8-bit AVR Microcontrollers

ATmega328/P

DATASHEET COMPLETE

Introduction

The Atmel® picoPower® ATmega328/P is a low-power CMOS 8-bit microcontroller based on the AVR® enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega328/P achieves throughputs close to 1MIPS per MHz. This empowers system designer to optimize the device for power consumption versus processing speed.

Feature

High Performance, Low Power Atmel®AVR® 8-Bit Microcontroller Family

- Advanced RISC Architecture
  - 131 Powerful Instructions
  - Most Single Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 20 MIPS Throughput at 20MHz
  - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory Segments
  - 32KBytes of In-System Self-Programmable Flash program Memory
  - 1KBytes EEPROM
  - 2KBytes Internal SRAM
  - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
  - Data Retention: 20 years at 85°C/100 years at 25°C(1)
  - Optional Boot Code Section with Independent Lock Bits
    - In-System Programming by On-chip Boot Program
    - True Read-While-Write Operation
  - Programming Lock for Software Security
- Atmel® QTouch® Library Support
  - Capacitive Touch Buttons, Sliders and Wheels
  - QTouch and QMatrix® Acquisition
  - Up to 64 sense channels

- Peripheral Features
  - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
  - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
  - Real Time Counter with Separate Oscillator
  - Six PWM Channels
  - 8-channel 10-bit ADC in TQFP and QFN/MLF package
    - Temperature Measurement
  - 6-channel 10-bit ADC in PDIP Package
    - Temperature Measurement
  - Two Master/Slave SPI Serial Interface
  - One Programmable Serial USART
  - One Byte-oriented 2-wire Serial Interface (Philips I^2C compatible)
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - One On-chip Analog Comparator
  - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated Oscillator
  - External and Internal Interrupt Sources
  - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
  - 23 Programmable I/O Lines
  - 28-pin PDIP, 32-lead TQFP, 28-pad QFN/MLF and 32-pad QFN/MLF
- Operating Voltage:
  - 1.8 - 5.5V
- Temperature Range:
  - -40°C to 105°C
- Speed Grade:
  - 0 - 4MHz @ 1.8 - 5.5V
  - 0 - 10MHz @ 2.7 - 5.5V
  - 0 - 20MHz @ 4.5 - 5.5V
- Power Consumption at 1MHz, 1.8V, 25°C
  - Active Mode: 0.2mA
  - Power-down Mode: 0.1µA
  - Power-save Mode: 0.75µA (Including 32kHz RTC)

5. Pin Configurations

5.1. Pin-out

Figure 5-1. 28-pin PDIP

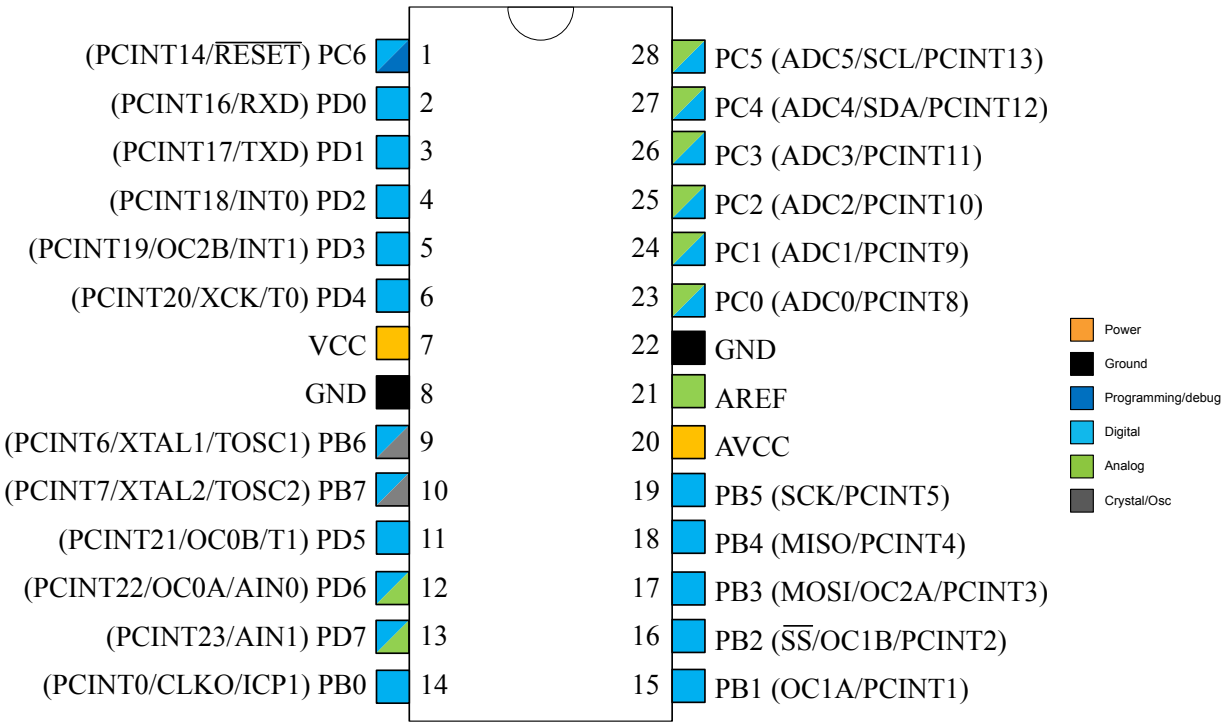


Figure 5-2. 28-pin MLF Top View

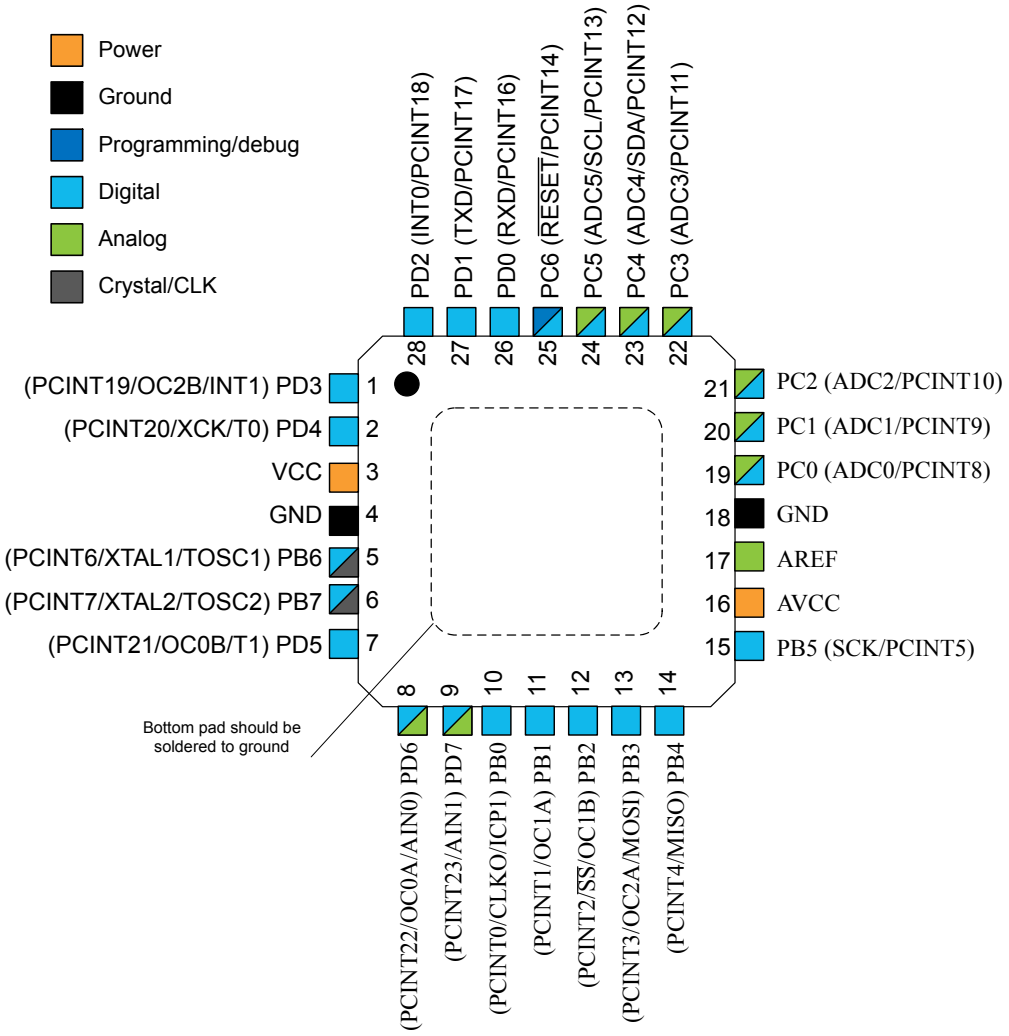


Figure 5-3. 32-pin TQFP Top View

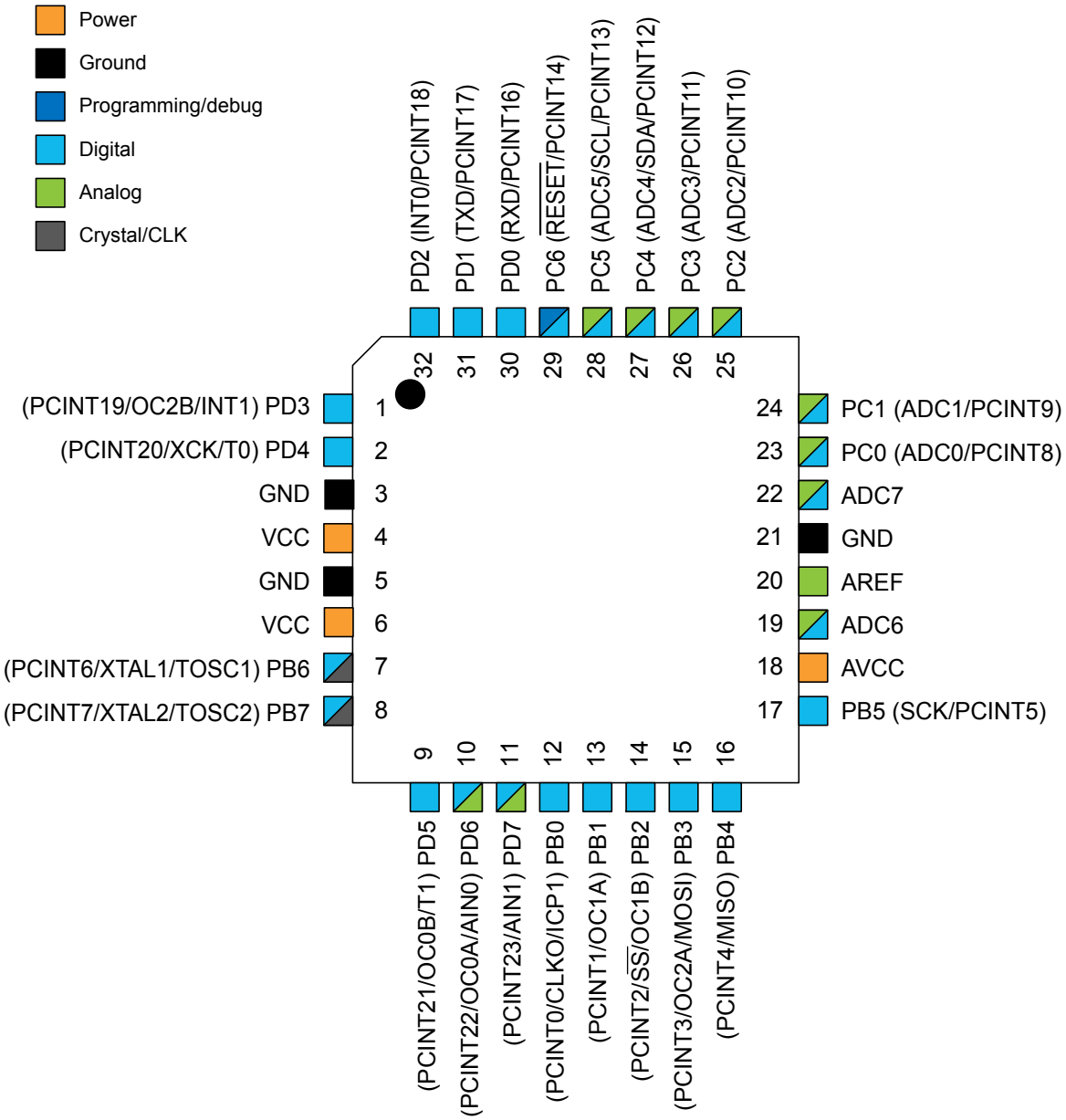
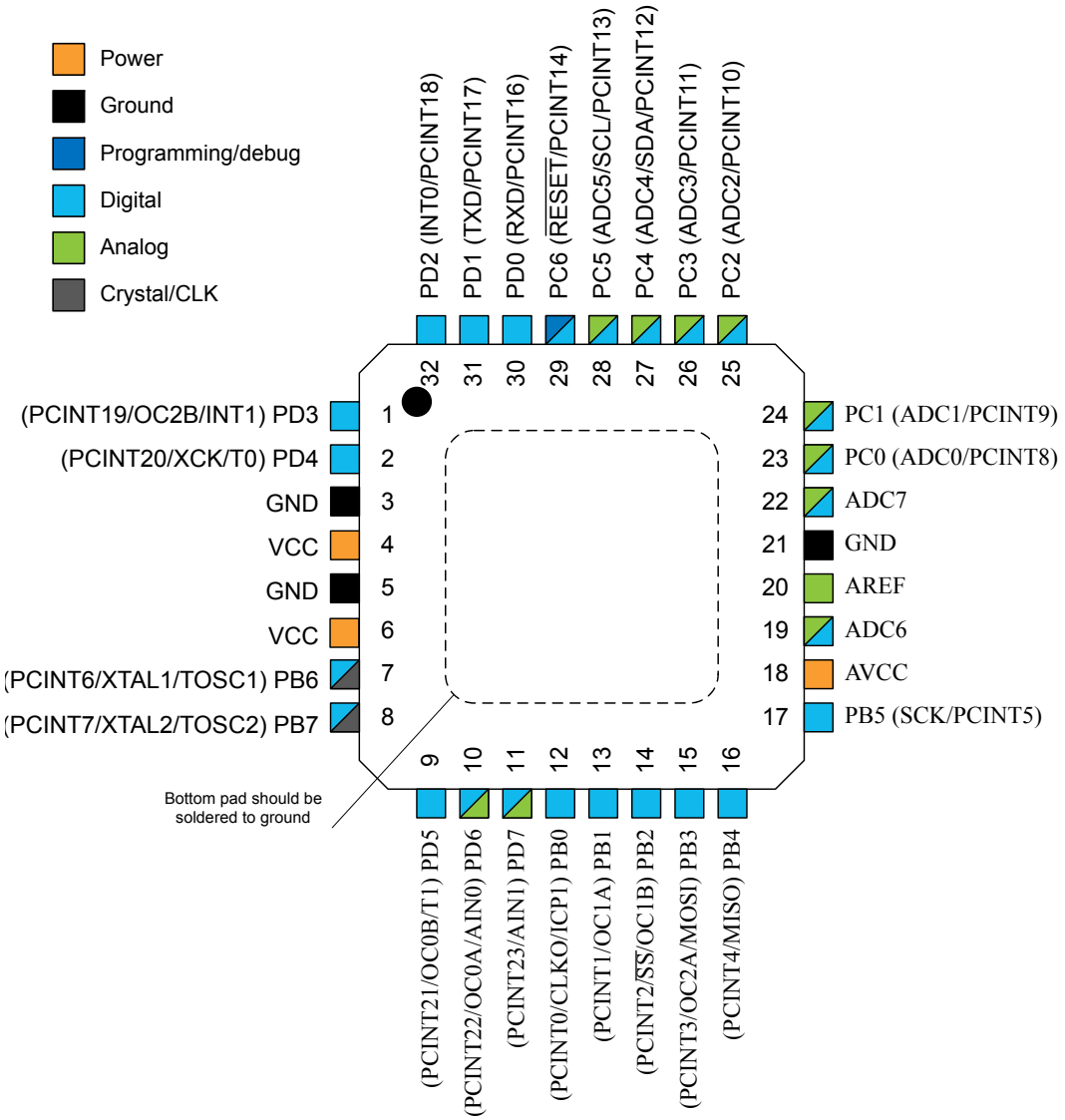


