Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	AVR
Core Size	8-Bit
Speed	16MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	23
Program Memory Size	8KB (4K x 16)
Program Memory Type	FLASH
EEPROM Size	512 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	32-TQFP
Supplier Device Package	32-TQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atmega8a-anr

3. Ordering Information

Speed (MHz)	Power Supply	Ordering Code[2]	Package[1]	Operational Range
16	2.7 - 5.5V	ATmega8A-AU ATmega8A-AUR[3] ATmega8A-PU ATmega8A-MU ATmega8A-MUR[3]	32A 32A 28P3 32M1-A 32M1-A	Industrial (-40°C to 85°C)
		ATmega8A-AN ATmega8A-ANR[3] ATmega8A-MN ATmega8A-MNR[3] ATmega8A-PN	32A 32A 32M1-A 32M1-A 28P3	Extended (-40°C to 105°C)

Note:

1. This device can also be supplied in wafer form. Please contact your local Atmel sales office for detailed ordering information and minimum quantities.
2. Pb-free packaging, complies to the European Directive for Restriction of Hazardous Substances (RoHS directive). Also Halide free and fully Green.
3. Tape and Reel

Package Type	
32A	32-lead, Thin (1.0mm) Plastic Quad Flat Package (TQFP)
28P3	28-lead, 0.300” Wide, Plastic Dual Inline Package (PDIP)
32M1-A	32-pad, 5 x 5 x 1.0mm body, lead pitch 0.50mm, Quad Flat No-Lead/Micro Lead Frame Package (QFN/MLF)

7. Resources

A comprehensive set of development tools, application notes and datasheets are available for download on http://www.atmel.com/avr.

12.6. Register Description

$\overline{SS}$: Slave Select input. When the SPI is enabled as a Slave, this pin is configured as an input regardless of the setting of DDB2. As a Slave, the SPI is activated when this pin is driven low. When the SPI is enabled as a Master, the data direction of this pin is controlled by DDB2. When the pin is forced by the SPI to be an input, the pull-up can still be controlled by the PORTB2 bit.

OC1B, Output Compare Match output: The PB2 pin can serve as an external output for the Timer/Counter1 Compare Match B. The PB2 pin has to be configured as an output (DDB2 set (one)) to serve this function. The OC1B pin is also the output pin for the PWM mode timer function.

• OC1A – Port B, Bit 1

OC1A, Output Compare Match output: The PB1 pin can serve as an external output for the Timer/Counter1 Compare Match A. The PB1 pin has to be configured as an output (DDB1 set (one)) to serve this function. The OC1A pin is also the output pin for the PWM mode timer function.

• ICP1 – Port B, Bit 0

ICP1 – Input Capture Pin: The PB0 pin can act as an Input Capture Pin for Timer/Counter1.

The tables below relate the alternate functions of Port B to the overriding signals shown in figure Figure 18-5 Alternate Port Functions(1) on page 82. SPI MSTR INPUT and SPI SLAVE OUTPUT constitute the MISO signal, while MOSI is divided into SPI MSTR OUTPUT and SPI SLAVE INPUT.

Table 18-4 Overriding Signals for Alternate Functions in PB7:PB4

Signal Name	PB7/XTAL2/ TOSC2[1][2]	PB6/XTAL1/ TOSC1[1]	PB5/SCK	PB4/MISO
PUOE	$\overline{EXT}$ • ($\overline{INTRC}$ + AS2)	$\overline{INTRC}$ + AS2	SPE • $\overline{MSTR}$	SPE • MSTR
PUO	0	0	PORTB5 • $\overline{PUD}$	PORTB4 • $\overline{PUD}$
DDOE	$\overline{EXT}$ • ($\overline{INTRC}$ + AS2)	$\overline{INTRC}$ + AS2	SPE • $\overline{MSTR}$	SPE • MSTR
DDOV	0	0	0	0
PVOE	0	0	SPE • MSTR	SPE • $\overline{MSTR}$
PVOV	0	0	SCK OUTPUT	SPI SLAVE OUTPUT
DIEOE	$\overline{EXT}$ • ($\overline{INTRC}$ + AS2)	$\overline{INTRC}$ + AS2	0	0
DIEOV	0	0	0	0
DI	–	–	SCK INPUT	SPI MSTR INPUT
AIO	Oscillator Output	Oscillator/Clock Input	–	–

Note:

1. INTRC means that the internal RC Oscillator is selected (by the CKSEL Fuse).
2. EXT means that the external RC Oscillator or an external clock is selected (by the CKSEL Fuse).

Table 18-5 Overriding Signals for Alternate Functions in PB3:PB0

Signal Name	PB3/MOSI/ OC2	PB2/$\overline{SS}$/ OC1B	PB1/OC1A	PB0/ICP1
PUOE	SPE • $\overline{MSTR}$	SPE • $\overline{MSTR}$	0	0
PUO	PORTB3 • $\overline{PUD}$	PORTB2 • $\overline{PUD}$	0	0
DDOE	SPE • $\overline{MSTR}$	SPE • $\overline{MSTR}$	0	0

Signal Name	PB3/MOSI/ OC2	PB2/$\overline{SS}$/ OC1B	PB1/OC1A	PB0/ICP1
DDOV	0	0	0	0
PVOE	SPE • MSTR + OC2 ENABLE	OC1B ENABLE	OC1A ENABLE	0
PVOV	SPI MSTR OUTPUT + OC2	OC1B	OC1A	0
DIEOE	0	0	0	0
DIEOV	0	0	0	0
DI	SPI SLAVE INPUT	SPI $\overline{SS}$	–	ICP1 INPUT
AIO	–	–	–	–

18.3.2. Alternate Functions of Port C

The Port C pins with alternate functions are shown in the table below:

Table 18-6 Port C Pins Alternate Functions

Port Pin	Alternate Function
PC6	$\overline{RESET}$ (Reset pin)
PC5	ADC5 (ADC Input Channel 5) SCL (Two-wire Serial Bus Clock Line)
PC4	ADC4 (ADC Input Channel 4) SDA (Two-wire Serial Bus Data Input/Output Line)
PC3	ADC3 (ADC Input Channel 3)
PC2	ADC2 (ADC Input Channel 2)
PC1	ADC1 (ADC Input Channel 1)
PC0	ADC0 (ADC Input Channel 0)

The alternate pin configuration is as follows:

• $\overline{RESET}$ – Port C, Bit 6

$\overline{RESET}$, Reset pin: When the RSTDISBL Fuse is programmed, this pin functions as a normal I/O pin, and the part will have to rely on Power-on Reset and Brown-out Reset as its reset sources. When the RSTDISBL Fuse is unprogrammed, the reset circuitry is connected to the pin, and the pin can not be used as an I/O pin.

