2 on page 99, Table 12-3 on page 101, Table 12-4 on page 103, Table 12-5 on page 104, Table 12-6 on page 104, and Table 12-7 on page 105.

12.7.1 Compare Output Mode and Waveform Generation

The Waveform Generator uses the COM1x[1:0] bits differently in Normal mode and PWM modes. For all modes, setting the COM1x[1:0] = 0 tells the Waveform Generator that no action on the OCW1x Output is to be performed on the next Compare Match. For compare output actions in the non-PWM modes refer to Table 12-8 on page 111. For fast PWM mode, refer to Table 12-9 on page 111, and for the Phase and Frequency Correct PWM refer to Table 12-10 on page 112. A change of the COM1x[1:0] bits state will have effect at the first Compare Match after the bits are written. For non-PWM modes, the action can be forced to have immediate effect by using the FOC1x strobe bits.

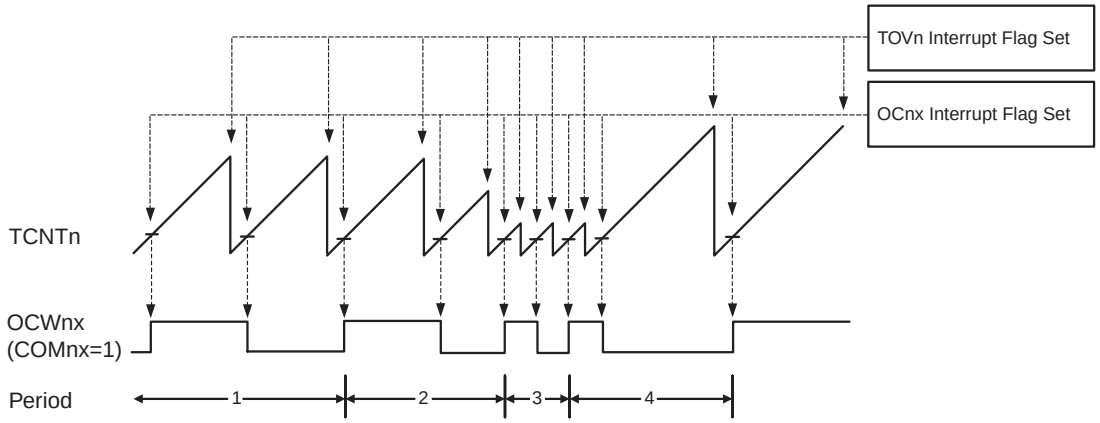
12.8 Modes of Operation

The mode of operation, i.e., the behavior of the Timer/Counter and the Output Compare pins, is defined by the combination of waveform generation mode bits (PWM1A, PWM1B, and WGM1[1:0]) and compare output mode bits (COM1x[1:0]). The Compare Output mode bits do not affect the counting sequence, while the Waveform Generation mode bits do. The COM1x[1:0] bits control whether the PWM output generated should be inverted, non-inverted or complementary. For non-PWM modes the COM1x[1:0] bits control whether the output should be set, cleared, or toggled at a Compare Match.

12.8.1 Normal Mode

The simplest mode of operation is Normal mode (PWM1A/PWM1B = 0), where the counter counts from BOTTOM to TOP (defined as OCR1C) then restarts from BOTTOM. The OCR1C defines the TOP value for the counter, hence also its resolution, and allows control of the Compare Match output frequency. In toggle Compare Output Mode the Waveform Output (OCW1x) is toggled at Compare Match between TCNT1 and OCR1x. In non-inverting Compare Output Mode the Waveform Output is cleared on the Compare Match. In inverting Compare Output Mode the Waveform Output is set on Compare Match. The timing diagram for Normal mode is shown in Figure 12-11.

Figure 12-11. Normal Mode, Timing Diagram



- **Bits 5:4 – COM1B[1:0]: 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 COM1B[1:0] bits are set, the OC1B output overrides the normal port functionality of the I/O pin it is connected to. The complementary $\overline{OC1B}$ output is connected only in PWM modes when the COM1B[1:0] 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 COM1B[1:0] bits depends on the PWM1B and WGM1[1:0] bit settings. Table 12-13 shows the COM1B[1:0] bit functionality when the PWM1B bit is set to Normal Mode (non-PWM).

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

COM1B[1:0]	OCW1B Behaviour	OC1B Pin	$\overline{OC1B}$ 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 COM1B[1:0] bit functionality when the PWM1B and WGM1[1:0] bits are set to Fast PWM Mode.

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

COM1B[1:0]	OCW1B Behaviour	OC1B Pin	$\overline{OC1B}$ 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 COM1B[1:0] bit functionality when the PWM1B and WGM1[1:0] bits are set to Phase and Frequency Correct PWM Mode.

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

COM1B[1:0]	OCW1B Behaviour	OC1B Pin	$\overline{OC1B}$ 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

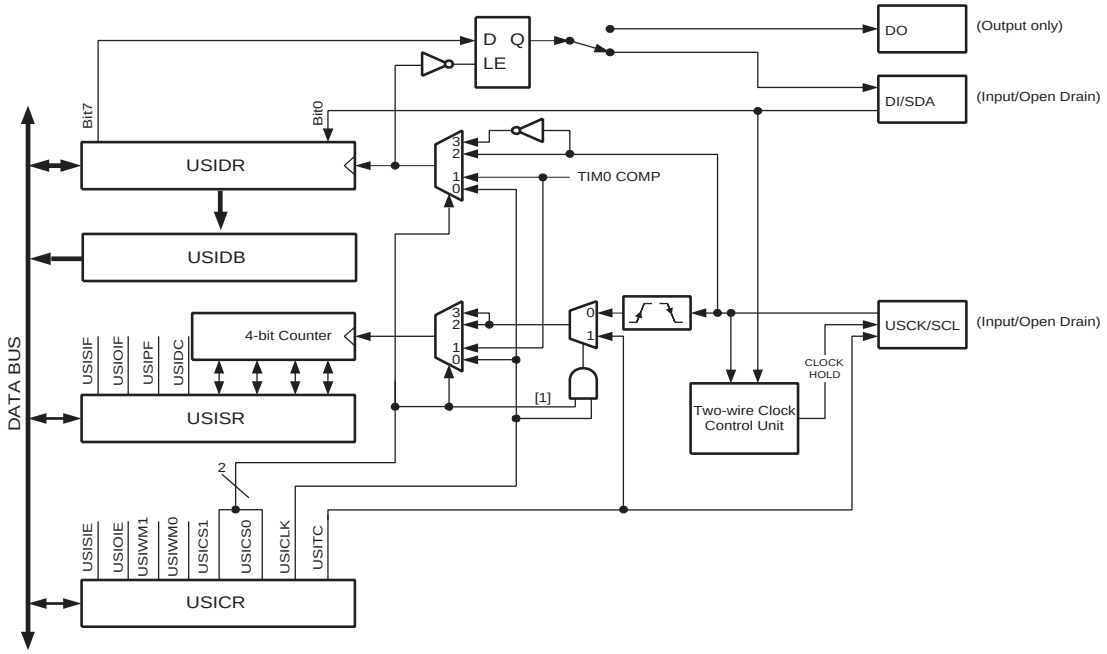
- **Two-wire Synchronous Data Transfer (Master or Slave)**
- **Three-wire Synchronous Data Transfer (Master or Slave)**
- **Data Received Interrupt**
- **Wakeup from Idle Mode**
- **In Two-wire Mode: Wake-up from All Sleep Modes, Including Power-down Mode**
- **Two-wire Start Condition Detector with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