Figure 5-4. 32-pin MLF Top View



5.2. Pin Descriptions

5.2.1. VCC

Digital supply voltage.

5.2.2. GND

Ground.

5.2.3. Port B (PB[7:0]) XTAL1/XTAL2/TOSC1/TOSC2

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier.

If the Internal Calibrated RC Oscillator is used as chip clock source, PB[7:6] is used as TOSC[2:1] input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

5.2.4. Port C (PC[5:0])

Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC[5:0] output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

5.2.5. PC6/$\overline{\text{RESET}}$

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C.

If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. Shorter pulses are not guaranteed to generate a Reset.

The various special features of Port C are elaborated in the *Alternate Functions of Port C* section.

5.2.6. Port D (PD[7:0])

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

5.2.7. AV_{CC}

AV_{CC} is the supply voltage pin for the A/D Converter, PC[3:0], and PE[3:2]. It should be externally connected to V_{CC}, even if the ADC is not used. If the ADC is used, it should be connected to V_{CC} through a low-pass filter. Note that PC[6:4] use digital supply voltage, V_{CC}.

5.2.8. AREF

AREF is the analog reference pin for the A/D Converter.

5.2.9. ADC[7:6] (TQFP and VFQFN Package Only)

In the TQFP and VFQFN package, ADC[7:6] serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

14. PM - Power Management and Sleep Modes

14.1. Overview

Sleep modes enable the application to shut down unused modules in the MCU, thereby saving power. The device provides various sleep modes allowing the user to tailor the power consumption to the application requirements.

When enabled, the Brown-out Detector (BOD) actively monitors the power supply voltage during the sleep periods. To further save power, it is possible to disable the BOD in some sleep modes. See also BOD Disable.

Note: BOD disable is only available for ATmega328P.

14.2. Sleep Modes

The following Table shows the different sleep modes, BOD disable ability and their wake-up sources.

Table 14-1. Active Clock Domains and Wake-up Sources in the Different Sleep Modes.

	Active Clock Domains					Oscillators		Wake-up Sources							Software BOD Disable
Sleep Mode	clk_{CPU}	clk_{FLASH}	clk_{IO}	clk_{ADC}	clk_{ASY}	Main Clock Source Enabled	Timer Oscillator Enabled	INT and PCINT	TWI Address Match	Timer2	SPM/EEPROM Ready	ADC	WDT	Other I/O	
Idle			Yes	Yes	Yes	Yes	Yes(2)	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
ADC Noise Reduction				Yes	Yes	Yes	Yes(2)	Yes(3)	Yes	Yes(2)	Yes	Yes	Yes		
Power-down								Yes(3)	Yes				Yes		Yes
Power-save					Yes		Yes(2)	Yes(3)	Yes	Yes			Yes		Yes
Standby(1)						Yes		Yes(3)	Yes				Yes		Yes
Extended Standby					Yes(2)	Yes	Yes(2)	Yes(3)	Yes	Yes			Yes		Yes

Note:

1. Only recommended with external crystal or resonator selected as clock source.
2. If Timer/Counter2 is running in asynchronous mode.
3. For INT1 and INT0, only level interrupt.

To enter any of the six sleep modes, the Sleep Enable bit in the Sleep Mode Control Register (SMCR.SE) must be written to '1' and a SLEEP instruction must be executed. Sleep Mode Select bits (SMCR.SM[2:0]) select which sleep mode (Idle, ADC Noise Reduction, Power-down, Power-save, Standby, or Extended Standby) will be activated by the SLEEP instruction.

Note: The block diagram in the section *System Clock and Clock Options* provides an overview over the different clock systems in the device, and their distribution. This figure is helpful in selecting an appropriate sleep mode.

If an enabled interrupt occurs while the MCU is in a sleep mode, the MCU wakes up. The MCU is then halted for four cycles in addition to the start-up time, executes the interrupt routine, and resumes execution from the instruction following SLEEP. The contents of the Register File and SRAM are unaltered when the device wakes up from sleep. If a reset occurs during sleep mode, the MCU wakes up and executes from the Reset Vector.

Related Links

System Clock and Clock Options on page 48

14.3. BOD Disable

When the Brown-out Detector (BOD) is enabled by BODLEVEL fuses (see also section *Fuse Bits*), the BOD is actively monitoring the power supply voltage during a sleep period. To save power, it is possible to disable the BOD by software for some of the sleep modes. The sleep mode power consumption will then be at the same level as when BOD is globally disabled by fuses. If BOD is disabled in software, the BOD function is turned off immediately after entering the sleep mode. Upon wake-up from sleep, BOD is automatically enabled again. This ensures safe operation in case the V_{CC} level has dropped during the sleep period.

When the BOD has been disabled, the wake-up time from sleep mode will be approximately 60µs to ensure that the BOD is working correctly before the MCU continues executing code.

BOD disable is controlled by the BOD Sleep bit in the MCU Control Register (MCUCR.BODS). Writing this bit to '1' turns off the BOD in relevant sleep modes, while a zero in this bit keeps BOD active. The default setting, BODS=0, keeps BOD active.

Note: Writing to the BODS bit is controlled by a timed sequence and an enable bit.

Note: BOD disable is only available for ATmega328P.

Related Links

MCUCR on page 69

14.4. Idle Mode

When the SM[2:0] bits are written to '000', the SLEEP instruction makes the MCU enter Idle mode, stopping the CPU but allowing the SPI, USART, Analog Comparator, 2-wire Serial Interface, Timer/Counters, Watchdog, and the interrupt system to continue operating. This sleep mode basically halts clk_{CPU} and clk_{FLASH}, while allowing the other clocks to run.

Idle mode enables the MCU to wake up from external triggered interrupts as well as internal ones like the Timer Overflow and USART Transmit Complete interrupts. If wake-up from the Analog Comparator interrupt is not required, the Analog Comparator can be powered down by setting the ACD bit in the Analog Comparator Control and Status Register – ACSR. This will reduce power consumption in Idle mode.

14.5. ADC Noise Reduction Mode

When the SM[2:0] bits are written to '001', the SLEEP instruction makes the MCU enter ADC Noise Reduction mode, stopping the CPU but allowing the ADC, the external interrupts, the 2-wire Serial Interface address watch, Timer/Counter2[(1)], and the Watchdog to continue operating (if enabled). This sleep mode basically halts $clk_{I/O}$, clk_{CPU}, and clk_{FLASH}, while allowing the other clocks to run.

This improves the noise environment for the ADC, enabling higher resolution measurements. If the ADC is enabled, a conversion starts automatically when this mode is entered. Apart from the ADC Conversion Complete interrupt, only these events can wake up the MCU from ADC Noise Reduction mode:

- External Reset
- Watchdog System Reset
- Watchdog Interrupt
- Brown-out Reset
- 2-wire Serial Interface address match

- Timer/Counter2 interrupt
- SPM/EEPROM ready interrupt
- External level interrupt on INT
- Pin change interrupt

Note: 1. Timer/Counter2 will only keep running in asynchronous mode.

Related Links

8-bit Timer/Counter2 with PWM and Asynchronous Operation on page 189

14.6. Power-Down Mode

When the SM[2:0] bits are written to '010', the SLEEP instruction makes the MCU enter Power-Down mode. In this mode, the external Oscillator is stopped, while the external interrupts, the 2-wire Serial Interface address watch, and the Watchdog continue operating (if enabled).

Only one of these events can wake up the MCU:

- External Reset
- Watchdog System Reset
- Watchdog Interrupt
- Brown-out Reset
- 2-wire Serial Interface address match
- External level interrupt on INT
- Pin change interrupt

This sleep mode basically halts all generated clocks, allowing operation of asynchronous modules only.

Note: If a level triggered interrupt is used for wake-up from Power-Down, the required level must be held long enough for the MCU to complete the wake-up to trigger the level interrupt. If the level disappears before the end of the Start-up Time, the MCU will still wake up, but no interrupt will be generated. The start-up time is defined by the SUT and CKSEL Fuses.

When waking up from Power-Down mode, there is a delay from the wake-up condition occurs until the wake-up becomes effective. This allows the clock to restart and become stable after having been stopped. The wake-up period is defined by the same CKSEL Fuses that define the Reset Time-out period.

Related Links

System Clock and Clock Options on page 48

14.7. Power-save Mode

When the SM[2:0] bits are written to 011, the SLEEP instruction makes the MCU enter Power-save mode. This mode is identical to Power-down, with one exception:

If Timer/Counter2 is enabled, it will keep running during sleep. The device can wake up from either Timer Overflow or Output Compare event from Timer/Counter2 if the corresponding Timer/Counter2 interrupt enable bits are set in TIMSK2, and the Global Interrupt Enable bit in SREG is set.

If Timer/Counter2 is not running, Power-down mode is recommended instead of Power-save mode.

The Timer/Counter2 can be clocked both synchronously and asynchronously in Power-save mode. If Timer/Counter2 is not using the asynchronous clock, the Timer/Counter Oscillator is stopped during

sleep. If Timer/Counter2 is not using the synchronous clock, the clock source is stopped during sleep. Even if the synchronous clock is running in Power-save, this clock is only available for Timer/Counter2.

14.8. Standby Mode

When the SM[2:0] bits are written to '110' and an external crystal/resonator clock option is selected, the SLEEP instruction makes the MCU enter Standby mode. This mode is identical to Power-Down with the exception that the Oscillator is kept running. From Standby mode, the device wakes up in six clock cycles.

14.9. Extended Standby Mode

When the SM[2:0] bits are written to '111' and an external crystal/resonator clock option is selected, the SLEEP instruction makes the MCU enter Extended Standby mode. This mode is identical to Power-Save mode with the exception that the Oscillator is kept running. From Extended Standby mode, the device wakes up in six clock cycles.

14.10. Power Reduction Register

The Power Reduction Register (PRR) provides a method to stop the clock to individual peripherals to reduce power consumption. The current state of the peripheral is frozen and the I/O registers can not be read or written. Resources used by the peripheral when stopping the clock will remain occupied, hence the peripheral should in most cases be disabled before stopping the clock. Waking up a module, which is done by clearing the corresponding bit in the PRR, puts the module in the same state as before shutdown.

Module shutdown can be used in Idle mode and Active mode to significantly reduce the overall power consumption. In all other sleep modes, the clock is already stopped.