If PC6 is used as a reset pin, DDC6, PORTC6 and PINC6 will all read 0.

• SCL/ADC5 – Port C, Bit 5

SCL, Two-wire Serial Interface Clock: When the TWEN bit in TWCR is set (one) to enable the Two-wire Serial Interface, pin PC5 is disconnected from the port and becomes the Serial Clock I/O pin for the Two-wire Serial Interface. In this mode, there is a spike filter on the pin to suppress spikes shorter than 50 ns on the input signal, and the pin is driven by an open drain driver with slew-rate limitation.

PC5 can also be used as ADC input Channel 5. Note that ADC input channel 5 uses digital power.

• SDA/ADC4 – Port C, Bit 4

```
    ; Read TCNT1 into r17:r16
    in    r16,TCNT1L
    in    r17,TCNT1H
    ; Restore global interrupt flag
    out   SREG,r18
    ret
```

C Code Example[1]

```
unsigned int TIM16_ReadTCNT1( void )
{
   unsigned char sreg;
   unsigned int i;
   /* Save global interrupt flag */
   sreg = SREG;
   /* Disable interrupts */
   _CLI();
   /* Read TCNT1 into i */
   i = TCNT1;
   /* Restore global interrupt flag */
   SREG = sreg;
   return i;
}
```

Note: 1. See *About Code Examples*.

The assembly code example returns the TCNT1 value in the r17:r16 Register pair.

The following code examples show how to do an atomic write of the TCNT1 Register contents. Writing any of the OCR1A/B or ICR1 Registers can be done by using the same principle.

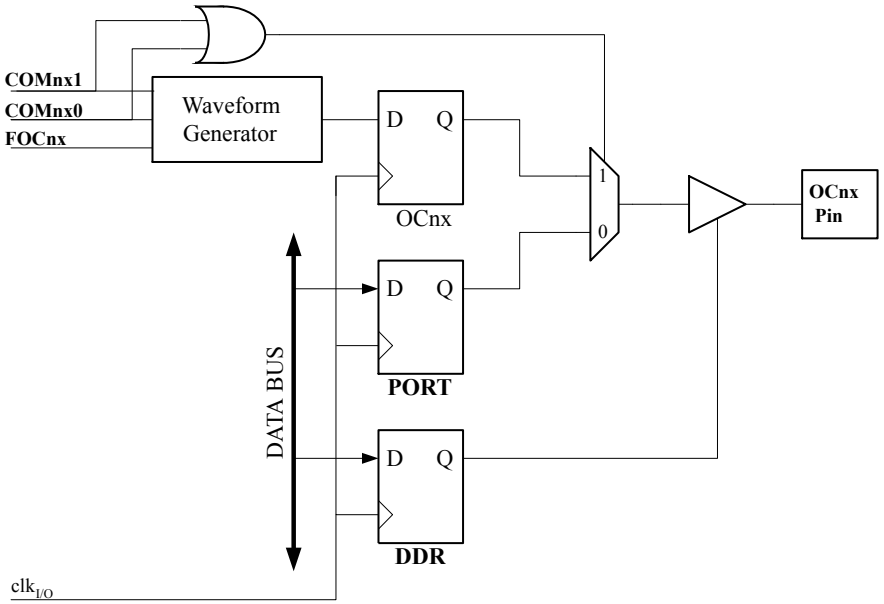
Assembly Code Example[1]

```
TIM16_WriteTCNT1:
   ; Save global interrupt flag
   in     r18,SREG
   ; Disable interrupts
   cli
   ; Set TCNT1 to r17:r16
   out    TCNT1H,r17
   out    TCNT1L,r16
   ; Restore global interrupt flag
   out    SREG,r18
   ret
```

C Code Example[1]

```
void TIM16_WriteTCNT1( unsigned int i )
{
   unsigned char sreg;
   unsigned int i;
   /* Save global interrupt flag */
   sreg = SREG;
   /* Disable interrupts */
   _CLI();
   /* Set TCNT1 to i */
   TCNT1 = i;
   /* Restore global interrupt flag */
```

Figure 21-5 Compare Match Output Unit, Schematic



The general I/O port function is overridden by the Output Compare (OC1x) from the waveform generator if either of the COM1x1:0 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 pin (DDR_OC1x) must be set as output before the OC1x value is visible on the pin. The port override function is generally independent of the Waveform Generation mode, but there are some exceptions. Refer to Table 21-2 Compare Output Mode, non-PWM on page 132, Table 21-3 Compare Output Mode, Fast PWM(1) on page 133 and Table 21-4 Compare Output Mode, Phase Correct and Phase and Frequency Correct PWM(1) on page 133 for details.

The design of the Output Compare Pin logic allows initialization of the OC1x state before the output is enabled. Note that some COM1x1:0 bit settings are reserved for certain modes of operation. See Register Description.

The COM1x1:0 bits have no effect on the Input Capture unit.

21.8.1. Compare Output Mode and Waveform Generation

The waveform generator uses the COM1x1:0 bits differently in normal, CTC, and PWM modes. For all modes, setting the COM1x1:0 = 0 tells the waveform generator that no action on the OC1x Register is to be performed on the next Compare Match. For compare output actions in the non-PWM modes refer to Table 21-2 Compare Output Mode, non-PWM on page 132. For fast PWM mode refer to Table 21-3 Compare Output Mode, Fast PWM(1) on page 133, and for phase correct and phase and frequency correct PWM refer to Table 21-4 Compare Output Mode, Phase Correct and Phase and Frequency Correct PWM(1) on page 133.

A change of the COM1x1:0 bits state will have effect at the first Compare Match after the bits are written. For nonPWM modes, the action can be forced to have immediate effect by using the FOC1x strobe bits.