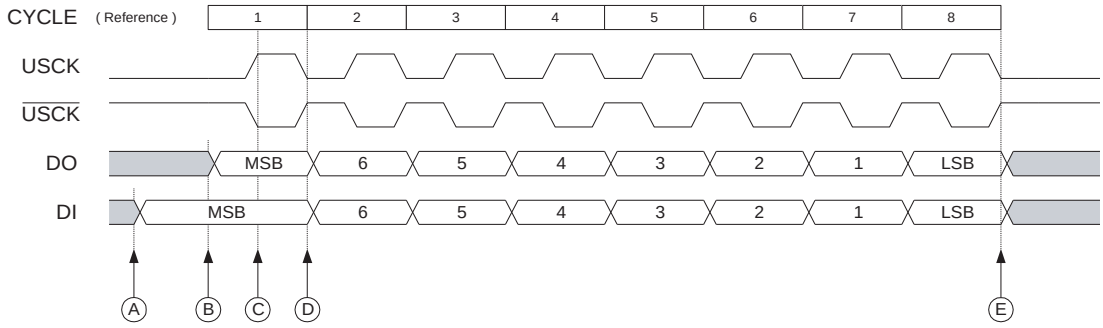
A simplified block diagram of the USI is shown in Figure 13-1 For actual placement of I/O pins refer to “Pinout ATtiny261A/461A/861A” on page 2. Device-specific I/O Register and bit locations are listed in the “Register Descriptions” on page 131.

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)
SPITransfer_loop:
    sts    USICR,r16
    lds    r16, USISR
```

13.3.3 SPI Slave Operation Example

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