14.11. Minimizing Power Consumption

There are several possibilities to consider when trying to minimize the power consumption in an AVR controlled system. In general, sleep modes should be used as much as possible, and the sleep mode should be selected so that as few as possible of the device's functions are operating. All functions not needed should be disabled. In particular, the following modules may need special consideration when trying to achieve the lowest possible power consumption.

14.11.1. Analog to Digital Converter

If enabled, the ADC will be enabled in all sleep modes. To save power, the ADC should be disabled before entering any sleep mode. When the ADC is turned off and on again, the next conversion will be an extended conversion.

Related Links

Analog-to-Digital Converter on page 305

14.11.2. Analog Comparator

When entering Idle mode, the Analog Comparator should be disabled if not used. When entering ADC Noise Reduction mode, the Analog Comparator should be disabled. In other sleep modes, the Analog Comparator is automatically disabled. However, if the Analog Comparator is set up to use the Internal Voltage Reference as input, the Analog Comparator should be disabled in all sleep modes. Otherwise, the Internal Voltage Reference will be enabled, independent of sleep mode.

Related Links

Analog Comparator on page 299

14.11.3. Brown-Out Detector

If the Brown-Out Detector (BOD) is not needed by the application, this module should be turned off. If the BOD is enabled by the BODLEVEL Fuses, it will be enabled in all sleep modes, and hence, always consume power. In the deeper sleep modes, this will contribute significantly to the total current consumption.

Related Links

System Control and Reset on page 72

14.11.4. Internal Voltage Reference

The Internal Voltage Reference will be enabled when needed by the Brown-Out Detection, the Analog Comparator or the Analog-to-Digital Converter. If these modules are disabled as described in the sections above, the internal voltage reference will be disabled and it will not be consuming power. When turned on again, the user must allow the reference to start up before the output is used. If the reference is kept on in sleep mode, the output can be used immediately.

Related Links

System Control and Reset on page 72

14.11.5. Watchdog Timer

If the Watchdog Timer is not needed in the application, the module should be turned off. If the Watchdog Timer is enabled, it will be enabled in all sleep modes and hence always consume power. In the deeper sleep modes, this will contribute significantly to the total current consumption.

Related Links

System Control and Reset on page 72

14.11.6. Port Pins

When entering a sleep mode, all port pins should be configured to use minimum power. The most important is then to ensure that no pins drive resistive loads. In sleep modes where both the I/O clock ($clk_{I/O}$) and the ADC clock (clk_{ADC}) are stopped, the input buffers of the device will be disabled. This ensures that no power is consumed by the input logic when not needed. In some cases, the input logic is needed for detecting wake-up conditions, and it will then be enabled. Refer to the section *Digital Input Enable and Sleep Modes* for details on which pins are enabled. If the input buffer is enabled and the input signal is left floating or have an analog signal level close to $V_{CC}/2$, the input buffer will use excessive power.

For analog input pins, the digital input buffer should be disabled at all times. An analog signal level close to $V_{CC}/2$ on an input pin can cause significant current even in active mode. Digital input buffers can be disabled by writing to the Digital Input Disable Registers (DIDR0 for ADC, DIDR1 for AC).

Related Links

Digital Input Enable and Sleep Modes on page 101

14.11.7. On-chip Debug System

If the On-chip debug system is enabled by the Fuse and the chip enters sleep mode, the main clock source is enabled and hence always consumes power. In the deeper sleep modes, this will contribute significantly to the total current consumption.

15.8. Watchdog Timer

If the watchdog timer is not needed in the application, the module should be turned off. If the watchdog timer is enabled, it will be enabled in all sleep modes and hence always consume power. In the deeper sleep modes, this will contribute significantly to the total current consumption.

Refer to Watchdog System Reset for details on how to configure the watchdog timer.

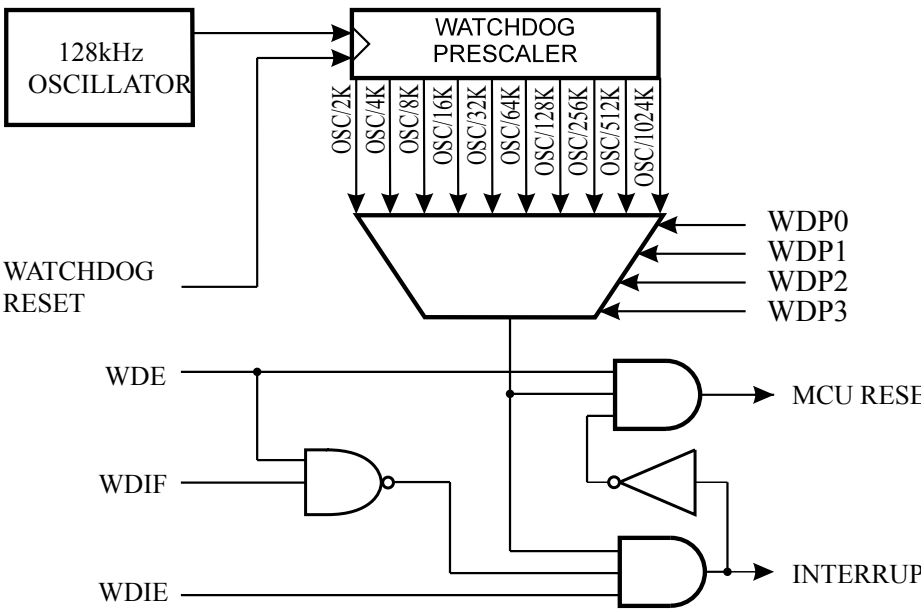
15.8.1. Features

- Clocked from separate On-chip Oscillator
- Three operating modes:
  - Interrupt
  - System Reset
  - Interrupt and System Reset
- Selectable Time-out period from 16ms to 8s
- Possible Hardware fuse Watchdog always on (WDTON) for fail-safe mode

15.8.2. Overview

The device has an Enhanced Watchdog Timer (WDT). The WDT is a timer counting cycles of a separate on-chip 128kHz oscillator. The WDT gives an interrupt or a system reset when the counter reaches a given time-out value. In normal operation mode, it is required that the system uses the Watchdog Timer Reset (WDR) instruction to restart the counter before the time-out value is reached. If the system doesn't restart the counter, an interrupt or system reset will be issued.

Figure 15-7. Watchdog Timer



In Interrupt mode, the WDT gives an interrupt when the timer expires. This interrupt can be used to wake the device from sleep-modes, and also as a general system timer. One example is to limit the maximum time allowed for certain operations, giving an interrupt when the operation has run longer than expected. In System Reset mode, the WDT gives a reset when the timer expires. This is typically used to prevent system hang-up in case of runaway code. The third mode, Interrupt and System Reset mode, combines the other two modes by first giving an interrupt and then switch to System Reset mode. This mode will for instance allow a safe shutdown by saving critical parameters before a system reset.

The Watchdog always on (WDTON) fuse, if programmed, will force the Watchdog Timer to System Reset mode. With the fuse programmed the System Reset mode bit (WDE) and Interrupt mode bit (WDIE) are locked to 1 and 0 respectively. To further ensure program security, alterations to the Watchdog set-up must follow timed sequences. The sequence for clearing WDE and changing time-out configuration is as follows:

1. In the same operation, write a logic one to the Watchdog change enable bit (WDCE) and Watchdog System Reset Enable (WDE) in Watchdog Timer Control Register (WDTCSR.WDCE and WDTCSR.WDE). A logic one must be written to WDTCSR.WDE regardless of the previous value of the WDTCSR.WDE.
2. Within the next four clock cycles, write the WDTCSR.WDE and Watchdog prescaler bits group (WDTCSR.WDP) as desired, but with the WDTCSR.WDCE cleared. This must be done in one operation.

The following examples show a function for turning off the Watchdog Timer. The examples assume that interrupts are controlled (e.g. by disabling interrupts globally) so that no interrupts will occur during the execution of these functions.

Assembly Code Example

```
WDT_off:
   ; Turn off global interrupt
   cli
   ; Reset Watchdog Timer
   wdr
   ; Clear WDRF in MCUSR
   in      r16, MCUSR
   andi    r16, (0xff & (0<<WDRF))
   out     MCUSR, r16
   ; Write '1' to WDCE and WDE
   ; Keep old prescaler setting to prevent unintentional time-out
   lds     r16, WDTCSR
   ori     r16, (1<<WDCE) | (1<<WDE)
   sts     WDTCSR, r16
   ; Turn off WDT
   ldi     r16, (0<<WDE)
   sts     WDTCSR, r16
   ; Turn on global interrupt
   sei
   ret
```

C Code Example

```
void WDT_off(void)
{
    __disable_interrupt();
    __watchdog_reset();
    /* Clear WDRF in MCUSR */
    MCUSR &= ~(1<<WDRF);
    /* Write logical one to WDCE and WDE */
    /* Keep old prescaler setting to prevent unintentional time-out */
    WDTCSR |= (1<<WDCE) | (1<<WDE);
    /* Turn off WDT */
    WDTCSR = 0x00;
    __enable_interrupt();
}
```

Note: If the Watchdog is accidentally enabled, for example by a runaway pointer or brown-out condition, the device will be reset and the Watchdog Timer will stay enabled. If the code is not set up to handle the Watchdog, this might lead to an eternal loop of time-out resets. To avoid this situation, the application software should always clear the Watchdog System Reset Flag (WDRF) and the WDE control bit in the initialization routine, even if the Watchdog is not in use.

The following code examples shows how to change the time-out value of the Watchdog Timer.

Assembly Code Example

```
WDT_Prescaler_Change:
   ; Turn off global interrupt
   cli
   ; Reset Watchdog Timer
   wdr
   ; Start timed sequence
   lds r16, WDTCSR
   ori r16, (1<<WDCE) | (1<<WDE)
   sts WDTCSR, r16
   ; -- Got four cycles to set the new values from here -
   ; Set new prescaler(time-out) value = 64K cycles (~0.5 s)
   ldi r16, (1<<WDE) | (1<<WDP2) | (1<<WDP0)
   sts WDTCSR, r16
   ; -- Finished setting new values, used 2 cycles -
   ; Turn on global interrupt
   sei
   ret
```

C Code Example

```
void WDT_Prescaler_Change(void)
{
    __disable_interrupt();
    __watchdog_reset();
    /* Start timed sequence */
    WDTCSR |= (1<<WDCE) | (1<<WDE);
    /* Set new prescaler(time-out) value = 64K cycles (~0.5 s) */
    WDTCSR = (1<<WDE) | (1<<WDP2) | (1<<WDP0);
    __enable_interrupt();
}
```

Note: The Watchdog Timer should be reset before any change of the WDTCSR.WDP bits, since a change in the WDTCSR.WDP bits can result in a time-out when switching to a shorter time-out period.

15.9. Register Description

15.9.1. MCU Status Register

To make use of the Reset Flags to identify a reset condition, the user should read and then Reset the MCUSR as early as possible in the program. If the register is cleared before another reset occurs, the source of the reset can be found by examining the Reset Flags.

When addressing I/O Registers as data space using LD and ST instructions, the provided offset must be used. When using the I/O specific commands IN and OUT, the offset is reduced by 0x20, resulting in an I/O address offset within 0x00 - 0x3F.

Name: MCUSR
Offset: 0x54
Reset: 0x00
Property: When addressing as I/O Register: address offset is 0x34

Bit	7	6	5	4	3	2	1	0
					WDRF	BORF	EXTRF	PORF
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit 3 – WDRF: Watchdog System Reset Flag

This bit is set if a Watchdog System Reset occurs. The bit is reset by a Power-on Reset, or by writing a '0' to it.

Bit 2 – BORF: Brown-out Reset Flag

This bit is set if a Brown-out Reset occurs. The bit is reset by a Power-on Reset, or by writing a '0' to it.

Bit 1 – EXTRF: External Reset Flag

This bit is set if an External Reset occurs. The bit is reset by a Power-on Reset, or by writing a '0' to it.

Bit 0 – PORF: Power-on Reset Flag

This bit is set if a Power-on Reset occurs. The bit is reset only by writing a '0' to it.

15.9.2. WDTCSR – Watchdog Timer Control Register

Name: WDTCSR
Offset: 0x60
Reset: 0x00
Property:

Bit	7	6	5	4	3	2	1	0
	WDIF	WDIE	WDP[3]	WDCE	WDE	WDP[2:0]		
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

Bit 7 – WDIF: Watchdog Interrupt Flag
This bit is set when a timeout occurs in the Watchdog Timer and the Watchdog Timer is configured for interrupt. WDIF is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, WDIF is cleared by writing a '1' to it. When the I-bit in SREG and WDIE are set, the Watchdog Timeout Interrupt is executed.

Bit 6 – WDIE: Watchdog Interrupt Enable
When this bit is written to '1' and the I-bit in the Status Register is set, the Watchdog Interrupt is enabled. If WDE is cleared in combination with this setting, the Watchdog Timer is in Interrupt Mode, and the corresponding interrupt is executed if timeout in the Watchdog Timer occurs. If WDE is set, the Watchdog Timer is in Interrupt and System Reset Mode. The first timeout in the Watchdog Timer will set WDIF. Executing the corresponding interrupt vector will clear WDIE and WDIF automatically by hardware (the Watchdog goes to System Reset Mode).

This is useful for keeping the Watchdog Timer security while using the interrupt. To stay in Interrupt and System Reset Mode, WDIE must be set after each interrupt. This should however not be done within the interrupt service routine itself, as this might compromise the safety function of the Watchdog System Reset mode. If the interrupt is not executed before the next timeout, a System Reset will be applied.