21.9. 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 the *Waveform Generation mode* (WGM13:0) and *Compare Output mode* (COM1x1:0) bits. The Compare Output mode bits do not affect the counting sequence, while the Waveform Generation mode bits do. The COM1x1:0 bits control whether the PWM output generated should be inverted or not (inverted or non-inverted PWM). For non-PWM modes the COM1x1:0 bits

or Standby mode due to unstable clock signal upon start-up, no matter whether the Oscillator is in use or a clock signal is applied to the TOSC1 pin.

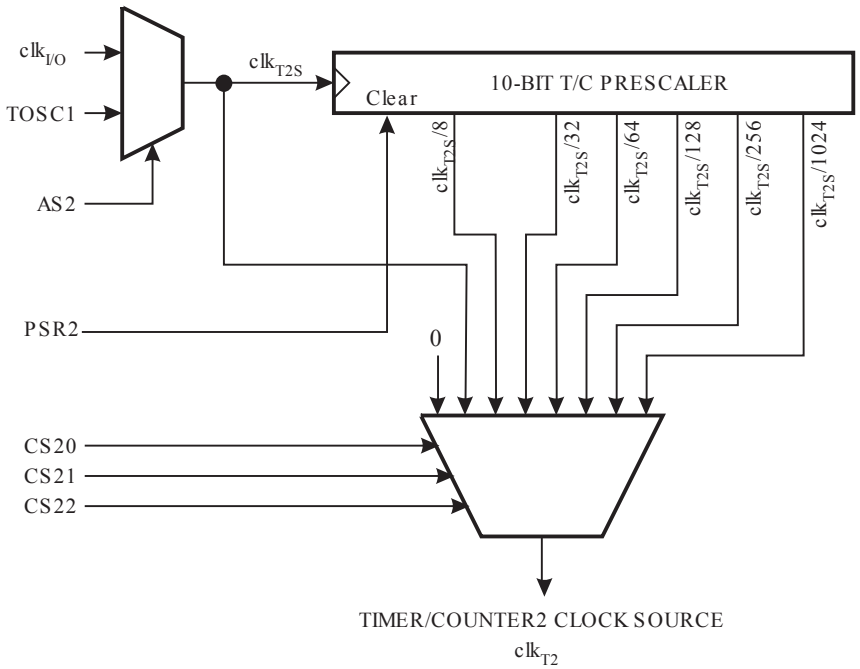
- Description of wake up from Power-save or Extended Standby mode when the timer is clocked asynchronously: When the interrupt condition is met, the wake up process is started on the following cycle of the timer clock, that is, the timer is always advanced by at least one before the processor can read the counter value. After wake-up, the MCU is halted for four cycles, it executes the interrupt routine, and resumes execution from the instruction following SLEEP.
- Reading of the TCNT2 Register shortly after wake-up from Power-save may give an incorrect result. Since TCNT2 is clocked on the asynchronous TOSC clock, reading TCNT2 must be done through a register synchronized to the internal I/O clock domain. Synchronization takes place for every rising TOSC1 edge. When waking up from Power-save mode, and the I/O clock ($clk_{I/O}$) again becomes active, TCNT2 will read as the previous value (before entering sleep) until the next rising TOSC1 edge. The phase of the TOSC clock after waking up from Power-save mode is essentially unpredictable, as it depends on the wake-up time. The recommended procedure for reading TCNT2 is thus as follows:

1. Write any value to either of the registers OCR2 or TCCR2.
2. Wait for the corresponding Update Busy Flag to be cleared.
3. Read TCNT2.

- During asynchronous operation, the synchronization of the Interrupt Flags for the asynchronous timer takes three processor cycles plus one timer cycle. The timer is therefore advanced by at least one before the processor can read the timer value causing the setting of the Interrupt Flag. The Output Compare Pin is changed on the timer clock and is not synchronized to the processor clock.

22.10. Timer/Counter Prescaler

Figure 22-12 Prescaler for Timer/Counter2



The clock source for Timer/Counter2 is named clk_{T2S}. clk_{T2S} is by default connected to the main system clock $clk_{I/O}$. By setting the AS2 bit in ASSR, Timer/Counter2 is asynchronously clocked from the TOSC1 pin. This enables use of Timer/Counter2 as a Real Time Counter (RTC). When AS2 is set, pins TOSC1

22.11.7. SFIOR – Special Function IO Register

When using the I/O specific commands IN and OUT, the I/O addresses 0x00 - 0x3F must be used. When addressing I/O Registers as data space using LD and ST instructions, 0x20 must be added to these offset addresses. The device is a complex microcontroller with more peripheral units than can be supported within the 64 location reserved in Opcode for the IN and OUT instructions. For the Extended I/O space from 0x60 in SRAM, only the ST/STS/STD and LD/LDS/LDD instructions can be used.

Name: SFIOR
Offset: 0x30
Reset: 0
Property: When addressing I/O Registers as data space the offset address is 0x50

Bit	7	6	5	4	3	2	1	0
							PSR2	
Access							R/W	
Reset							0	

Bit 1 – PSR2: Prescaler Reset Timer/Counter2

When this bit is written to one, the Timer/Counter2 prescaler will be reset. The bit will be cleared by hardware after the operation is performed. Writing a zero to this bit will have no effect. This bit will always be read as zero if Timer/Counter2 is clocked by the internal CPU clock. If this bit is written when Timer/Counter2 is operating in Asynchronous mode, the bit will remain one until the prescaler has been reset.

23.5.3. SPDR – SPI Data Register is a read/write register

When using the I/O specific commands IN and OUT, the I/O addresses 0x00 - 0x3F must be used. When addressing I/O Registers as data space using LD and ST instructions, 0x20 must be added to these offset addresses. The device is a complex microcontroller with more peripheral units than can be supported within the 64 location reserved in Opcode for the IN and OUT instructions. For the Extended I/O space from 0x60 in SRAM, only the ST/STS/STD and LD/LDS/LDD instructions can be used.