```
init:
    ldi    r16,(1<<USIWM0)|(1<<USICS1)
    sts    USICR,r16
...
SlaveSPITransfer:
    sts    USIDR,r16
    ldi    r16,(1<<USIOIF)
    sts    USISR,r16
SlaveSPITransfer_loop:
    lds    r16, USISR
    sbrs   r16, USIOIF
    rjmp   SlaveSPITransfer_loop
    lds    r16,USIDR
    ret
```

The code is size optimized using only eight instructions (+ ret). The code example assumes that the DO is configured as output and USCK pin is configured as input in the DDR Register. The value stored in register r16 prior to the function is called is transferred to the master device, and when the transfer is completed the data received from the Master is stored back into the r16 Register.

Note that the first two instructions are for initialization, only, and need only be executed once. These instructions set three-wire mode and positive edge clock. The loop is repeated until the USI Counter Overflow Flag is set.

13.3.4 Two-wire Mode

The USI Two-wire mode is compliant to the Inter IC (TWI) bus protocol, but without slew rate limiting on outputs and input noise filtering. Pin names used by this mode are SCL and SDA.

Figure 13-4 on page 129 shows two USI units operating in two-wire mode, one as master and one as slave. It is only the physical layer that is shown since the system operation is highly dependent of the communication scheme used. The main differences between the master and slave operation at this level is the serial clock generation which is always done by the master. Only the slave uses the clock control unit.