Table 15-1. Watchdog Timer Configuration

WDTON(1)	WDE	WDIE	Mode	Action on Time-out
1	0	0	Stopped	None
1	0	1	Interrupt Mode	Interrupt
1	1	0	System Reset Mode	Reset
1	1	1	Interrupt and System Reset Mode	Interrupt, then go to System Reset Mode
0	x	x	System Reset Mode	Reset

Note: 1. WDTON Fuse set to '0' means programmed and '1' means unprogrammed.

Bit 5 – WDP[3]: Watchdog Timer Prescaler 3

Bit 4 – WDCE: Watchdog Change Enable
This bit is used in timed sequences for changing WDE and prescaler bits. To clear the WDE bit, and/or change the prescaler bits, WDCE must be set. Once written to '1', hardware will clear WDCE after four clock cycles.

Bit 3 – WDE: Watchdog System Reset Enable

WDE is overridden by WDRF in MCUSR. This means that WDE is always set when WDRF is set. To clear WDE, WDRF must be cleared first. This feature ensures multiple resets during conditions causing failure, and a safe startup after the failure.

Bits 2:0 – WDP[2:0]: Watchdog Timer Prescaler 2, 1, and 0

The WDP[3:0] bits determine the Watchdog Timer prescaling when the Watchdog Timer is running. The different prescaling values and their corresponding timeout periods are shown in the following table.

Table 15-2. Watchdog Timer Prescale Select

WDP[3]	WDP[2]	WDP[1]	WDP[0]	Number of WDT Oscillator (Cycles)	Oscillator
0	0	0	0	2K (2048)	16ms
0	0	0	1	4K (4096)	32ms
0	0	1	0	8K (8192)	64ms
0	0	1	1	16K (16384)	0.125s
0	1	0	0	32K (32768)	0.25s
0	1	0	1	64K (65536)	0.5s
0	1	1	0	128K (131072)	1.0s
0	1	1	1	256K (262144)	2.0s
1	0	0	0	512K (524288)	4.0s
1	0	0	1	1024K (1048576)	8.0s
1	0	1	0	Reserved	
1	0	1	1		
1	1	0	0		
1	1	0	1		
1	1	1	0		
1	1	1	1		

16. Interrupts

This section describes the specifics of the interrupt handling of the device. For a general explanation of the AVR interrupt handling, refer to the description of *Reset and Interrupt Handling*.

- Each Interrupt Vector occupies two instruction words .
- Reset Vector is affected by the BOOTRST fuse, and the Interrupt Vector start address is affected by the IVSEL bit in MCUCR

16.1. Interrupt Vectors in ATmega328/P

Table 16-1. Reset and Interrupt Vectors in ATmega328/P

Vector No	Program Address(2)	Source	Interrupts definition
1	0x0000(1)	RESET	External Pin, Power-on Reset, Brown-out Reset and Watchdog System Reset
2	0x0002	INT0	External Interrupt Request 0
3	0x0004	INT1	External Interrupt Request 0
4	0x0006	PCINT0	Pin Change Interrupt Request 0
5	0x0008	PCINT1	Pin Change Interrupt Request 1
6	0x000A	PCINT2	Pin Change Interrupt Request 2
7	0x000C	WDT	Watchdog Time-out Interrupt
8	0x000E	TIMER2_COMPA	Timer/Counter2 Compare Match A
9	0x0010	TIMER2_COMPB	Timer/Coutner2 Compare Match B
10	0x0012	TIMER2_OVF	Timer/Counter2 Overflow
11	0x0014	TIMER1_CAPT	Timer/Counter1 Capture Event
12	0x0016	TIMER1_COMPA	Timer/Counter1 Compare Match A
13	0x0018	TIMER1_COMPB	Timer/Coutner1 Compare Match B
14	0x001A	TIMER1_OVF	Timer/Counter1 Overflow
15	0x001C	TIMER0_COMPA	Timer/Counter0 Compare Match A
16	0x001E	TIMER0_COMPB	Timer/Coutner0 Compare Match B
17	0x0020	TIMER0_OVF	Timer/Counter0 Overflow
18	0x0022	SPI STC	SPI Serial Transfer Complete
19	0x0024	USART_RX	USART Rx Complete
20	0x0026	USART_UDRE	USART Data Register Empty
21	0x0028	USART_TX	USART Tx Complete
22	0x002A	ADC	ADC Conversion Complete
23	0x002C	EE READY	EEPROM Ready
24	0x002E	ANALOG COMP	Analog Comparator

Vector No	Program Address(2)	Source	Interrupts definition
25	0x0030	TWI	2-wire Serial Interface (I^2C)
26	0x0032	SPM READY	Store Program Memory Ready

Note:

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

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

Table 16-2. Reset and Interrupt Vectors Placement

BOOTRST(1)	IVSEL	Reset Address	Interrupt Vectors Start Address
1	0	0x000	0x002
1	1	0x000	Boot Reset Address + 0x0002
0	0	Boot Reset Address	0x002
0	1	Boot Reset Address	Boot Reset Address + 0x0002

Note: 1. For the BOOTRST Fuse “1” means unprogrammed while “0” means programmed.

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

```
Address   Labels    Code                        Comments
0x0000              jmp     RESET               ; Reset
0x0002              jmp     INT0                ; IRQ0
0x0004              jmp     INT1                ; IRQ1
0x0006              jmp     PCINT0              ; PCINT0
0x0008              jmp     PCINT1              ; PCINT1
0x000A              jmp     PCINT2              ; PCINT2
0x000C              jmp     WDT                 ; Watchdog Timeout
0x000E              jmp     TIM2_COMPA          ; Timer2 CompareA
0x0010              jmp     TIM2_COMPB          ; Timer2 CompareB
0x0012              jmp     TIM2_OVF            ; Timer2 Overflow
0x0014              jmp     TIM1_CAPT           ; Timer1 Capture
0x0016              jmp     TIM1_COMPA          ; Timer1 CompareA
0x0018              jmp     TIM1_COMPB          ; Timer1 CompareB
0x001A              jmp     TIM1_OVF            ; Timer1 Overflow
0x001C              jmp     TIM0_COMPA          ; Timer0 CompareA
0x001E              jmp     TIM0_COMPB          ; Timer0 CompareB
0x0020              jmp     TIM0_OVF            ; Timer0 Overflow
0x0022              jmp     SPI_STC             ; SPI Transfer Complete
0x0024              jmp     USART_RXC           ; USART RX Complete
0x0026              jmp     USART_UDRE          ; USART UDR Empty
0x0028              jmp     USART_TXC           ; USART TX Complete
0x002A              jmp     ADC                 ; ADC Conversion Complete
0x002C              jmp     EE_RDY              ; EEPROM Ready
0x002E              jmp     ANA_COMP            ; Analog Comparator
0x0030              jmp     TWI                 ; 2-wire Serial
0x0032              jmp     SPM_RDY             ; SPM Ready
;
0x0034    RESET:    ldi     r16,high(RAMEND)    ; Main program start
0x0035              out     SPH,r16             ; Set Stack Pointer to top of RAM
0x0036              ldi     r16,low(RAMEND)
0x0037              out     SPL,r16
0x0038              sei                         ; Enable interrupts
```

```
0x0039              <instr>  xxx
...       ...       ...                          ...
```

When the BOOTRST Fuse is unprogrammed, the Boot section size set to 2Kbytes and the MCUCR.IVSEL is set before any interrupts are enabled, the most typical and general program setup for the Reset and Interrupt Vector Addresses is:

```
Address   Labels     Code                          Comments
0x0000   RESET:     ldi      r16,high(RAMEND)    ; Main program start
0x0001              out      SPH,r16             ; Set Stack Pointer to top of RAM
0x0002              ldi      r16,low(RAMEND)
0x0003              out      SPL,r16
0x0004              sei                          ; Enable interrupts
0x0005              <instr> xxx
;
.org 0x3C02
0x3C02              jmp      EXT_INT0            ; IRQ0 Handler
0x3C04              jmp      EXT_INT1            ; IRQ1 Handler
...                  ...      ...                 ;
0x3C32              jmp      SPM_RDY             ; SPM Ready Handler
```

When the BOOTRST Fuse is programmed and the Boot section size set to 2Kbytes, the most typical and general program setup for the Reset and Interrupt Vector Addresses is:

```
Address   Labels     Code                          Comments
.org 0x0002
0x0002              jmp      EXT_INT0            ; IRQ0 Handler
0x0004              jmp      EXT_INT1            ; IRQ1 Handler
...                  ...      ...                 ;
0x0032              jmp      SPM_RDY             ; SPM Ready Handler
;
.org  0x3C00
0x3C00    RESET:    ldi      r16,high(RAMEND)    ; Main program start
0x3C01              out      SPH,r16             ; Set Stack Pointer to top of RAM
0x3C02              ldi      r16,low(RAMEND)
0x3C03              out      SPL,r16
0x3C04              sei                          ; Enable interrupts
0x3C05              <instr>  xxx
```

When the BOOTRST Fuse is programmed, the Boot section size set to 2K bytes and the MCUCR.IVSEL is set before any interrupts are enabled, the most typical and general program setup for the Reset and Interrupt Vector Addresses is:

```
Address    Labels    Code                          Comments
;
.org 0x3C00
0x3C00              jmp      RESET               ; Reset handler
0x3C02              jmp      EXT_INT0            ; IRQ0 Handler
0x3C04              jmp      EXT_INT1            ; IRQ1 Handler
...                  ...      ...                 ;
0x3C32              jmp      SPM_RDY             ; SPM Ready Handler
;
0x3C34    RESET:    ldi      r16,high(RAMEND)    ; Main program start
0x3C35              out      SPH,r16             ; Set Stack Pointer to top of RAM
0x3C36              ldi      r16,low(RAMEND)
0x3C37              out      SPL,r16
0x3C38              sei                          ; Enable interrupts
0x3C39              <instr>  xxx
```

16.2. Register Description

16.2.1. Moving Interrupts Between Application and Boot Space

The MCU Control Register controls the placement of the Interrupt Vector table.

16.2.2. MCU Control Register

When addressing I/O Registers as data space using LD and ST instructions, the provided offset must be used. When using the I/O specific commands IN and OUT, the offset is reduced by 0x20, resulting in an I/O address offset within 0x00 - 0x3F.

Name: MCUCR
Offset: 0x55
Reset: 0x00
Property: When addressing as I/O Register: address offset is 0x35

Bit	7	6	5	4	3	2	1	0
		BODS	BODSE	PUD			IVSEL	IVCE
Access		R/W	R/W	R/W			R/W	R/W
Reset		0	0	0			0	0

Bit 6 – BODS: BOD Sleep

The BODS bit must be written to '1' in order to turn off BOD during sleep. Writing to the BODS bit is controlled by a timed sequence and the enable bit BODSE. To disable BOD in relevant sleep modes, both BODS and BODSE must first be written to '1'. Then, BODS must be written to '1' and BODSE must be written to zero within four clock cycles.

The BODS bit is active three clock cycles after it is set. A sleep instruction must be executed while BODS is active in order to turn off the BOD for the actual sleep mode. The BODS bit is automatically cleared after three clock cycles.

Note: BOD disable is only available for ATmega328P.

Bit 5 – BODSE: BOD Sleep Enable

BODSE enables setting of BODS control bit, as explained in BODS bit description. BOD disable is controlled by a timed sequence.

Note: BOD disable is only available for ATmega328P.

Bit 4 – PUD: Pull-up Disable

When this bit is written to one, the pull-ups in the I/O ports are disabled even if the DDxn and PORTxn Registers are configured to enable the pull-ups ({DDxn, PORTxn} = 0b01).

Bit 1 – IVSEL: Interrupt Vector Select

When the IVSEL bit is cleared (zero), the Interrupt Vectors are placed at the start of the Flash memory. When this bit is set (one), the Interrupt Vectors are moved to the beginning of the Boot Loader section of the Flash. The actual address of the start of the Boot Flash Section is determined by the BOOTSZ Fuses. To avoid unintentional changes of Interrupt Vector tables, a special write procedure must be followed to change the IVSEL bit:

1. Write the Interrupt Vector Change Enable (IVCE) bit to one.
2. Within four cycles, write the desired value to IVSEL while writing a zero to IVCE.

Interrupts will automatically be disabled while this sequence is executed. Interrupts are disabled in the cycle IVCE is set, and they remain disabled until after the instruction following the write to IVSEL. If IVSEL is not written, interrupts remain disabled for four cycles. The I-bit in the Status Register is unaffected by the automatic disabling.

Note: If Interrupt Vectors are placed in the Boot Loader section and Boot Lock bit BLB02 is programmed, interrupts are disabled while executing from the Application section. If Interrupt Vectors are placed in the Application section and Boot Lock bit BLB12 is programed, interrupts are disabled while executing from the Boot Loader section.

Bit 0 – IVCE: Interrupt Vector Change Enable

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

Assembly Code Example

```
Move_interrupts:
; Get MCUCR
in    r16, MCUCR
mov   r17, r16
; Enable change of Interrupt Vectors
ori   r16, (1<<IVCE)
out   MCUCR, r16
; Move interrupts to Boot Flash section
ori   r17, (1<<IVSEL)
out   MCUCR, r17
ret
```

C Code Example

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

17. EXINT - External Interrupts

The External Interrupts are triggered by the INT pins or any of the PCINT pins. Observe that, if enabled, the interrupts will trigger even if the INT or PCINT pins are configured as outputs. This feature provides a way of generating a software interrupt.