Name: SPDR
Offset: 0x0F
Reset: 0xXX
Property: When addressing I/O Registers as data space the offset address is 0x2F

Bit	7	6	5	4	3	2	1	0
	SPID7	SPID6	SPID5	SPID4	SPID3	SPID2	SPID1	SPID0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	x	x	x	x	x	x	x	x

Bits 7:0 – SPIDn: SPI Data

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

- SPID7 is MSB
- SPID0 is LSB

24.11.3. UCSRB – USART Control and Status Register B

When using the I/O specific commands IN and OUT, the I/O addresses 0x00 - 0x3F must be used. When addressing I/O Registers as data space using LD and ST instructions, 0x20 must be added to these offset addresses. The device is a complex microcontroller with more peripheral units than can be supported within the 64 location reserved in Opcode for the IN and OUT instructions. For the Extended I/O space from 0x60 in SRAM, only the ST/STS/STD and LD/LDS/LDD instructions can be used.

Name: UCSRB
Offset: 0x0A
Reset: 0x00
Property: When addressing I/O Registers as data space the offset address is 0x2A

Bit	7	6	5	4	3	2	1	0
	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8
Access	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W
Reset	0	0	0	0	0	0	0	0

Bit 7 – RXCIE: RX Complete Interrupt Enable
Writing this bit to one enables interrupt on the RXC Flag. A USART Receive Complete interrupt will be generated only if the RXCIE bit is written to one, the Global Interrupt Flag in SREG is written to one and the RXC bit in UCSRA is set.

Bit 6 – TXCIE: TX Complete Interrupt Enable
Writing this bit to one enables interrupt on the TXC Flag. A USART Transmit Complete interrupt will be generated only if the TXCIE bit is written to one, the Global Interrupt Flag in SREG is written to one and the TXC bit in UCSRA is set.

Bit 5 – UDRIE: USART Data Register Empty Interrupt Enable
Writing this bit to one enables interrupt on the UDRE Flag. A Data Register Empty interrupt will be generated only if the UDRIE bit is written to one, the Global Interrupt Flag in SREG is written to one and the UDRE bit in UCSRA is set.

Bit 4 – RXEN: Receiver Enable
Writing this bit to one enables the USART Receiver. The Receiver will override normal port operation for the RxD pin when enabled. Disabling the Receiver will flush the receive buffer invalidating the FE, DOR and PE Flags.

Bit 3 – TXEN: Transmitter Enable
Writing this bit to one enables the USART Transmitter. The Transmitter will override normal port operation for the TxD pin when enabled. The disabling of the Transmitter (writing TXEN to zero) will not become effective until ongoing and pending transmissions are completed (i.e., when the Transmit Shift Register and Transmit Buffer Register do not contain data to be transmitted). When disabled, the Transmitter will no longer override the TxD port.

Bit 2 – UCSZ2: Character Size
The UCSZ2 bits combined with the UCSZ1:0 bit in UCSRC sets the number of data bits (Character Size) in a frame the Receiver and Transmitter use.

25.8.1. TWBR – TWI Bit Rate Register

When using the I/O specific commands IN and OUT, the I/O addresses 0x00 - 0x3F must be used. When addressing I/O Registers as data space using LD and ST instructions, 0x20 must be added to these offset addresses. The device is a complex microcontroller with more peripheral units than can be supported within the 64 location reserved in Opcode for the IN and OUT instructions. For the Extended I/O space from 0x60 in SRAM, only the ST/STS/STD and LD/LDS/LDD instructions can be used.

Name: TWBR
Offset: 0x00
Reset: 0x00
Property: When addressing I/O Registers as data space the offset address is 0x20

Bit	7	6	5	4	3	2	1	0
	TWBR7	TWBR6	TWBR5	TWBR4	TWBR3	TWBR2	TWBR1	TWBR0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 7:0 – TWBRn: TWI Bit Rate Register [n = 7:0]

TWBR selects the division factor for the bit rate generator. The bit rate generator is a frequency divider which generates the SCL clock frequency in the Master modes. See Bit Rate Generator Unit for calculating bit rates.

26.3.2. ACSR – Analog Comparator Control and Status Register

When using the I/O specific commands IN and OUT, the I/O addresses 0x00 - 0x3F must be used. When addressing I/O Registers as data space using LD and ST instructions, 0x20 must be added to these offset addresses. The device is a complex microcontroller with more peripheral units than can be supported within the 64 location reserved in Opcode for the IN and OUT instructions. For the Extended I/O space from 0x60 in SRAM, only the ST/STS/STD and LD/LDS/LDD instructions can be used.

Name: ACSR
Offset: 0x08
Reset: N/A
Property: When addressing I/O Registers as data space the offset address is 0x28

Bit	7	6	5	4	3	2	1	0
	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0
Access	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	a	0	0	0	0	0

Bit 7 – ACD: Analog Comparator Disable
When this bit is written logic one, the power to the Analog Comparator is switched off. This bit can be set at any time to turn off the Analog Comparator. This will reduce power consumption in Active and Idle mode. 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 – ACBG: Analog Comparator Bandgap Select
When this bit is set, a fixed bandgap reference voltage replaces the positive input to the Analog Comparator. When this bit is cleared, AIN0 is applied to the positive input of the Analog Comparator. Refer to Internal Voltage Reference in System Control and Reset.