Clock generation must be implemented in software, but the shift operation is done automatically in both devices. Note that clocking only on negative edges for shifting data is of practical use in this mode. The slave can insert wait states at start or end of transfer by forcing the SCL clock low. This means that the master must always check if the SCL line was actually released after it has generated a positive edge.

Since the clock also increments the counter, a counter overflow can be used to indicate that the transfer is completed. The clock is generated by the master by toggling the USCK pin via the PORTA register.

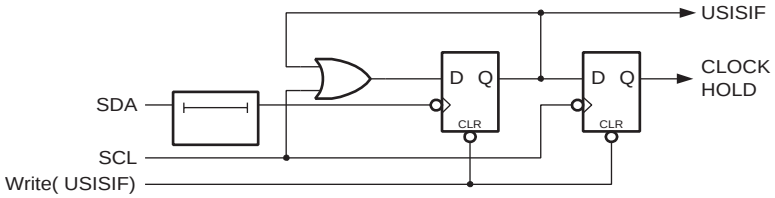
3. The master set the first bit to be transferred and releases the SCL line (C). The slave samples the data and shifts it into the USI Data Register at the positive edge of the SCL clock.
4. After eight bits containing slave address and data direction (read or write) have been transferred, the slave counter overflows and the SCL line is forced low (D). If the slave is not the one the master has addressed, it releases the SCL line and waits for a new start condition.
5. When the slave is addressed, it holds the SDA line low during the acknowledgment cycle before holding the SCL line low again (i.e., the USI Counter Register must be set to 14 before releasing SCL at (D)). Depending on the R/W bit the master or slave enables its output. If the bit is set, a master read operation is in progress (i.e., the slave drives the SDA line) The slave can hold the SCL line low after the acknowledge (E).
6. Multiple bytes can now be transmitted, all in same direction, until a stop condition is given by the master (F), or a new start condition is given.