The Pin Change Interrupt Request 2 (PCI2) will trigger if any enabled PCINT[23:16] pin toggles. The Pin Change Interrupt Request 1 (PCI1) will trigger if any enabled PCINT[14:8] pin toggles. The Pin Change Interrupt Request 0 (PCI0) will trigger if any enabled PCINT[7:0] pin toggles. The PCMSK2, PCMSK1 and PCMSK0 Registers control which pins contribute to the pin change interrupts. Pin change interrupts on PCINT are detected asynchronously. This implies that these interrupts can be used for waking the part also from sleep modes other than Idle mode.

The External Interrupts can be triggered by a falling or rising edge or a low level. This is set up as indicated in the specification for the External Interrupt Control Register A (EICRA). When the External Interrupts are enabled and are configured as level triggered, the interrupts will trigger as long as the pin is held low. Note that recognition of falling or rising edge interrupts on INT requires the presence of an I/O clock. Low level interrupt on INT is detected asynchronously. This implies that this interrupt can be used for waking the part also from sleep modes other than Idle mode. The I/O clock is halted in all sleep modes except Idle mode.

Note: If a level triggered interrupt is used for wake-up from Power-down, the required level must be held long enough for the MCU to complete the wake-up to trigger the level interrupt. If the level disappears before the end of the Start-up Time, the MCU will still wake up, but no interrupt will be generated. The start-up time is defined by the SUT and CKSEL Fuses.

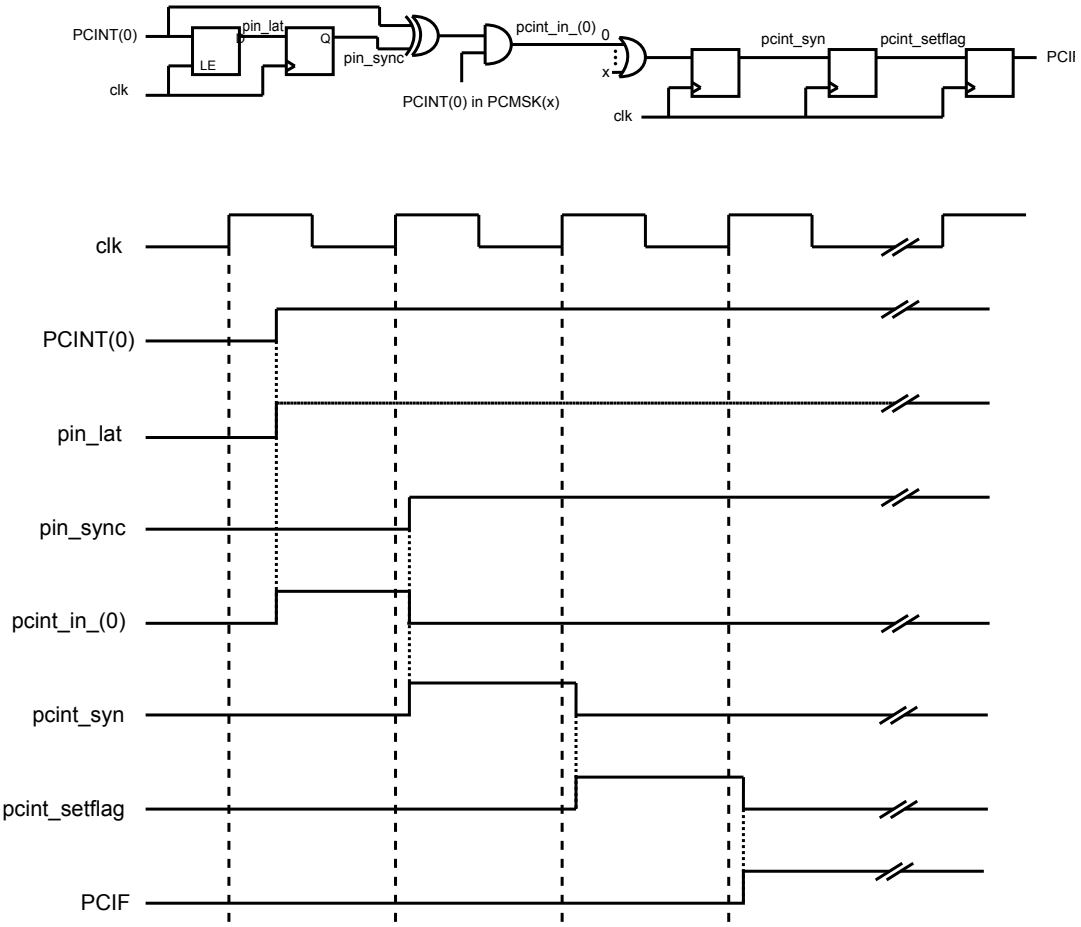
Related Links

System Control and Reset on page 72
Clock Systems and Their Distribution on page 48
System Clock and Clock Options on page 48

17.1. Pin Change Interrupt Timing

An example of timing of a pin change interrupt is shown in the following figure.

Figure 17-1. Timing of pin change interrupts



Related Links

System Control and Reset on page 72

Clock Systems and Their Distribution on page 48

System Clock and Clock Options on page 48

17.2. Register Description

17.2.1. External Interrupt Control Register A

The External Interrupt Control Register A contains control bits for interrupt sense control.

Name: EICRA
Offset: 0x69
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
					ISC11	ISC10	ISC01	ISC00
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bits 3:2 – ISC1n: Interrupt Sense Control 1 [n = 1:0]

The External Interrupt 1 is activated by the external pin INT1 if the SREG I-flag and the corresponding interrupt mask are set. The level and edges on the external INT1 pin that activate the interrupt are defined in the table below. The value on the 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.

Value	Description
00	The low level of INT1 generates an interrupt request.
01	Any logical change on INT1 generates an interrupt request.
10	The falling edge of INT1 generates an interrupt request.
11	The rising edge of INT1 generates an interrupt request.

Bits 1:0 – ISC0n: Interrupt Sense Control 0 [n = 1:0]

The External Interrupt 0 is activated by the external pin INT0 if the SREG I-flag and the corresponding interrupt mask are set. The level and edges on the external INT0 pin that activate the interrupt are defined in table below. The value on the INT0 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.

Value	Description
00	The low level of INT0 generates an interrupt request.
01	Any logical change on INT0 generates an interrupt request.
10	The falling edge of INT0 generates an interrupt request.
11	The rising edge of INT0 generates an interrupt request.

17.2.2. External Interrupt Mask Register

When addressing I/O Registers as data space using LD and ST instructions, the provided offset must be used. When using the I/O specific commands IN and OUT, the offset is reduced by 0x20, resulting in an I/O address offset within 0x00 - 0x3F.

Name: EIMSK
Offset: 0x3D
Reset: 0x00
Property: When addressing as I/O Register: address offset is 0x1D

Bit	7	6	5	4	3	2	1	0
							INT1	INT0
Access							R/W	R/W
Reset							0	0

Bit 1 – INT1: External Interrupt Request 1 Enable

When the INT1 bit is set and the I-bit in the Status Register (SREG) is set, the external pin interrupt is enabled. The Interrupt Sense Control1 bits 1/0 (ISC11 and ISC10) in the External Interrupt Control Register A (EICRA) 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 0 – INT0: External Interrupt Request 0 Enable

When the INT0 bit is set and the I-bit in the Status Register (SREG) is set, the external pin interrupt is enabled. The Interrupt Sense Control0 bits 1/0 (ISC01 and ISC00) in the External Interrupt Control Register A (EICRA) define whether the external interrupt is activated on rising and/or falling edge of the INT0 pin or level sensed. Activity on the pin will cause an interrupt request even if INT0 is configured as an output. The corresponding interrupt of External Interrupt Request 0 is executed from the INT0 Interrupt Vector.

17.2.3. External Interrupt Flag Register

When addressing I/O Registers as data space using LD and ST instructions, the provided offset must be used. When using the I/O specific commands IN and OUT, the offset is reduced by 0x20, resulting in an I/O address offset within 0x00 - 0x3F.

Name: EIFR
Offset: 0x3C
Reset: 0x00
Property: When addressing as I/O Register: address offset is 0x1C

Bit	7	6	5	4	3	2	1	0
							INTF1	INTF0
Access							R/W	R/W
Reset							0	0

Bit 1 – INTF1: External Interrupt Flag 1

When an edge or logic change on the INT1 pin triggers an interrupt request, INTF1 will be set. If the I-bit in SREG and the INT1 bit in EIMSK are set, the MCU will jump to the corresponding Interrupt Vector. The flag is cleared when the interrupt routine is executed. Alternatively, the flag can be cleared by writing '1' to it. This flag is always cleared when INT1 is configured as a level interrupt.

Bit 0 – INTF0: External Interrupt Flag 0

When an edge or logic change on the INT0 pin triggers an interrupt request, INTF0 will be set. If the I-bit in SREG and the INT0 bit in EIMSK are set, the MCU will jump to the corresponding Interrupt Vector. The flag is cleared when the interrupt routine is executed. Alternatively, the flag can be cleared by writing '1' to it. This flag is always cleared when INT0 is configured as a level interrupt.

17.2.4. Pin Change Interrupt Control Register

Name: PCICR
Offset: 0x68
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
						PCIE2	PCIE1	PCIE0
Access						R/W	R/W	R/W
Reset						0	0	0

Bit 2 – PCIE2: Pin Change Interrupt Enable 2
When the PCIE2 bit is set and the I-bit in the Status Register (SREG) is set, pin change interrupt 2 is enabled. Any change on any enabled PCINT[23:16] pin will cause an interrupt. The corresponding interrupt of Pin Change Interrupt Request is executed from the PCI2 Interrupt Vector. PCINT[23:16] pins are enabled individually by the PCMSK2 Register.

Bit 1 – PCIE1: Pin Change Interrupt Enable 1
When the PCIE1 bit is set and the I-bit in the Status Register (SREG) is set, pin change interrupt 1 is enabled. Any change on any enabled PCINT[14:8] pin will cause an interrupt. The corresponding interrupt of Pin Change Interrupt Request is executed from the PCI1 Interrupt Vector. PCINT[14:8] pins are enabled individually by the PCMSK1 Register.

Bit 0 – PCIE0: Pin Change Interrupt Enable 0
When the PCIE0 bit is set and the I-bit in the Status Register (SREG) is set, pin change interrupt 0 is enabled. Any change on any enabled PCINT[7:0] pin will cause an interrupt. The corresponding interrupt of Pin Change Interrupt Request is executed from the PCI0 Interrupt Vector. PCINT[7:0] pins are enabled individually by the PCMSK0 Register.

17.2.5. Pin Change Interrupt Flag Register

When addressing I/O Registers as data space using LD and ST instructions, the provided offset must be used. When using the I/O specific commands IN and OUT, the offset is reduced by 0x20, resulting in an I/O address offset within 0x00 - 0x3F.

Name: PCIFR
Offset: 0x3B
Reset: 0x00
Property: When addressing as I/O Register: address offset is 0x1B

Bit	7	6	5	4	3	2	1	0
						PCIF2	PCIF1	PCIF0
Access						R/W	R/W	R/W
Reset						0	0	0

Bit 2 – PCIF2: Pin Change Interrupt Flag 2
When a logic change on any PCINT[23:16] pin triggers an interrupt request, PCIF2 will be set. If the I-bit in SREG and the PCIE2 bit in PCICR are set, the MCU will jump to the corresponding Interrupt Vector. The flag is cleared when the interrupt routine is executed. Alternatively, the flag can be cleared by writing '1' to it.

Bit 1 – PCIF1: Pin Change Interrupt Flag 1
When a logic change on any PCINT[14:8] pin triggers an interrupt request, PCIF1 will be set. If the I-bit in SREG and the PCIE1 bit in PCICR are set, the MCU will jump to the corresponding Interrupt Vector. The flag is cleared when the interrupt routine is executed. Alternatively, the flag can be cleared by writing '1' to it.

Bit 0 – PCIF0: Pin Change Interrupt Flag 0
When a logic change on any PCINT[7:0] pin triggers an interrupt request, PCIF0 will be set. If the I-bit in SREG and the PCIE0 bit in PCICR are set, the MCU will jump to the corresponding Interrupt Vector. The flag is cleared when the interrupt routine is executed. Alternatively, the flag can be cleared by writing '1' to it.

17.2.6. Pin Change Mask Register 2

Name: PCMSK2
Offset: 0x6D
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	PCINT23	PCINT22	PCINT21	PCINT20	PCINT19	PCINT18	PCINT17	PCINT16
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 0, 1, 2, 3, 4, 5, 6, 7 – PCINT16, PCINT17, PCINT18, PCINT19, PCINT20, PCINT21, PCINT22, PCINT23: Pin Change Enable Mask
Each PCINT[23:16]-bit selects whether pin change interrupt is enabled on the corresponding I/O pin. If PCINT[23:16] is set and the PCIE2 bit in PCICR is set, pin change interrupt is enabled on the corresponding I/O pin. If PCINT[23:16] is cleared, pin change interrupt on the corresponding I/O pin is disabled.