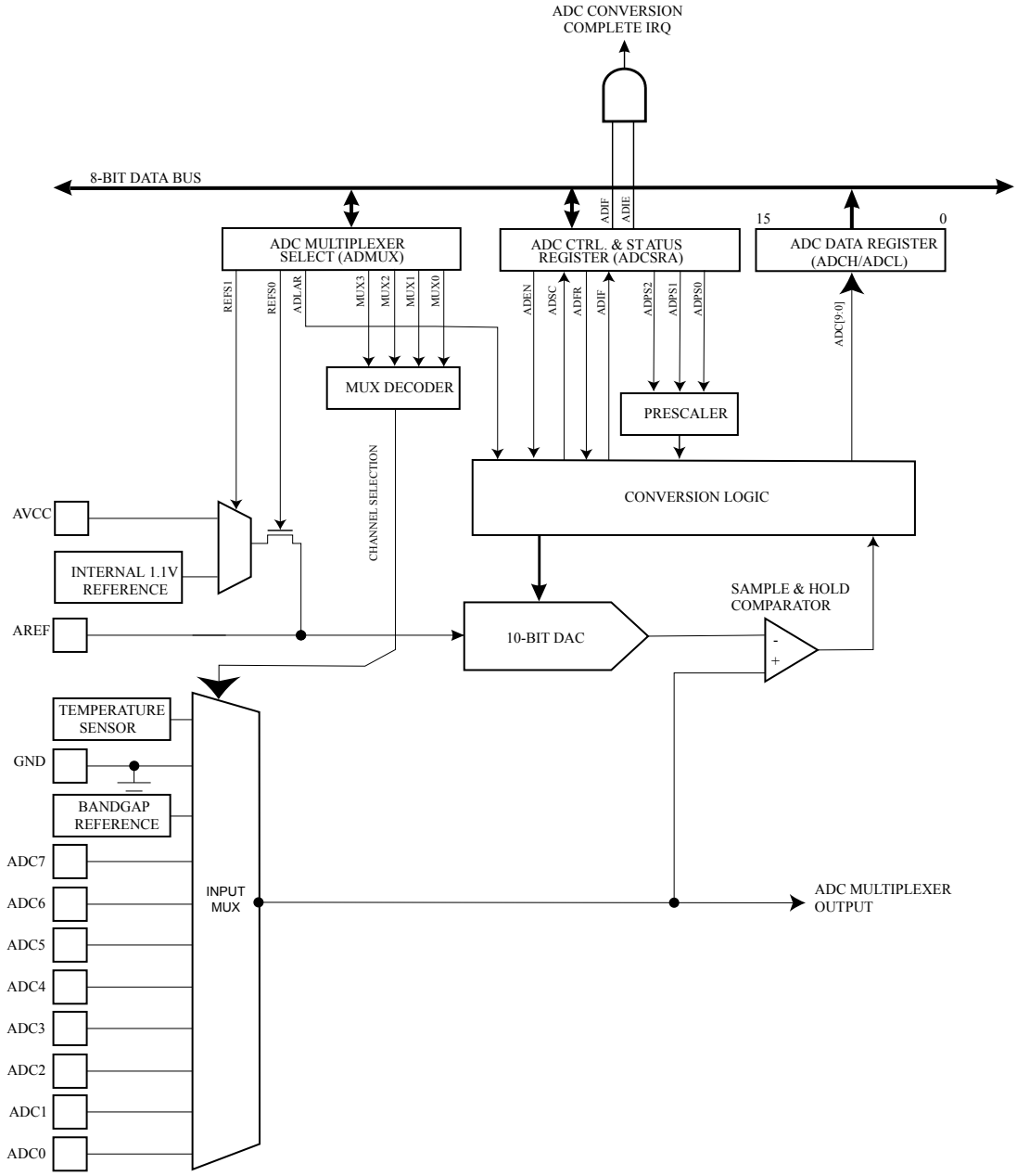
Bit 5 – ACO: Analog Comparator Output
The output of the Analog Comparator is synchronized and then directly connected to ACO. The synchronization introduces a delay of 1 - 2 clock cycles.

Bit 4 – ACI: Analog Comparator Interrupt Flag
This bit is set by hardware when a comparator output event triggers the interrupt mode defined by ACIS1 and ACIS0. The Analog Comparator interrupt routine is executed if the ACIE bit is set and the I-bit in SREG is set. ACI is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, ACI is cleared by writing a logic one to the flag.

Bit 3 – ACIE: Analog Comparator Interrupt Enable
When the ACIE bit is written logic one and the I-bit in the Status Register is set, the Analog Comparator interrupt is activated. When written logic zero, the interrupt is disabled.

Bit 2 – ACIC: Analog Comparator Input Capture Enable
When written logic one, this bit enables the input capture function in Timer/Counter1 to be triggered by the Analog Comparator. The comparator output is in this case directly connected to the input capture front-end logic, making the comparator utilize the noise canceler and edge select features of the Timer/Counter1 Input Capture interrupt. When written logic zero, no connection between the Analog Comparator and the input capture function exists. To make the comparator trigger the Timer/Counter1 Input Capture interrupt, the ICIE1 bit in the Timer Interrupt Mask Register (TIMSK1) must be set.

Figure 27-1 Analog to Digital Converter Block Schematic Operation



The ADC converts an analog input voltage to a 10-bit digital value through successive approximation. The minimum value represents GND and the maximum value represents the voltage on the AREF pin minus 1LSB. Optionally, AV_{CC} or an internal 2.56V reference voltage may be connected to the AREF pin by writing to the REFSn bits in the ADMUX Register. The internal voltage reference may thus be decoupled by an external capacitor at the AREF pin to improve noise immunity.

The analog input channel is selected by writing to the MUX bits in ADMUX. Any of the ADC input pins, as well as GND and a fixed bandgap voltage reference, can be selected as single ended inputs to the ADC. The ADC is enabled by setting the ADC Enable bit, ADEN in ADCSRA. Voltage reference and input channel selections will not go into effect until ADEN is set. The ADC does not consume power when ADEN is cleared, so it is recommended to switch off the ADC before entering power saving sleep modes.

29. Memory Programming

29.1. Program and Data Memory Lock Bits

The ATmega8A provides six Lock bits. These can be left unprogrammed ('1') or can be programmed ('0') to obtain the additional features listed in the *Lock Bit Protection Modes* table below. The Lock Bits can only be erased to “1” with the Chip Erase command.

Table 29-1 Lock Bit Byte

Lock Bit Byte	Bit No.	Description	Default Value[1]
	7	–	1 (unprogrammed)
	6	–	1 (unprogrammed)
BLB12	5	Boot Lock bit	1 (unprogrammed)
BLB11	4	Boot Lock bit	1 (unprogrammed)
BLB02	3	Boot Lock bit	1 (unprogrammed)
BLB01	2	Boot Lock bit	1 (unprogrammed)
LB2	1	Lock bit	1 (unprogrammed)
LB1	0	Lock bit	1 (unprogrammed)

Note: 1. “1” means unprogrammed, “0” means programmed.