If the slave is not able to receive more data it does not acknowledge the data byte it has last received. When the master does a read operation it must terminate the operation by forcing the acknowledge bit low after the last byte transmitted.

13.3.5 Start Condition Detector

The start condition detector is shown in Figure 13-6. The SDA line is delayed (in the range of 50 to 300 ns) to ensure valid sampling of the SCL line. The start condition detector is only enabled in Two-wire mode.

Figure 13-6. Start Condition Detector, Logic Diagram



The start condition detector works asynchronously and can therefore wake up the processor from power-down sleep mode. However, the protocol used might have restrictions on the SCL hold time. Therefore, when using this feature in this case the Oscillator start-up time set by the CKSEL Fuses (see “Clock System” on page 24) must also be taken into the consideration. Refer to the USISIF bit description on page Page 132 for further details.

13.3.6 Clock speed considerations

Maximum frequency for SCL and SCK is f_{CK} / 2. This is also the maximum data transmit and receive rate in both two- and three-wire mode. In two-wire slave mode the Two-wire Clock Control Unit will hold the SCL low until the slave is ready to receive more data. This may reduce the actual data rate in two-wire mode.

13.4 Alternative USI Usage

The flexible design of the USI allows it to be used for other tasks when serial communication is not needed. Below are some examples.

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
I_{IH}	Input Leakage Current I/O Pin	V_{CC} = 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
I_{CC}	Power Supply Current [7]	Active 1MHz, V_{CC} = 2V		0.2	0.5	mA
		Active 4MHz, V_{CC} = 3V		1.2	2	mA
		Active 8MHz, V_{CC} = 5V		3.6	7	mA