17.2.7. Pin Change Mask Register 1

Name: PCMSK1
Offset: 0x6C
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
		PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT8
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 0, 1, 2, 3, 4, 5, 6 – PCINT8, PCINT9, PCINT10, PCINT11, PCINT12, PCINT13, PCINT14: Pin Change Enable Mask
Each PCINT[15:8]-bit selects whether pin change interrupt is enabled on the corresponding I/O pin. If PCINT[15:8] is set and the PCIE1 bit in PCICR is set, pin change interrupt is enabled on the corresponding I/O pin. If PCINT[15:8] is cleared, pin change interrupt on the corresponding I/O pin is disabled.

17.2.8. Pin Change Mask Register 0

Name: PCMSK0
Offset: 0x6B
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	PCINT7	PCINT6	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0
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 – PCINTn: Pin Change Enable Mask [n = 7:0]
Each PCINT[7:0] bit selects whether pin change interrupt is enabled on the corresponding I/O pin. If PCINT[7:0] is set and the PCIE0 bit in PCICR is set, pin change interrupt is enabled on the corresponding I/O pin. If PCINT[7:0] is cleared, pin change interrupt on the corresponding I/O pin is disabled.

32. Electrical Characteristics

32.1. Absolute Maximum Ratings

Table 32-1. 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	40.0mA
DC Current V_{CC} and GND Pins	200.0mA

Note: 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.

32.2. Common DC Characteristics

Table 32-2. Common DC characteristics T_A = -40°C to 105°C, V_{CC} = 1.8V to 5.5V (unless otherwise noted)

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
V_{IL}	Input Low Voltage, except XTAL1 and $\overline{\text{RESET}}$ pin	V_{CC} = 1.8V - 2.4V	-0.5		$0.2V_{CC}$(1)	V
		V_{CC} = 2.4V - 5.5V	-0.5		$0.3V_{CC}$(1)	
V_{IH}	Input High Voltage, except XTAL1 and $\overline{\text{RESET}}$ pins	V_{CC} = 1.8V - 2.4V	$0.7V_{CC}$(2)		V_{CC} + 0.5	V
		V_{CC} = 2.4V - 5.5V	$0.6V_{CC}$(2)		V_{CC} + 0.5	
V_{IL1}	Input Low Voltage, XTAL1 pin	V_{CC} = 1.8V - 5.5V	-0.5		$0.1V_{CC}$(1)	V
V_{IH1}	Input High Voltage, XTAL1 pin	V_{CC} = 1.8V - 2.4V	$0.8V_{CC}$(2)		V_{CC} + 0.5	V
		V_{CC} = 2.4V - 5.5V	$0.7V_{CC}$(2)		V_{CC} + 0.5	
V_{IL2}	Input Low Voltage, $\overline{\text{RESET}}$ pin	V_{CC} = 1.8V - 5.5V	-0.5		$0.1V_{CC}$(1)	V
V_{IH2}	Input High Voltage, $\overline{\text{RESET}}$ pin	V_{CC} = 1.8V - 5.5V	$0.9V_{CC}$(2)		V_{CC} + 0.5	V
V_{IL3}	Input Low Voltage, $\overline{\text{RESET}}$ pin as I/O	V_{CC} = 1.8V - 2.4V	-0.5		$0.2V_{CC}$(1)	V
		V_{CC} = 2.4V - 5.5V	-0.5		$0.3V_{CC}$(1)	

Symbol	Parameter	Condition		Min.	Typ.	Max.	Units
V_{IH3}	Input High Voltage, $\overline{\text{RESET}}$ pin as I/O	V_{CC} = 1.8V - 2.4V		$0.7V_{CC}$(2)		V_{CC} + 0.5	V
		V_{CC} = 2.4V - 5.5V		$0.6V_{CC}$(2)		V_{CC} + 0.5	
V_{OL}	Output Low Voltage(4) except RESET pin	I_{OL} = 20mA, V_{CC} = 5V	T_A=85°C			0.9	V
			T_A=105°C(5)			1.0	V
		I_{OL} = 10mA, V_{CC} = 3V	T_A=85°C			0.6	V
			T_A=105°C(5)			0.7	V
V_{OH}	Output High Voltage(3) except Reset pin	I_{OH} = -20mA, V_{CC} = 5V	T_A=85°C	4.2			V
			T_A=105°C(5)	4.1			V
		I_{OH} = -10mA, V_{CC} = 3V	T_A=85°C	2.3			V
			T_A=105°C(5)	2.1			V
I_{IL}	Input Leakage Current I/O Pin	V_{CC} = 5.5V, pin low (absolute value)				1	µA
I_{IH}	Input Leakage Current I/O Pin	V_{CC} = 5.5V, pin high (absolute value)				1	µA
R_{RST}	Reset Pull-up Resistor			30		60	kΩ
R_{PU}	I/O Pin Pull-up Resistor			20		50	kΩ
V_{ACIO}	Analog Comparator Input Offset Voltage	V_{CC} = 5V V_{in} = V_{CC}/2			<10	40	mV
I_{ACLK}	Analog Comparator Input Leakage Current	V_{CC} = 5V V_{in} = V_{CC}/2		-50		50	nA
t_{ACID}	Analog Comparator Propagation Delay	V_{CC} = 2.7V			750		ns
		V_{CC} = 4.0V			500		

Note:

1. “Max.” means the highest value where the pin is guaranteed to be read as low.
2. “Min.” means the lowest value where the pin is guaranteed to be read as high.
3. Although each I/O port can source more than the test conditions (20mA at V_{CC} = 5V, 10mA at V_{CC} = 3V) under steady state conditions (non-transient), the following must be observed:
   - 3.1. The sum of all I_{OH}, for ports C0 - C5, D0- D4, ADC7, $\overline{\text{RESET}}$ should not exceed 100mA.
   - 3.2. The sum of all I_{OH}, for ports B0 - B5, D5 - D7, ADC6, XTAL1, XTAL2 should not exceed 100mA.

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.

4. Although each I/O port can sink more than the test conditions (20mA at V_{CC} = 5V, 10mA at V_{CC} = 3V) under steady state conditions (non-transient), the following must be observed:
   4.1. The sum of all I_{OL}, for ports C0 - C5, ADC7, ADC6 should not exceed 100mA.
   4.2. The sum of all I_{OL}, for ports B0 - B5, D5 - D7, XTAL1, XTAL2 should not exceed 100mA.
   4.3. The sum of all I_{OL}, for ports D0 - D4, $\overline{\text{RESET}}$ should not exceed 100mA.

   If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test condition.
5. Only for ATmega328P

Related Links

Minimizing Power Consumption on page 65

32.2.1. ATmega328 DC Characteristics – Current Consumption

Table 32-3. DC characteristics - T_A = -40°C to 85°C, V_{CC} = 1.8V to 5.5V (unless otherwise noted)

Symbol	Parameter	Condition		Min.	Typ.(2)	Max.	Units
I_{CC}	Power Supply Current(1)	Active 1MHz, V_{CC} = 2V	T = 85°C		0.3	0.5	mA
		Active 4MHz, V_{CC} = 3V	T = 85°C		1.7	3.5	
		Active 8MHz, V_{CC} = 5V	T = 85°C		5.2	12	
		Idle 1MHz, V_{CC} = 2V	T = 85°C		0.04	0.5	
		Idle 4MHz, V_{CC} = 3V	T = 85°C		0.3	1.5	
		Idle 8MHz,V_{CC} = 5V	T = 85°C		1.2	5.5	
	Power-save mode(3)	32kHz TOSC enabled, V_{CC} = 1.8V	T = 85°C		0.8		µA
		32kHz TOSC enabled, V_{CC} = 3V	T = 85°C		0.9		
	Power-down mode(3)	WDT enabled, V_{CC} = 3V	T = 85°C		4.2	15	
		WDT disabled, V_{CC} = 3V	T = 85°C		0.1	2	

Note:

1. Values with *Minimizing Power Consumption* enabled (0xFF).
2. Typical values at 25°C. Maximum values are test limits in production.
3. The current consumption values include input leakage current.

32.2.2. ATmega328P DC Characteristics – Current Consumption

Table 32-4. ATmega328P DC characteristics - T_A = -40°C to 85/105°C, V_{CC} = 1.8V to 5.5V (unless otherwise noted)

Symbol	Parameter	Condition		Min.	Typ.(2)	Max.	Units
I_{CC}	Power Supply Current(1)	Active 1MHz, V_{CC} = 2V	T = 85°C		0.3	0.5	mA
			T = 105°C		0.3	0.5	
		Active 4MHz, V_{CC} = 3V	T = 85°C		1.7	2.5	
			T = 105°C		1.7	2.5	
		Active 8MHz, V_{CC} = 5V	T = 85°C		5.2	9.0	
			T = 105°C		5.2	9.0	
		Idle 1MHz, V_{CC} = 2V	T = 85°C		0.04	0.15	
			T = 105°C		0.04	0.15	
		Idle 4MHz, V_{CC} = 3V	T = 85°C		0.3	0.7	
			T = 105°C		0.3	0.7	
		Idle 8MHz,V_{CC} = 5V	T = 85°C		1.2	2.7	
			T = 105°C		1.2	2.7	
	Power-save mode(3)	32kHz TOSC enabled, V_{CC} = 1.8V	T = 85°C		0.8		µA
			T = 105°C		0.8		
		32kHz TOSC enabled, V_{CC} = 3V	T = 85°C		0.9		
			T = 105°C		0.9		
	Power-down mode(3)(4)	WDT enabled, V_{CC} = 3V	T = 85°C		4.2	8	
			T = 105°C		4.2	10	
		WDT disabled, V_{CC} = 3V	T = 85°C		0.1	2	
			T = 105°C		0.1	5	

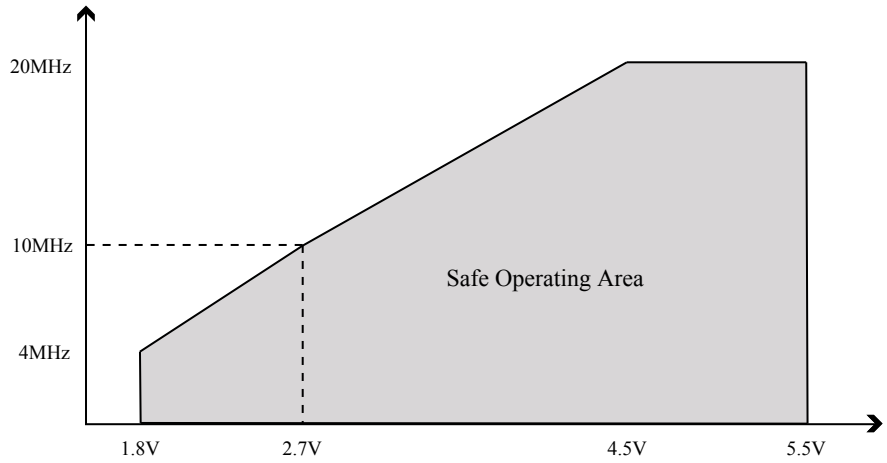
Note:

1. Values with *Minimizing Power Consumption* enabled (0xFF).
2. Typical values at 25°C. Maximum values are test limits in production.
3. The current consumption values include input leakage current.
4. No clock is applied to the pad during power-down mode.

32.3. Speed Grades

Maximum frequency is dependent on V_{CC}. As shown in Figure. Maximum Frequency vs. V_{CC}, the Maximum Frequency vs. V_{CC} curve is linear between 1.8V < V_{CC} < 2.7V and between 2.7V < V_{CC} < 4.5V.

Figure 32-1. Maximum Frequency vs. V_{CC}



32.4. Clock Characteristics

Related Links

Calibrated Internal RC Oscillator on page 54

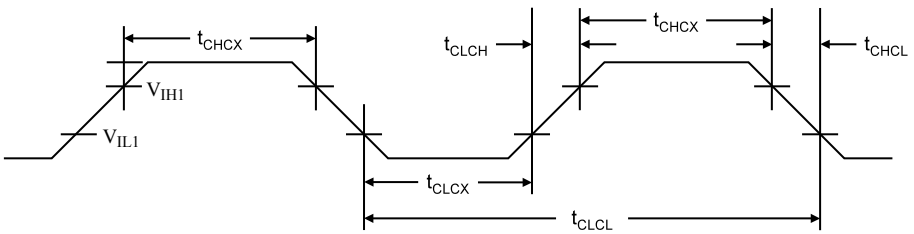
32.4.1. Calibrated Internal RC Oscillator Accuracy

Table 32-5. Calibration Accuracy of Internal RC Oscillator

	Frequency	V_{CC}	Temperature	Calibration Accuracy
Factory Calibration	8.0MHz	3.0V	25°C	±10%
User Calibration	7.3 - 8.1MHz	1.8V - 5.5V	-40°C to - 85°C	±1%

32.4.2. External Clock Drive Waveforms

Figure 32-2. External Clock Drive Waveforms



32.4.3. External Clock Drive

Table 32-6. External Clock Drive

Symbol	Parameter	V_{CC}= 1.8 - 5.5V		V_{CC}= 2.7 - 5.5V		V_{CC}= 4.5 - 5.5V		Units
		Min.	Max.	Min.	Max.	Min.	Max.	
$1/t_{CLCL}$	Oscillator Frequency	0	4	0	10	0	20	MHz
t_{CLCL}	Clock Period	250	-	100	-	50	-	ns
t_{CHCX}	High Time	100	-	40	-	20	-	ns