Table 29-2 Lock Bit Protection Modes[2]

Memory Lock Bits			Protection Type
LB Mode	LB2	LB1	
1	1	1	No memory lock features enabled.
2	1	0	Further programming of the Flash and EEPROM is disabled in Parallel and Serial Programming mode. The Fuse bits are locked in both Serial and Parallel Programming mode.[1]
3	0	0	Further programming and verification of the Flash and EEPROM is disabled in parallel and Serial Programming mode. The Fuse Bits are locked in both Serial and Parallel Programming modes.[1]
BLB0 Mode	BLB02	BLB01	
1	1	1	No restrictions for SPM or Load Program Memory (LPM) instruction accessing the Application section.
2	1	0	SPM is not allowed to write to the Application section.
3	0	0	SPM is not allowed to write to the Application section, and LPM executing from the Boot Loader section is not allowed to read from the Application section. If Interrupt Vectors are placed in the Boot Loader section, interrupts are disabled while executing from the Application section.

Step A. Load Command “Write Flash”.

1. Set XA1, XA0 to “10”. This enables command loading.
2. Set BS1 to “0”.
3. Set DATA to “0001 0000”. This is the command for Write Flash.
4. Give XTAL1 a positive pulse. This loads the command.

Step B. Load Address Low Byte.

1. Set XA1, XA0 to “00”. This enables address loading.
2. Set BS1 to “0”. This selects low address.
3. Set DATA = Address low byte (0x00 - 0xFF).
4. Give XTAL1 a positive pulse. This loads the address low byte.

Step C. Load Data Low Byte.

1. Set XA1, XA0 to “01”. This enables data loading.
2. Set DATA = Data low byte (0x00 - 0xFF).
3. Give XTAL1 a positive pulse. This loads the data byte.

Step D. Load Data High Byte.

1. Set BS1 to “1”. This selects high data byte.
2. Set XA1, XA0 to “01”. This enables data loading.
3. Set DATA = Data high byte (0x00 - 0xFF).
4. Give XTAL1 a positive pulse. This loads the data byte.

Step E. Latch Data.

1. Set BS1 to “1”. This selects high data byte.
2. Give PAGEL a positive pulse. This latches the data bytes. (Refer to figure Programming the Flash Waveforms in this section for signal waveforms)

Step F. Repeat B through E until the entire buffer is filled or until all data within the page is loaded.

While the lower bits in the address are mapped to words within the page, the higher bits address the pages within the FLASH. This is illustrated in the following figure, Addressing the Flash Which is Organized in Pages, in this section. Note that if less than eight bits are required to address words in the page (pagesize < 256), the most significant bit(s) in the address low byte are used to address the page when performing a Page Write.

Step G. Load Address High byte.

1. Set XA1, XA0 to “00”. This enables address loading.
2. Set BS1 to “1”. This selects high address.
3. Set DATA = Address high byte (0x00 - 0xFF).
4. Give XTAL1 a positive pulse. This loads the address high byte.

Step H. Program Page.

1. Set BS1 = “0”
2. Give $\overline{WR}$ a negative pulse. This starts programming of the entire page of data. RDY/$\overline{BSY}$ goes low.
3. Wait until RDY/$\overline{BSY}$ goes high (Refer to figure Programming the Flash Waveforms in this section).

29.7.13. Reading the Calibration Byte

The algorithm for reading the Calibration byte is as follows (Please refer to Programming the Flash on page 289 for details on Command and Address loading):

1. Step A: Load Command “0000 1000”.
2. Step B: Load Address Low byte, (0x00 - 0x03).
3. Set $\overline{OE}$ to “0”, and BS1 to “1”. The Calibration byte can now be read at DATA.
4. Set $\overline{OE}$ to “1”.

29.7.14. Parallel Programming Characteristics

Figure 29-6 Parallel Programming Timing, Including some General Timing Requirements