		Idle 1MHz, V_{CC} = 2V		0.04	0.15	mA
		Idle 4MHz, V_{CC} = 3V		0.25	0.4	mA
		Idle 8MHz, V_{CC} = 5V		0.9	1.5	mA
	Power-down mode [8]	WDT enabled, V_{CC} = 3V		4	10	µA
		WDT disabled, V_{CC} = 3V		0.15	2	µA

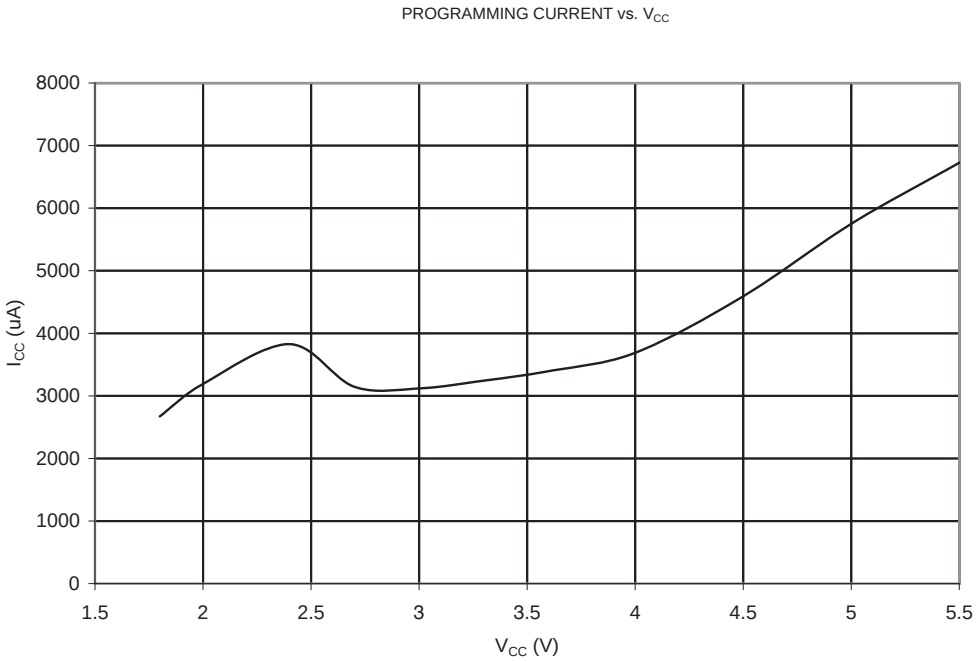
Notes:
1. Typical values at +25°C.
2. “Min” means the lowest value where the pin is guaranteed to be read as high.
3. “Max” means the highest value where the pin is guaranteed to be read as low.
4. Although each I/O port can sink more than the test conditions (10 mA at V_{CC} = 5V, 5 mA at V_{CC} = 3V) under steady state conditions (non-transient), the sum of all I_{OL} (for all ports) should not exceed 100 mA. If I_{OL} exceeds the test conditions, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test condition.
5. Although each I/O port can source more than the test conditions (10 mA at V_{CC} = 5V, 5 mA at V_{CC} = 3V) under steady state conditions (non-transient), the sum of all I_{OH} (for all ports) should not exceed 100 mA. If I_{OH} exceeds the test condition, V_{OH} may exceed the related specification. Pins are not guaranteed to source current greater than the listed test condition.
6. The $\overline{\text{RESET}}$ pin must tolerate high voltages when entering and operating in programming modes and, as a consequence, has a weak drive strength as compared to regular I/O pins. See Figure 20-32, Figure 20-33, Figure 20-34, and Figure 20-35 (starting on page 214).
7. Values are with external clock using methods described in “Minimizing Power Consumption” on page 37. Power Reduction is enabled (PRR = 0xFF) and there is no I/O drive.
8. BOD Disabled.

Table 19-9. ADC Characteristics, Differential Channels (Bipolar Mode). T = -40°C to +85°C