Symbol	Parameter	V_{CC}= 1.8 - 5.5V		V_{CC}= 2.7 - 5.5V		V_{CC}= 4.5 - 5.5V		Units
		Min.	Max.	Min.	Max.	Min.	Max.	
t_{CLCX}	Low Time	100	-	40	-	20	-	ns
t_{CLCH}	Rise Time	-	2.0	-	1.6	-	0.5	µs
t_{CHCL}	Fall Time	-	2.0	-	1.6	-	0.5	µs
Δt_{CLCL}	Change in period from one clock cycle to the next	-	2	-	2	-	2	%

32.5. System and Reset Characteristics

Table 32-7. Reset, Brown-out and Internal Voltage Characteristics(1)

Symbol	Parameter	Condition	Min.	Typ	Max	Units
V_{POT}	Power-on Reset Threshold Voltage (rising)		1.1	1.5	1.7	V
	Power-on Reset Threshold Voltage (falling)(2)		0.6	1.0	1.7	V
SR_{ON}	Power-on Slope Rate		0.01	-	10	V/ms
V_{RST}	$\overline{\text{RESET}}$ Pin Threshold Voltage		0.2 V_{CC}	-	0.9 V_{CC}	V
t_{RST}	Minimum pulse width on $\overline{\text{RESET}}$ Pin		-	-	2.5	µs
V_{HYST}	Brown-out Detector Hysteresis		-	50	-	mV
t_{BOD}	Min. Pulse Width on Brown-out Reset		-	2	-	µs
V_{BG}	Bandgap reference voltage	V_{CC}=2.7 T_A=25°C	1.0	1.1	1.2	V
t_{BG}	Bandgap reference start-up time	V_{CC}=2.7 T_A=25°C	-	40	70	µs
I_{BG}	Bandgap reference current consumption	V_{CC}=2.7 T_A=25°C	-	10	-	µA

Note:

1. Values are guidelines only.
2. The Power-on Reset will not work unless the supply voltage has been below VPOT (falling)

Table 32-8. BODLEVEL Fuse Coding(1)(2)

BODLEVEL [2:0] Fuses	Min. V_{BOT}	Typ. V_{BOT}	Max V_{BOT}	Units
111	BOD Disabled			
110	1.7	1.8	2.0	V
101	2.5	2.7	2.9	
100	4.1	4.3	4.5	

BODLEVEL [2:0] Fuses	Min. V_{BOT}	Typ. V_{BOT}	Max V_{BOT}	Units
011	Reserved			
010				
001				
000				

Note: V_{BOT} may be below nominal minimum operating voltage for some devices. For devices where this is the case, the device is tested down to $V_{CC} = V_{BOT}$ during the production test. This guarantees that a Brown-Out Reset will occur before V_{CC} drops to a voltage where correct operation of the microcontroller is no longer guaranteed. The test is performed using BODLEVEL = 110, 101 and 100.

Note: V_{BOT} tested at 25°C and 85°C in production

32.6. SPI Timing Characteristics

Table 32-9. SPI Timing Parameters

Description	Mode	Min.	Typ	Max	Units
SCK period	Master	-	See Table. Relationship Between SCK and the Oscillator Frequency in "SPCR – SPI Control Register"	-	ns
SCK high/low	Master	-	50% duty cycle	-	
Rise/Fall time	Master	-	3.6	-	
Setup	Master	-	10	-	
Hold	Master	-	10	-	
Out to SCK	Master	-	0.5 • t_{sck}	-	
SCK to out	Master	-	10	-	
SCK to out high	Master	-	10	-	
SS low to out	Slave	-	15	-	
SCK period	Slave	4 • t_{ck}	-	-	
SCK high/low(1)	Slave	2 • t_{ck}	-	-	
Rise/Fall time	Slave	-	-	1600	
Setup	Slave	10	-	-	
Hold	Slave	t_{ck}	-	-	
SCK to out	Slave	-	15	-	
SCK to SS high	Slave	20	-	-	
SS high to tri-state	Slave		10	-	
SS low to SCK	Slave	2 • t_{ck}	-	-	

Note: In SPI Programming mode the minimum SCK high/low period is:

- 2 • t_{CLCLCL} for f_{CK} < 12MHz

- $3 \cdot t_{CLCL}$ for $f_{CK} > 12MHz$

Figure 32-3. SPI Interface Timing Requirements (Master Mode)

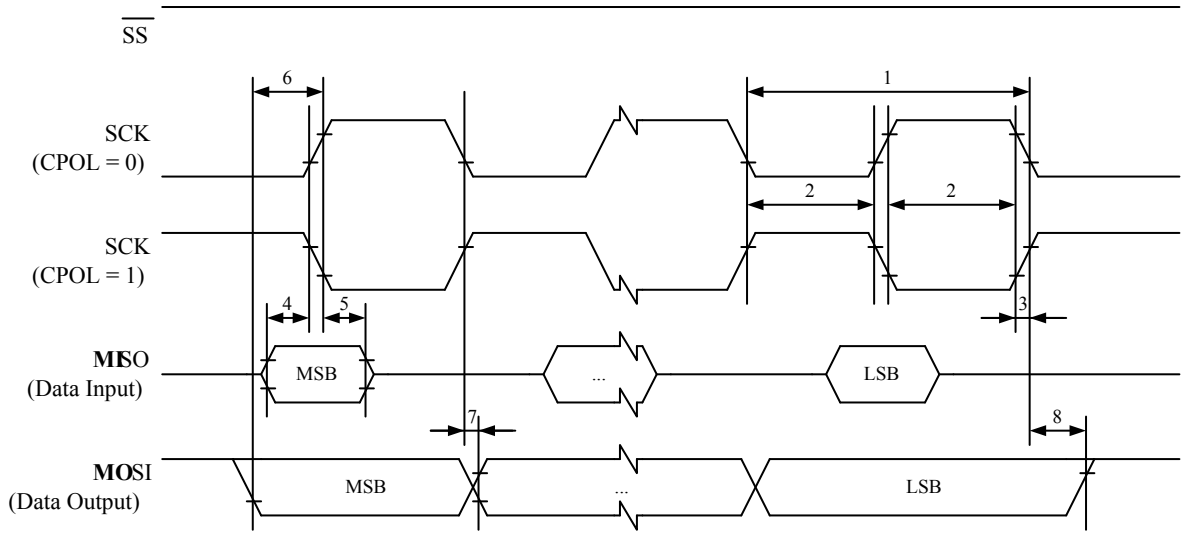
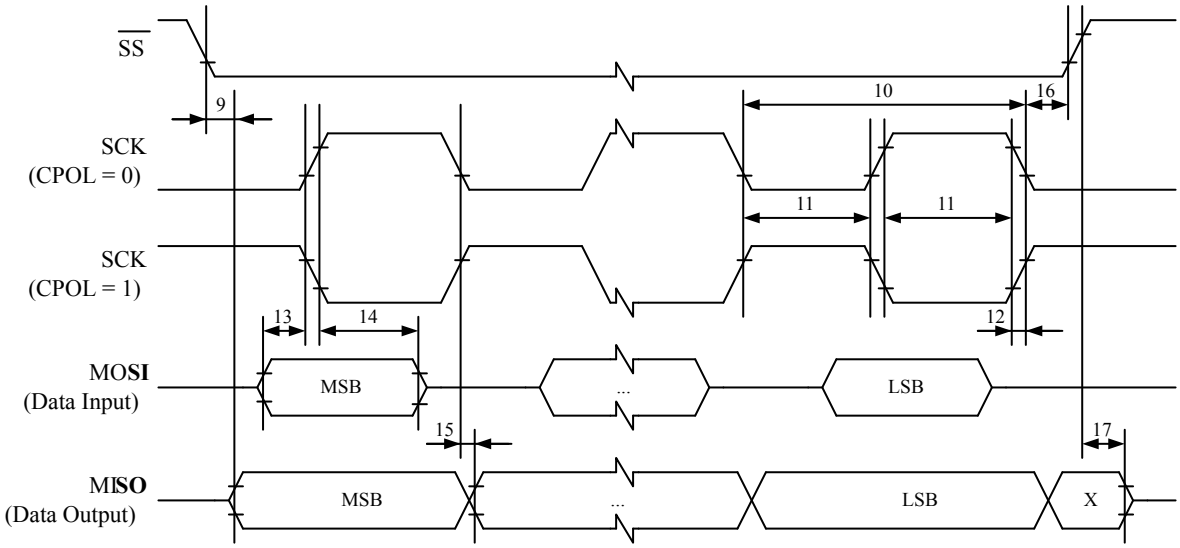


Figure 32-4. SPI Interface Timing Requirements (Slave Mode)



32.7. Two-wire Serial Interface Characteristics

Table in this section describes the requirements for devices connected to the 2-wire Serial Bus. The 2-wire Serial Interface meets or exceeds these requirements under the noted conditions.

Timing symbols refer to Figure 32-5.

Table 32-10. Two-wire Serial Bus Requirements

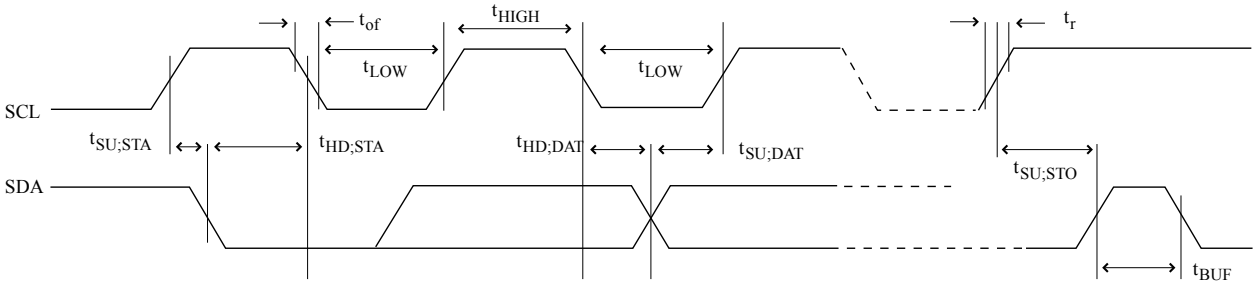
Symbol	Parameter	Condition	Min.	Max	Units
V_{IL}	Input Low-voltage		-0.5	0.3 V_{CC}	V
V_{IH}	Input High-voltage		0.7 V_{CC}	V_{CC} + 0.5	V

Symbol	Parameter	Condition	Min.	Max	Units
V_{hys}(1)	Hysteresis of Schmitt Trigger Inputs		0.05 V_{CC}(2)	–	V
V_{OL}(1)	Output Low-voltage	3mA sink current	0	0.4	V
t_r(1)	Rise Time for both SDA and SCL		20 + 0.1C_b(3)(2)	300	ns
t_{of}(1)	Output Fall Time from V_{IHmin} to V_{ILmax}	10pF < C_b < 400pF(3)	20 + 0.1C_b(3)(2)	250	ns
t_{SP}(1)	Spikes Suppressed by Input Filter		0	50(2)	ns
I_i	Input Current each I/O Pin	0.1V_{CC} < V_i < 0.9V_{CC}	-10	10	µA
C_i(1)	Capacitance for each I/O Pin		–	10	pF
f_{SCL}	SCL Clock Frequency	f_{CK}(4) > max(16f_{SCL}, 250kHz)(5)	0	400	kHz
Rp	Value of Pull-up resistor	f_{SCL} ≤ 100kHz	$\frac{V_{CC} - 0.4V}{3mA}$	$\frac{1000ns}{C_b}$	Ω
		f_{SCL} > 100kHz	$\frac{V_{CC} - 0.4V}{3mA}$	$\frac{300ns}{C_b}$	Ω
$t_{HD;STA}$	Hold Time (repeated) START Condition	f_{SCL} ≤ 100kHz	4.0	–	µs
		f_{SCL} > 100kHz	0.6	–	µs
t_{LOW}	Low Period of the SCL Clock	f_{SCL} ≤ 100kHz	4.7	–	µs
		f_{SCL} > 100kHz	1.3	–	µs
t_{HIGH}	High period of the SCL clock	f_{SCL} ≤ 100kHz	4.0	–	µs
		f_{SCL} > 100kHz	0.6	–	µs
$t_{SU;STA}$	Set-up time for a repeated START condition	f_{SCL} ≤ 100kHz	4.7	–	µs
		f_{SCL} > 100kHz	0.6	–	µs
$t_{HD;DAT}$	Data hold time	f_{SCL} ≤ 100kHz	0	3.45	µs
		f_{SCL} > 100kHz	0	0.9	µs
$t_{SU;DAT}$	Data setup time	f_{SCL} ≤ 100kHz	250	–	ns
		f_{SCL} > 100kHz	100	–	ns
$t_{SU;STO}$	Setup time for STOP condition	f_{SCL} ≤ 100kHz	4.0	–	µs
		f_{SCL} > 100kHz	0.6	–	µs
t_{BUF}	Bus free time between a STOP and START condition	f_{SCL} ≤ 100kHz	4.7	–	µs
		f_{SCL} > 100kHz	1.3	–	µs

Note:

1. This parameter is characterized and not 100% tested.
2. Required only for f_{SCL} > 100kHz.
3. C_b = capacitance of one bus line in pF.
4. f_{CK} = CPU clock frequency.
5. This requirement applies to all 2-wire Serial Interface operation. Other devices connected to the 2-wire Serial Bus need only obey the general f_{SCL} requirement.