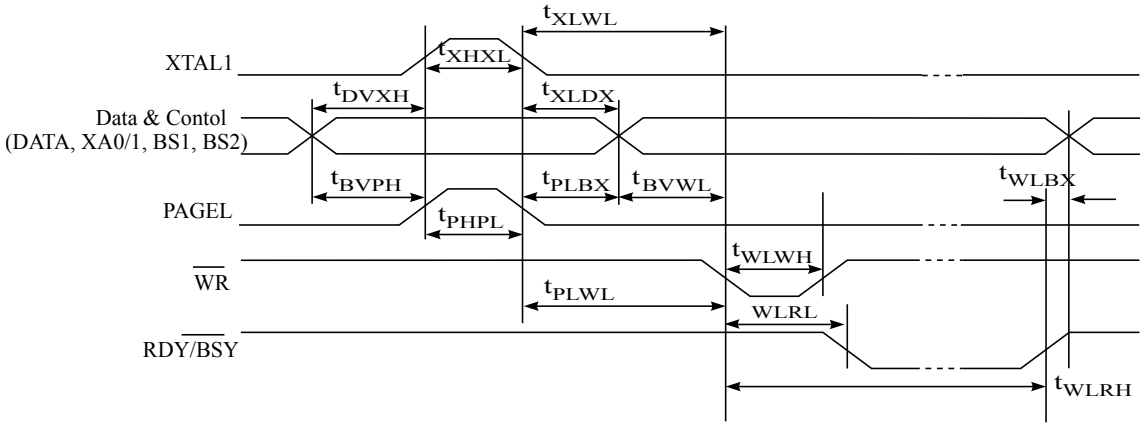
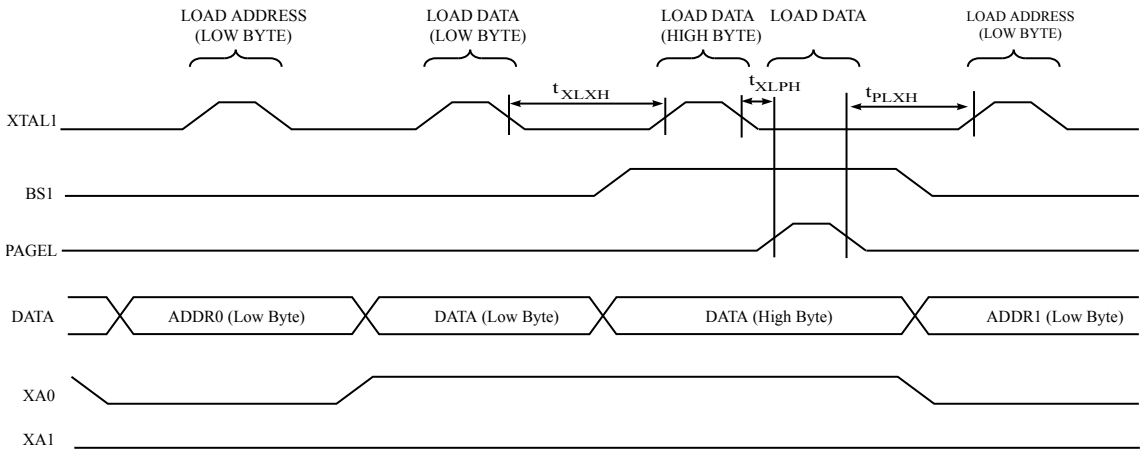
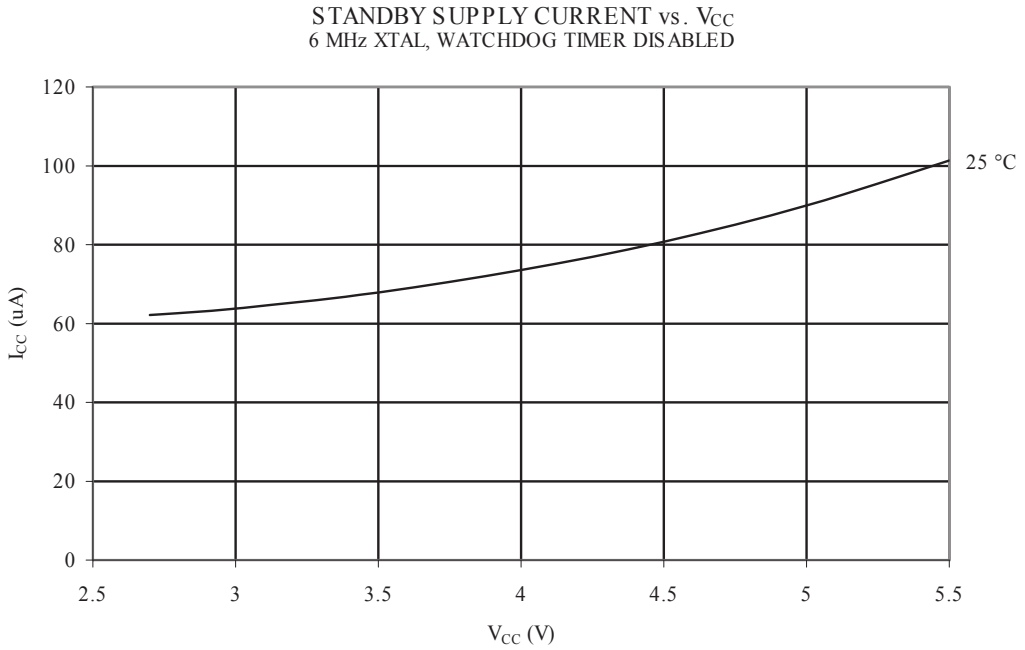


Figure 29-7 Parallel Programming Timing, Loading Sequence with Timing Requirements(1)



Note: 1. The timing requirements shown in the first figure in this section (i.e., t_{DVXH}, t_{XHXL}, and t_{XLDX}) also apply to loading operation.

Figure 32-23 Standby Supply Current vs. V_{CC} (6MHz Xtal, Watchdog Timer Disabled)



32.6. Pin Pull-up

Figure 32-24 I/O Pin Pull-up Resistor Current vs. Input Voltage (V_{CC} = 5V)

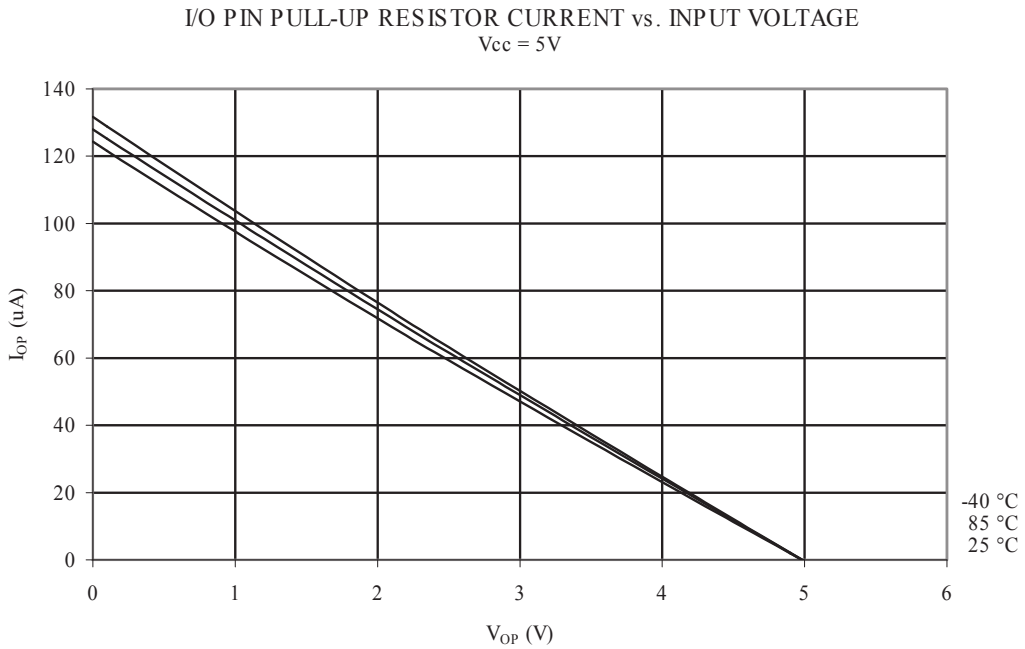
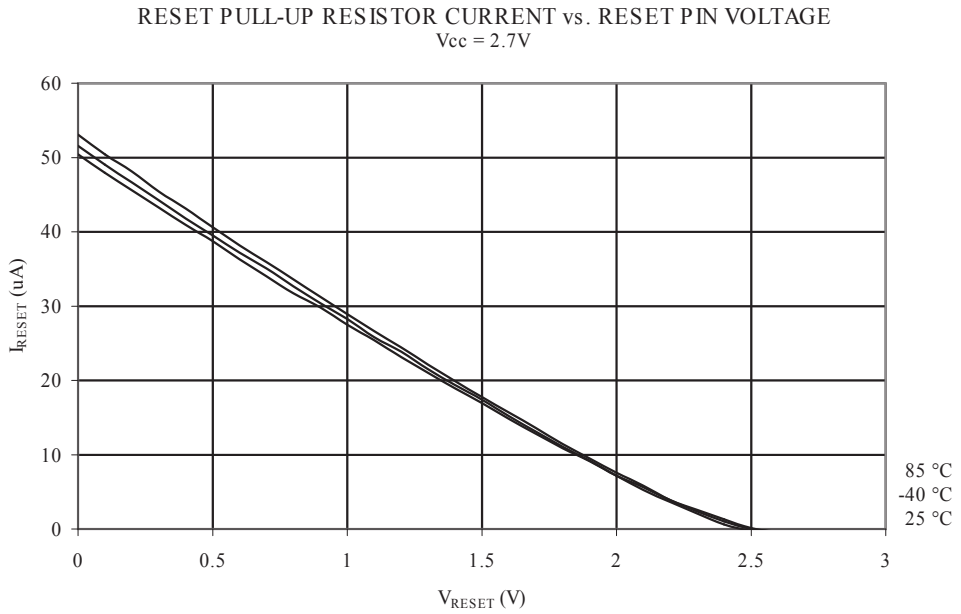