Symbol	Parameter	Condition	Min	Typ	Max	Units
	Resolution	Gain = 1x / 8x / 20x / 32x			10	Bits
	Absolute accuracy (Including INL, DNL, and Quantization, Gain and Offset Errors)	Gain = 1x / 8x V_{REF} = 4V, V_{CC} = 5V ADC clock = 50 - 200 kHz		8		LSB
		Gain = 20x / 32x V_{REF} = 4V, V_{CC} = 5V ADC clock = 50 - 200 kHz		8		LSB
	Integral Non-Linearity (INL) (Accuracy after Offset and Gain Calibration)	Gain = 1x / 8x V_{REF} = 4V, V_{CC} = 5V ADC clock = 50 - 200 kHz		4		LSB
		Gain = 20x / 32x V_{REF} = 4V, V_{CC} = 5V ADC clock = 50 - 200 kHz		5		LSB
	Gain Error	Gain = 1x / 8x		4		LSB
		Gain = 20x / 32x		5		LSB
	Offset Error	Gain = 1x / 8x V_{REF} = 4V, V_{CC} = 5V ADC clock = 50 - 200 kHz		3		LSB
		Gain = 20x / 32x V_{REF} = 4V, V_{CC} = 5V ADC clock = 50 - 200 kHz		4		LSB
	Conversion Time	Free Running Conversion	65		260	µs
	Clock Frequency		50		200	kHz
V_{IN}	Input Voltage		GND		AV_{CC} (1)	V
V_{DIFF}	Input Differential Voltage				V_{REF}/Gain	V
	Input Bandwidth			4		kHz
AV_{CC}	Analog Supply Voltage		V_{CC} - 0.3		V_{CC} + 0.3	V
A_{REF}	External Voltage Reference		2.0		AVCC - 1.0	V
V_{INT}	Internal 1.1V Reference		1.0	1.1	1.2	V
	Internal 2.56V Reference	V_{CC} > 3.0V	2.3	2.56	2.8	V
R_{REF}	Reference Input Resistance			35		kΩ
R_{AIN}	Analog Input Resistance			100		MΩ
	ADC Conversion Output		-512		511	LSB

Notes: 1. V_{DIFF} must be below V_{REP}

Figure 20-19. Programming Current vs. V_{CC}



20.2.6 Pull-up Resistors

Figure 20-20. Pull-Up Resistor Current vs. Input Voltage (I/O Pin, V_{CC} = 1.8V)

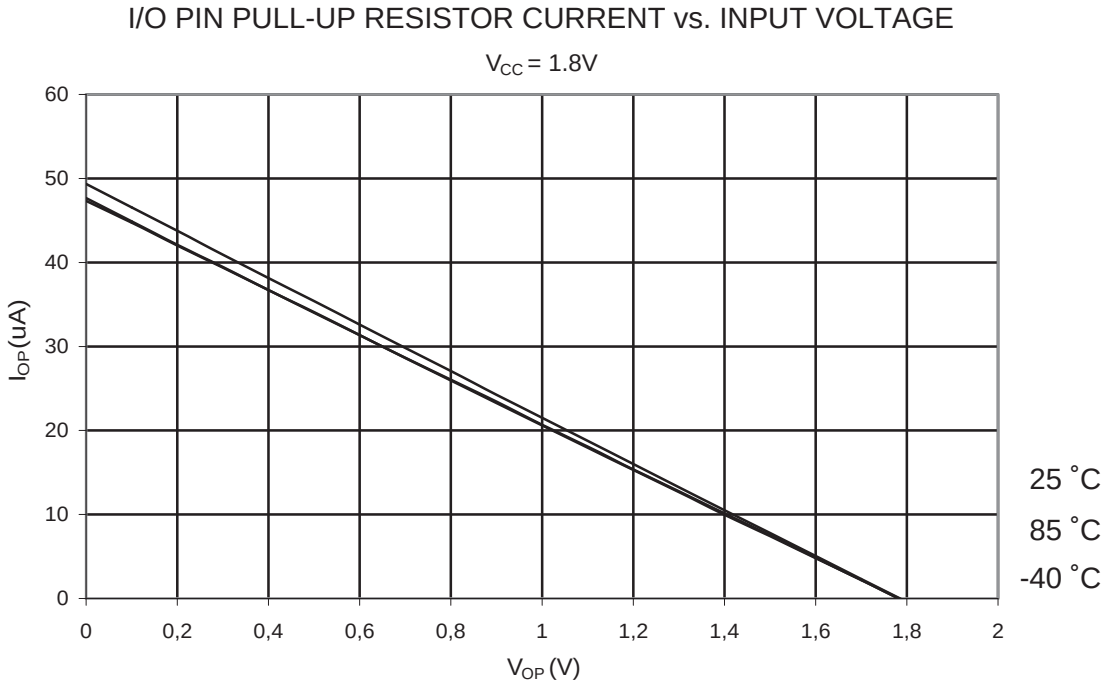


Figure 20-21. Pull-Up Resistor Current vs. Input Voltage (I/O Pin, V_{CC} = 3V)