Figure 32-27 Reset Pull-up Resistor Current vs. Reset Pin Voltage (V_{CC} = 2.7V)



32.7. Pin Driver Strength

Figure 32-28 I/O Pin Output Voltage vs. Source Current (V_{CC} = 5.0V)

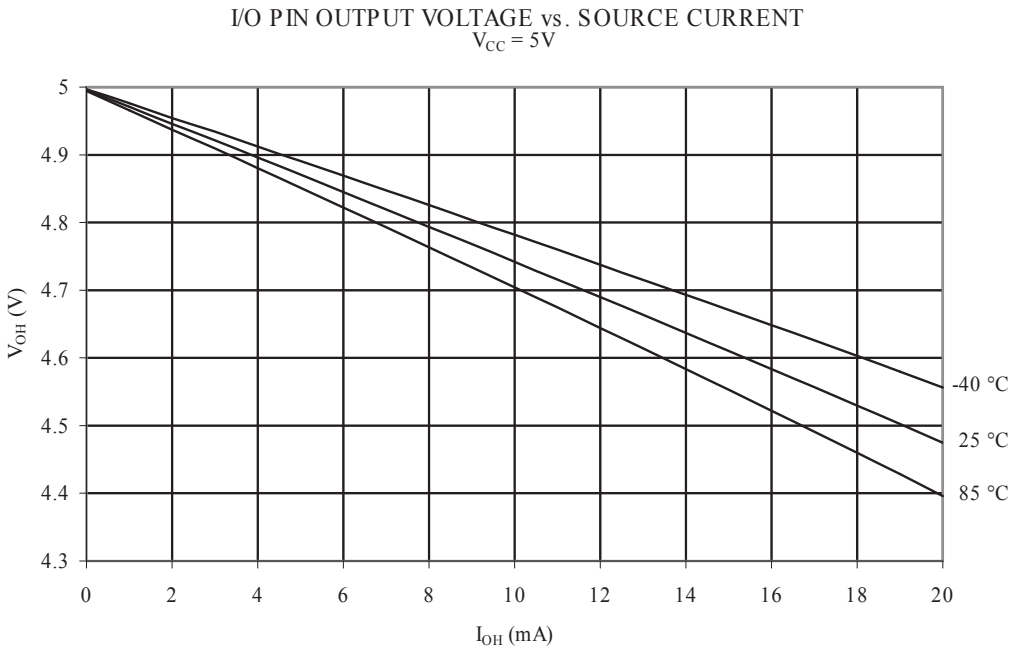


Figure 33-19 I/O Pin Output Voltage vs. Source Current (V_{CC} = 3V)

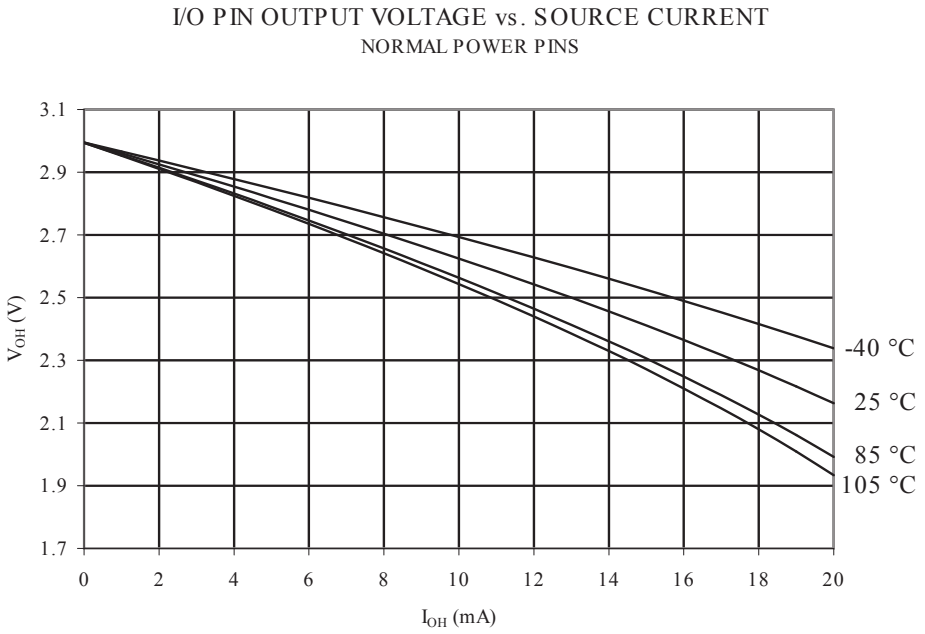


Figure 33-20 I/O Pin Output Voltage vs. Sink Current (V_{CC} = 5V)