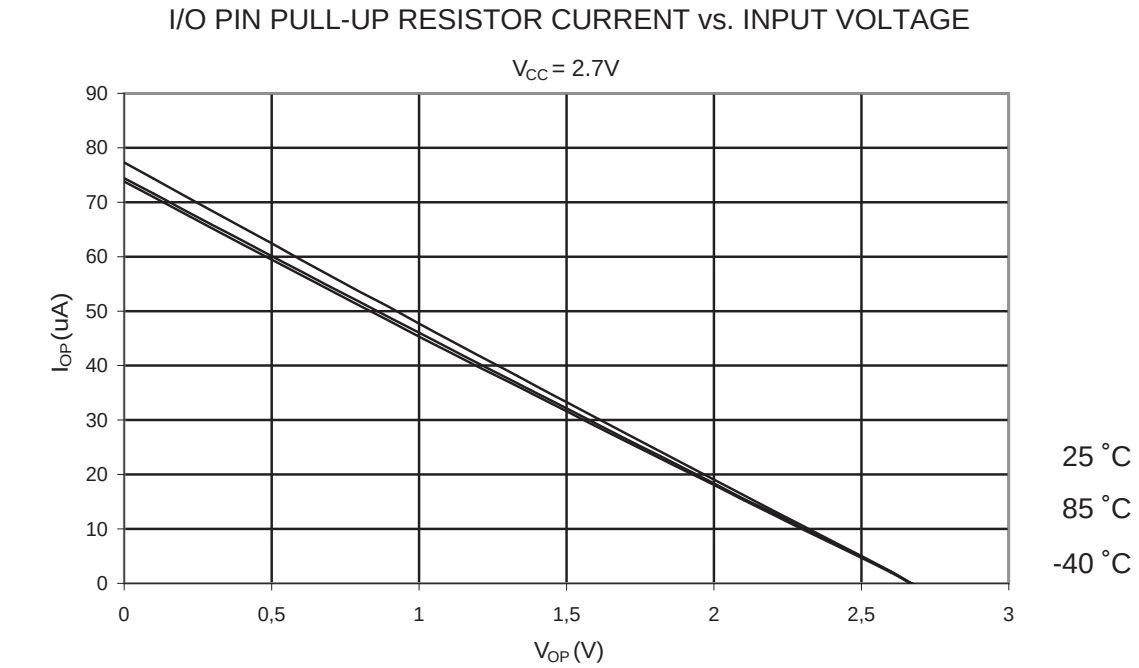


Figure 20-22. Pull-Up Resistor Current vs. Input Voltage (I/O Pin, V_{CC} = 5V)

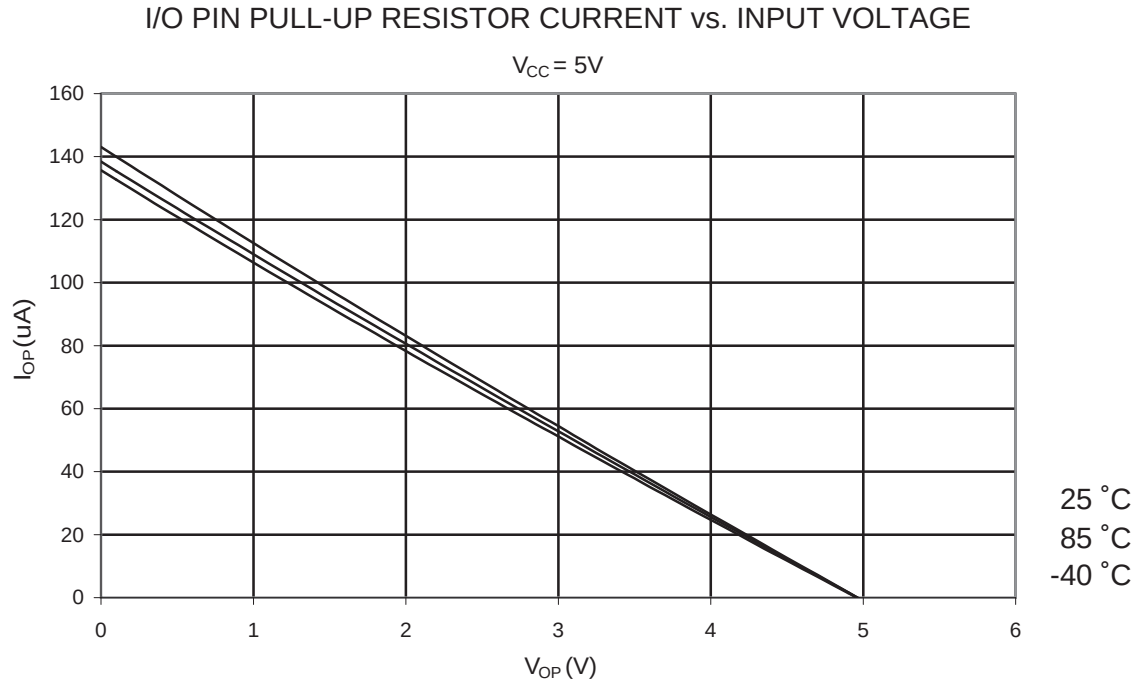


Figure 20-23. Pull-Up Resistor Current vs. Input Voltage (Reset Pin, V_{CC} = 1.8V)

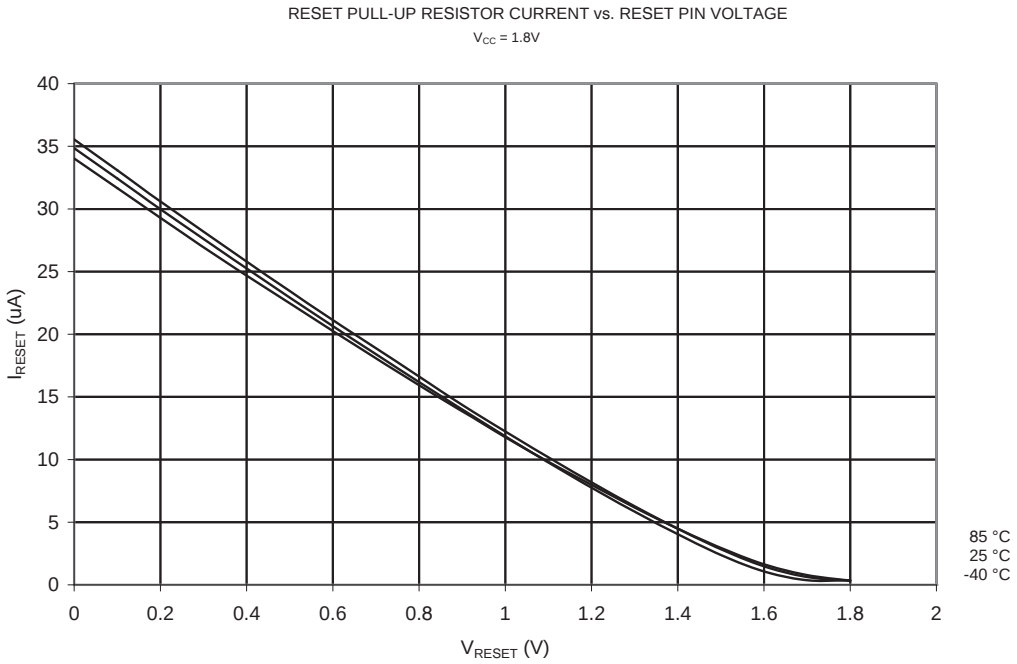


Figure 20-24. Pull-Up Resistor Current vs. Input Voltage (Reset Pin, V_{CC} = 3V)

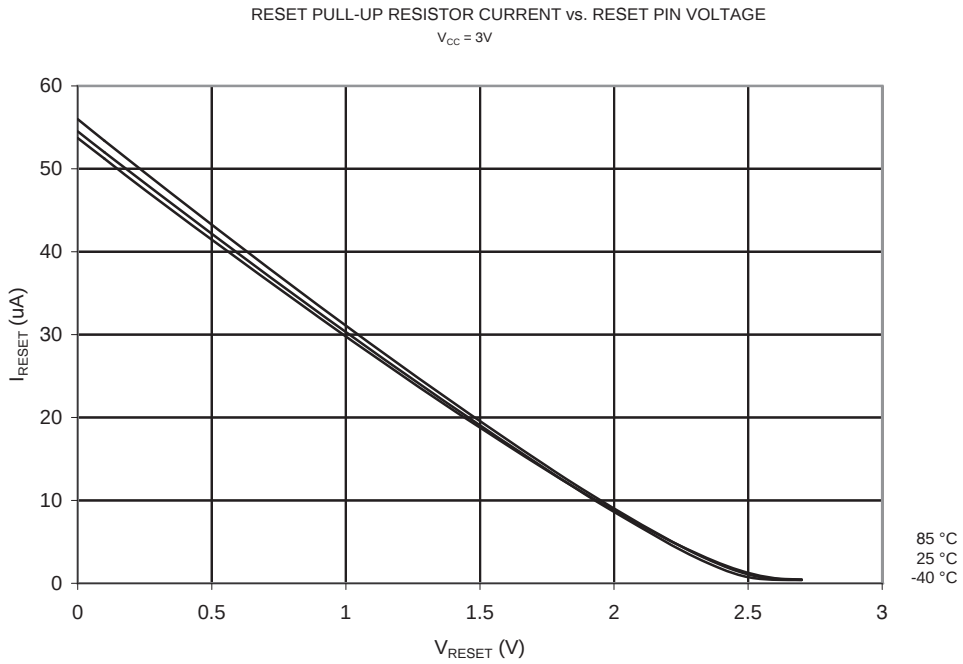
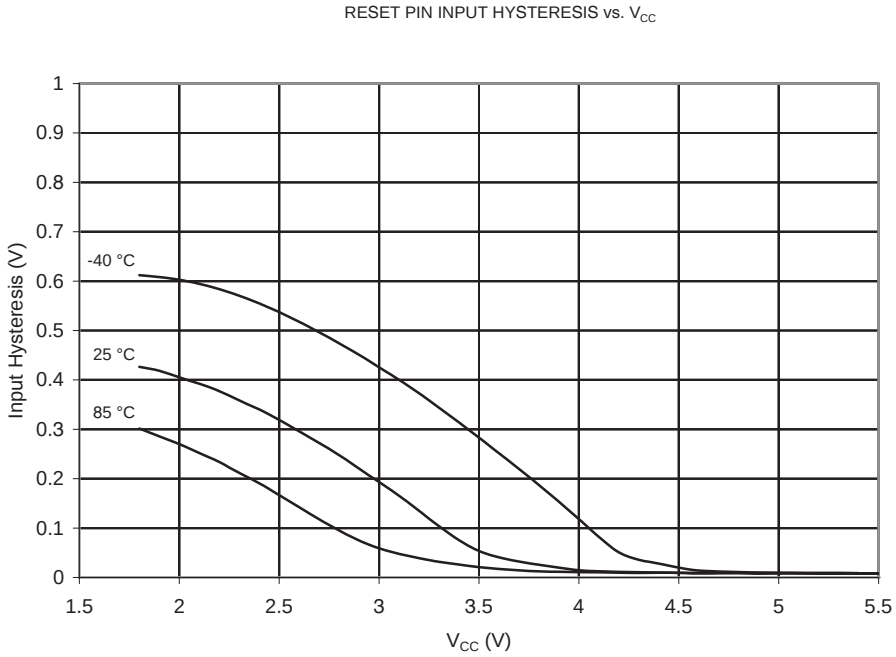


Figure 20-41. V_{IH}-V_{IL}: Input Hysteresis vs. V_{CC} (Reset Pin)



20.2.9 BOD, Bandgap and Reset

Figure 20-42. BOD Threshold vs. Temperature (BOD Level set to 4.3V)

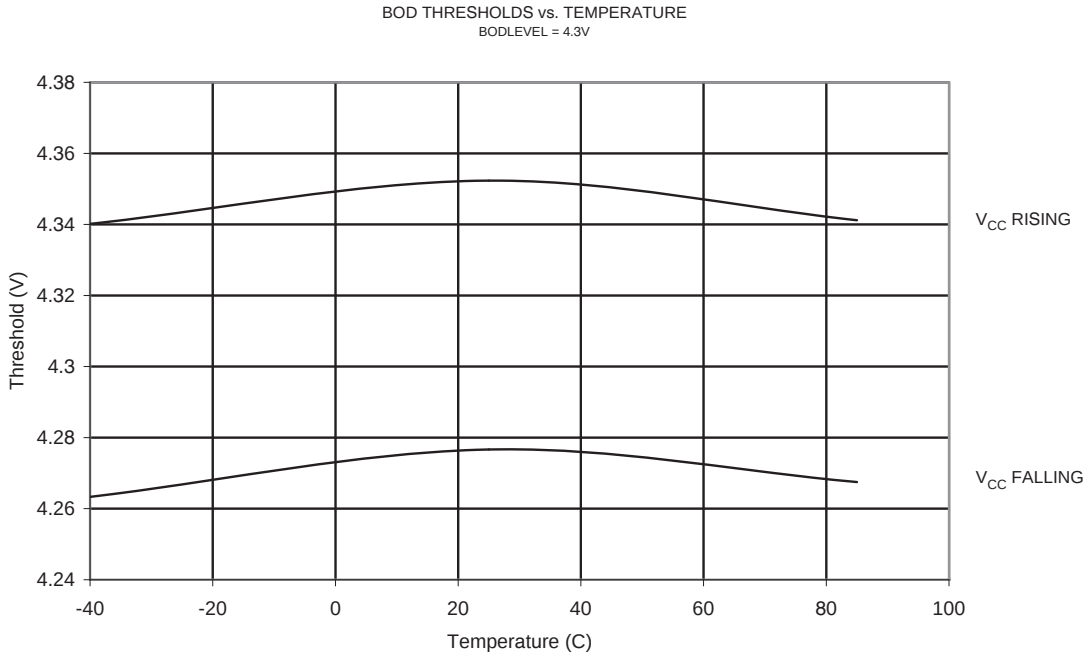
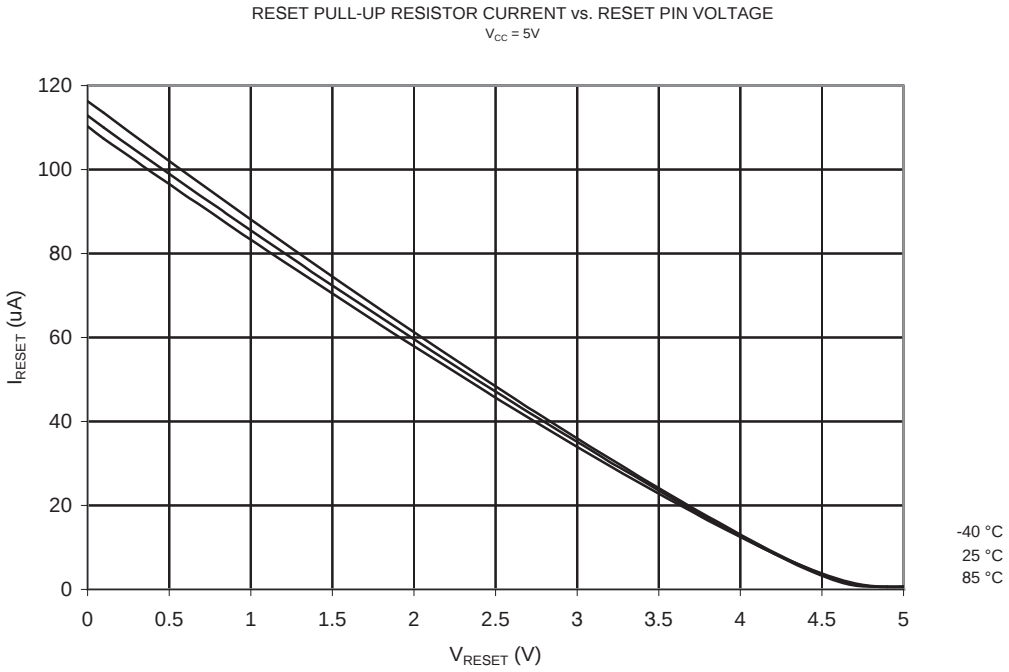


Figure 20-76. Pull-Up Resistor Current vs. Input Voltage (Reset Pin, V_{CC} = 5V)



20.3.7 Output Driver Strength

Figure 20-77. V_{OL}: Output Voltage vs. Sink Current (I/O Pin, V_{CC} = 1.8V)

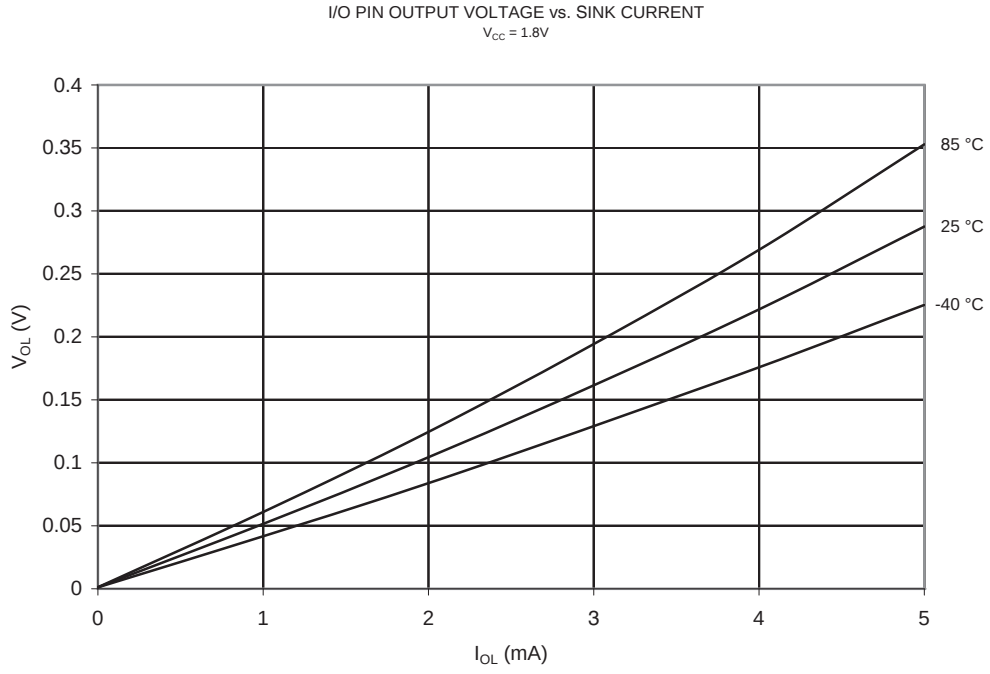
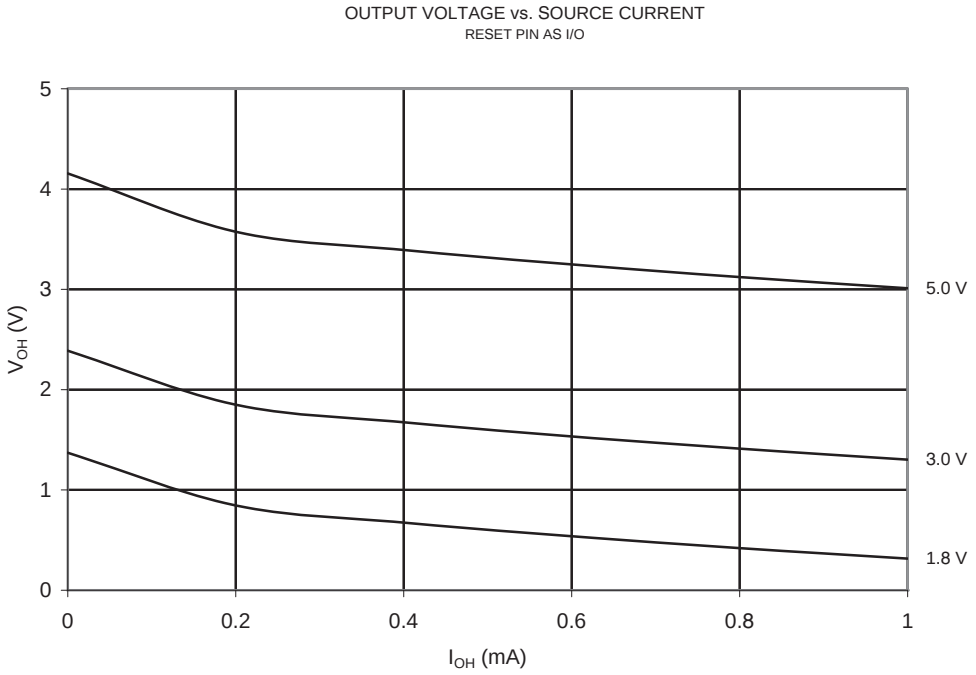


Figure 20-86. V_{OH}: Output Voltage vs. Source Current (Reset Pin as I/O, T = 25°C)



20.3.8 Input Thresholds and Hysteresis

Figure 20-87. V_{IH}: Input Threshold Voltage vs. V_{CC} (I/O Pin, Read as '1')

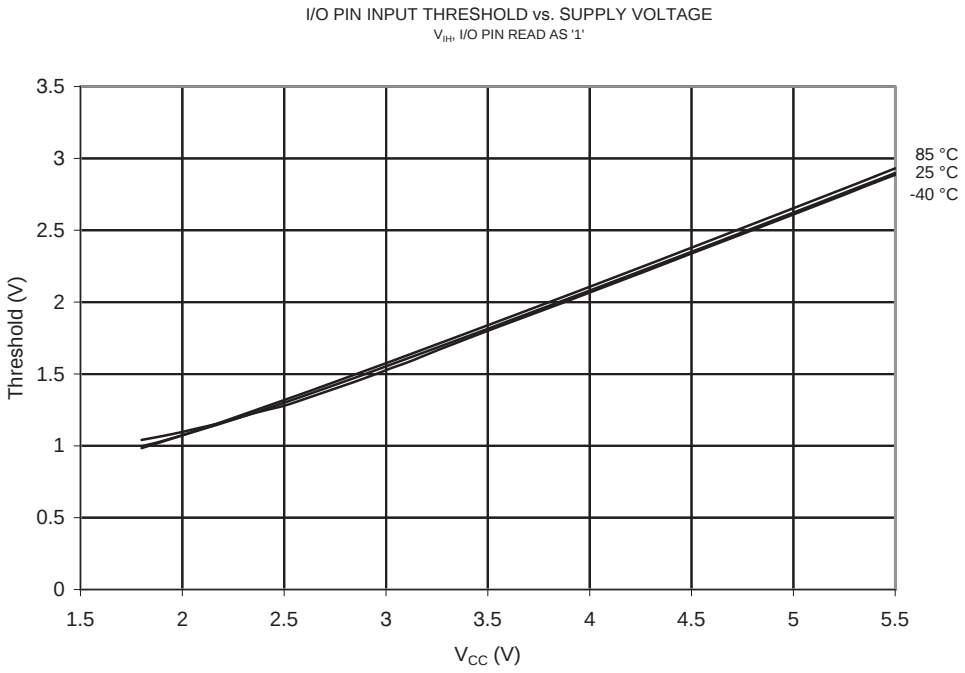


Figure 20-88. V_{IL}: Input Threshold Voltage vs. V_{CC} (I/O Pin, Read as '0')

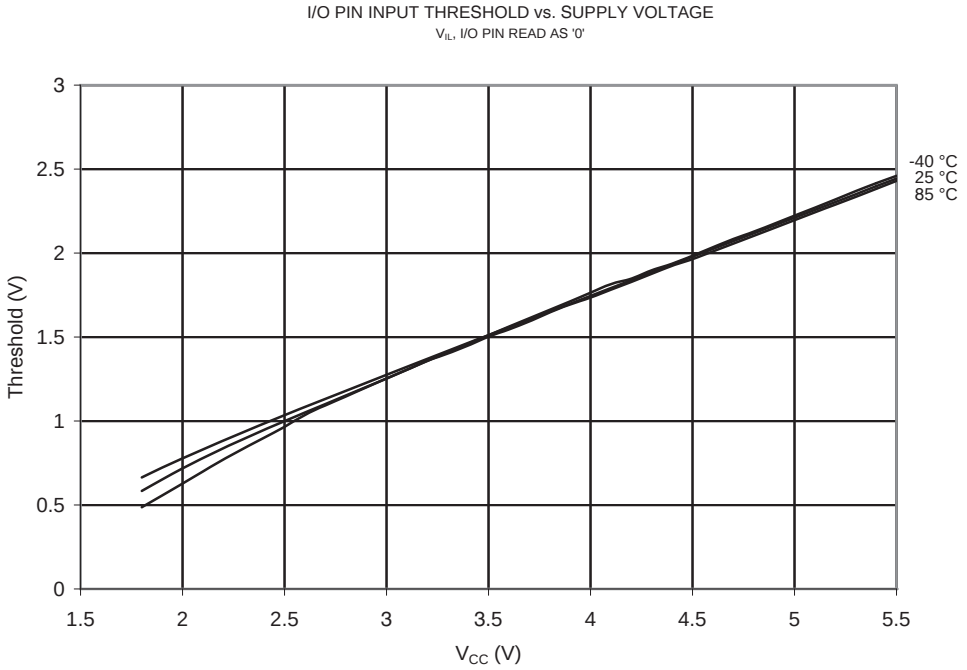


Figure 20-89. V_{IH}-V_{IL}: Input Hysteresis vs. V_{CC} (I/O Pin)

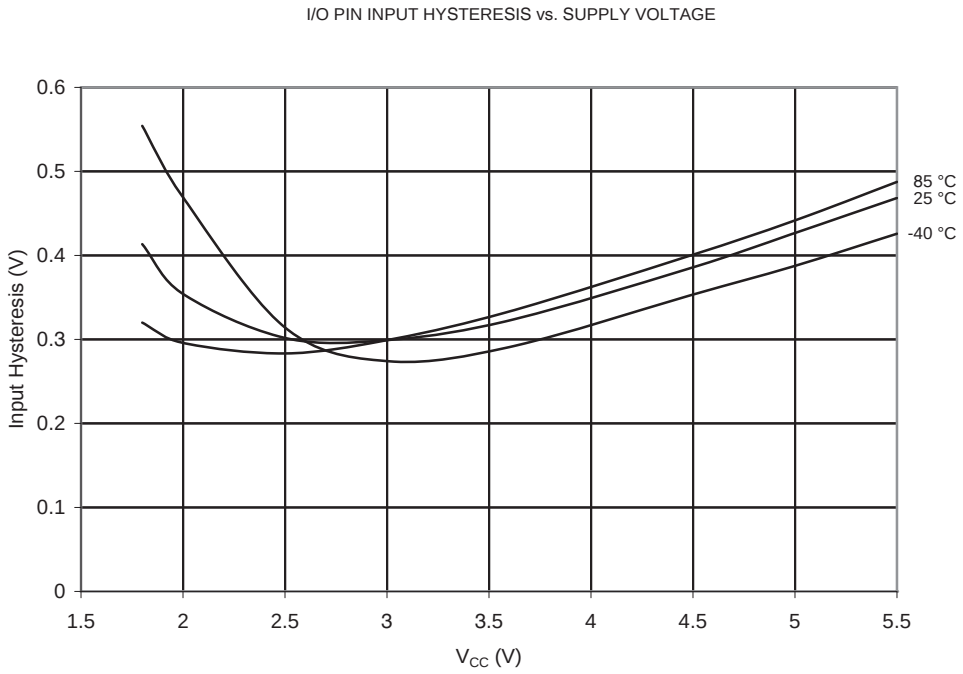
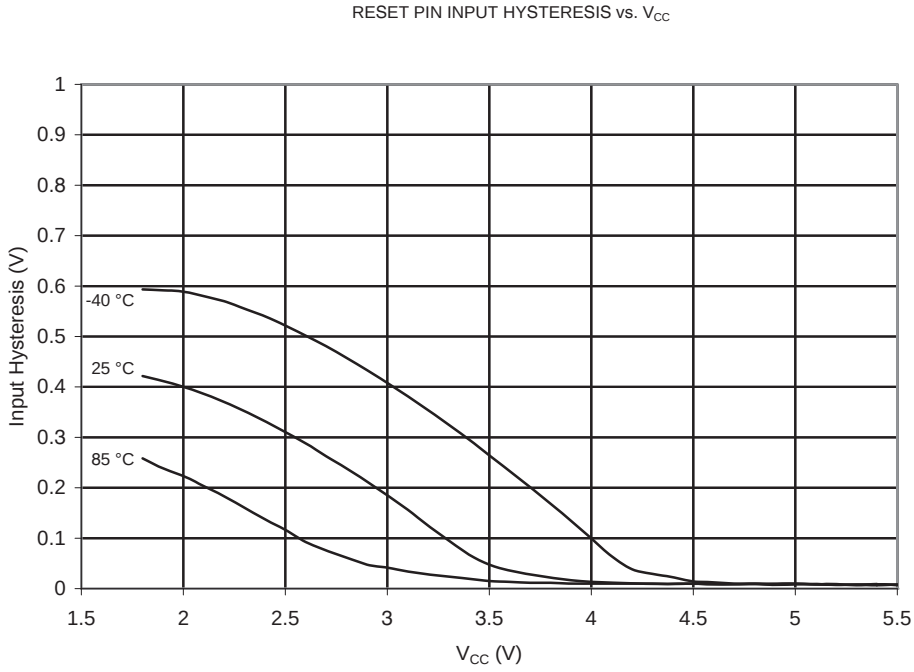


Figure 20-92. V_{IH}-V_{IL}: Input Hysteresis vs. V_{CC} (Reset Pin)



20.3.9 BOD, Bandgap and Reset

Figure 20-93. BOD Threshold vs. Temperature (BOD Level set to 4.3V)