Figure 32-5. Two-wire Serial Bus Timing



32.8. ADC Characteristics

Table 32-11. ADC Characteristics

Symbol	Parameter	Condition	Min.	Typ	Max	Units
	Resolution		-	10	-	Bits
	Absolute accuracy (Including INL, DNL, quantization error, gain and offset error)	V_{REF} = 4V, V_{CC} = 4V, ADC clock = 200kHz	-	2	-	LSB
		V_{REF} = 4V, V_{CC} = 4V, ADC clock = 1MHz	-	4	-	LSB
		V_{REF} = 4V, V_{CC} = 4V, ADC clock = 200kHz Noise Reduction Mode	-	2	-	LSB
		V_{REF} = 4V, V_{CC} = 4V, ADC clock = 1MHz Noise Reduction Mode	-	4	-	LSB
	Integral Non-Linearity (INL)	V_{REF} = 4V, V_{CC} = 4V, ADC clock = 200kHz	-	0.5	-	LSB
	Differential Non-Linearity (DNL)	V_{REF} = 4V, V_{CC} = 4V, ADC clock = 200kHz	-	0.25	-	LSB
	Gain Error	V_{REF} = 4V, V_{CC} = 4V, ADC clock = 200kHz	-	2	-	LSB
	Offset Error	V_{REF} = 4V, V_{CC} = 4V, ADC clock = 200kHz	-	2	-	LSB
	Conversion Time	Free Running Conversion	13	-	260	µs
	Clock Frequency		50	-	1000	kHz

Symbol	Parameter	Condition	Min.	Typ	Max	Units
AV_{CC}(1)	Analog Supply Voltage		V_{CC} - 0.3	-	V_{CC} + 0.3	V
VREF	Reference Voltage		1.0	-	AV_{CC}	V
V_{IN}	Input Voltage		GND	-	V_{REF}	V
	Input Bandwidth		-	38.5		kHz
V_{INT}	Internal Voltage Reference		1.0	1.1	1.2	V
R_{REF}	Reference Input Resistance		-	50	-	kΩ
R_{AIN}	Analog Input Resistance		-	100	-	MΩ

Note:

1. AV_{CC} absolute min./max: 1.8V/5.5V

32.9. Parallel Programming Characteristics

Table 32-12. Parallel Programming Characteristics, V_{CC} = 5V ± 10%

Symbol	Parameter	Min.	Max	Units
V_{PP}	Programming Enable Voltage	11.5	12.5	V
I_{PP}	Programming Enable Current	-	250	µA
t_{DVXH}	Data and Control Valid before XTAL1 High	67	-	ns
t_{XLXH}	XTAL1 Low to XTAL1 High	200	-	ns
t_{XHXL}	XTAL1 Pulse Width High	150	-	ns
t_{XLDX}	Data and Control Hold after XTAL1 Low	67	-	ns
t_{XLWL}	XTAL1 Low to $\overline{WR}$ Low	0	-	ns
t_{XLPH}	XTAL1 Low to PAGEL high	0	-	ns
t_{PLXH}	PAGEL low to XTAL1 high	150	-	ns
t_{BVPH}	BS1 Valid before PAGEL High	67	-	ns
t_{PHPL}	PAGEL Pulse Width High	150	-	ns
t_{PLBX}	BS1 Hold after PAGEL Low	67	-	ns
t_{WLBX}	BS2/1 Hold after RDY/$\overline{BSY}$ high	67	-	ns
t_{PLWL}	PAGEL Low to $\overline{WR}$ Low	67	-	ns
t_{BVWL}	BS1 Valid to $\overline{WR}$ Low	67	-	ns
t_{WLWH}	$\overline{WR}$ Pulse Width Low	150	-	ns
t_{WLRL}	$\overline{WR}$ Low to RDY/$\overline{BSY}$ Low	0	1	µs
t_{WLRH}	$\overline{WR}$ Low to RDY/$\overline{BSY}$ High(1)	3.2	3.4	ms
t_{WLRH_CE}	$\overline{WR}$ Low to RDY/$\overline{BSY}$ High for Chip Erase(2)	9.8	10.5	ms

34. Typical Characteristics (T_A = -40°C to 105°C)

The following charts show typical behavior. These figures are not tested during manufacturing. All current consumption measurements are performed with all I/O pins configured as inputs and with internal pull-ups enabled. A sine wave generator with rail-to-rail output is used as clock source.

The power consumption in Power-down mode is independent of clock selection.

The current consumption is a function of several factors such as: operating voltage, operating frequency, loading of I/O pins, switching rate of I/O pins, code executed and ambient temperature. The dominating factors are operating voltage and frequency.

The current drawn from capacitive loaded pins may be estimated (for one pin) as C_L*V_{CC}*f where C_L = load capacitance, V_{CC} = operating voltage and f = average switching frequency of I/O pin.

The parts are characterized at frequencies higher than test limits. Parts are not guaranteed to function properly at frequencies higher than the ordering code indicates.

The difference between current consumption in Power-down mode with Watchdog Timer enabled and Power-down mode with Watchdog Timer disabled represents the differential current drawn by the Watchdog Timer.

34.1. ATmega328P Typical Characteristics

34.1.1. Active Supply Current

Figure 34-1. ATmega328P: Active Supply Current vs. Low Frequency (0.1MHz - 1.0MHz)

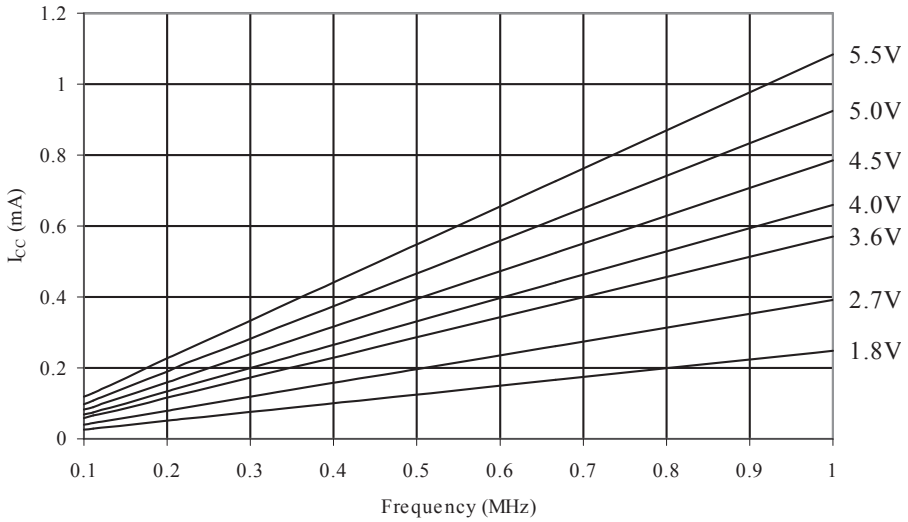


Figure 34-2. ATmega328P: Active Supply Current vs. Frequency (1MHz - 20MHz)

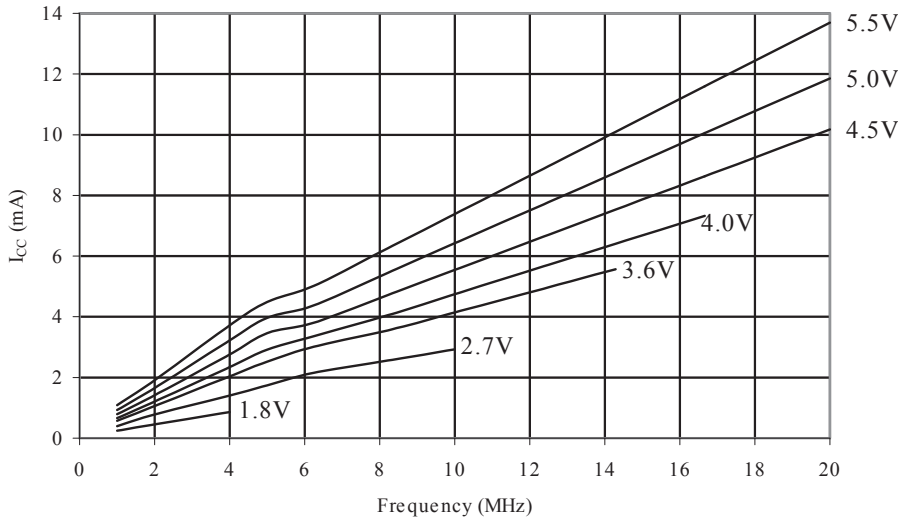


Figure 34-3. ATmega328P: Active Supply Current vs. V_{CC} (Internal RC Oscillator, 128kHz)

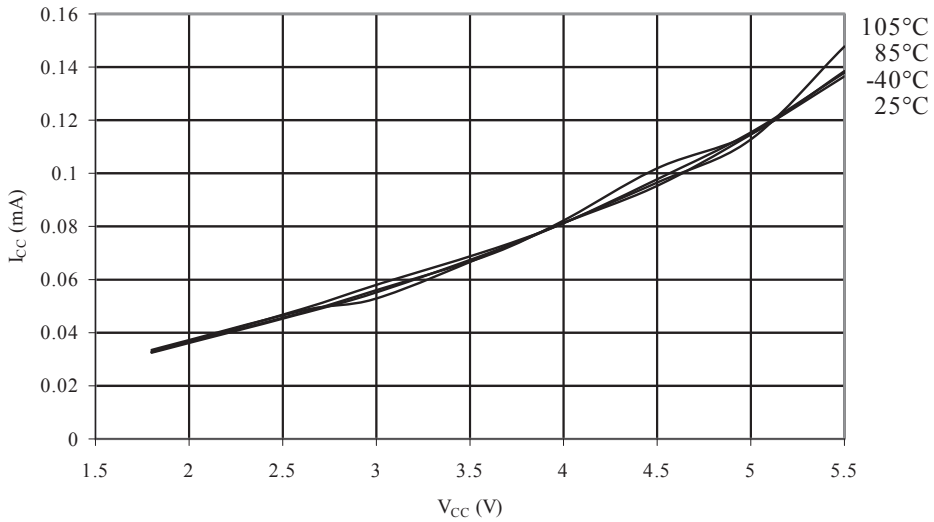


Figure 34-4. ATmega328P: Active Supply Current vs. V_{CC} (Internal RC Oscillator, 1MHz)

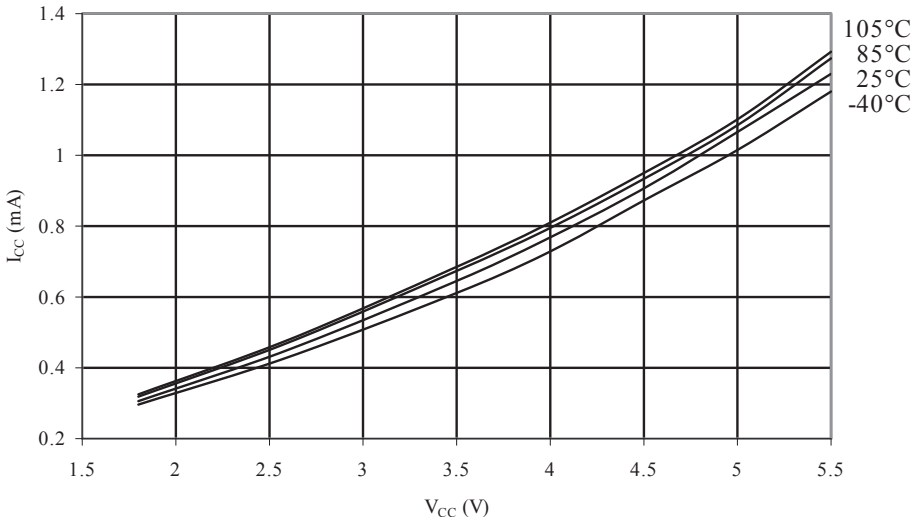
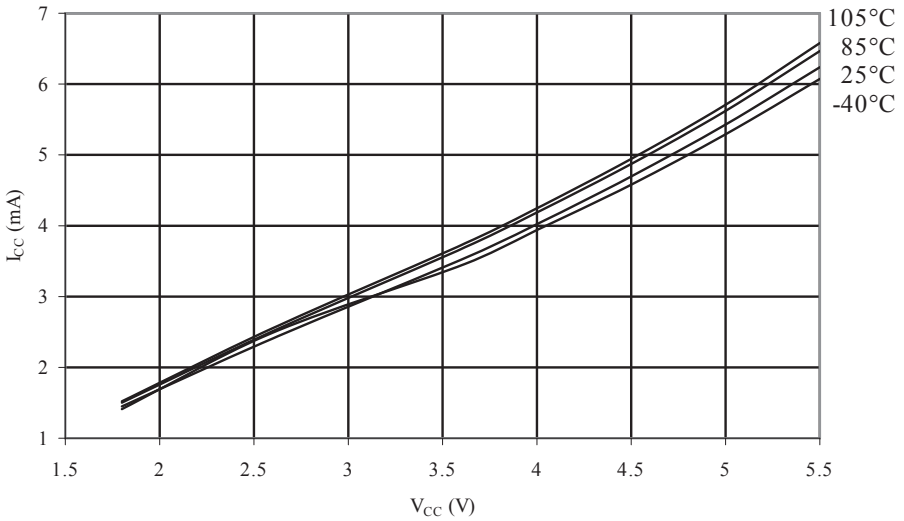


Figure 34-5. ATmega328P: Active Supply Current vs. V_{CC} (Internal RC Oscillator, 8MHz)



34.1.2. Idle Supply Current

Figure 34-6. ATmega328P: Idle Supply Current vs. Low Frequency (0.1MHz - 1.0MHz)

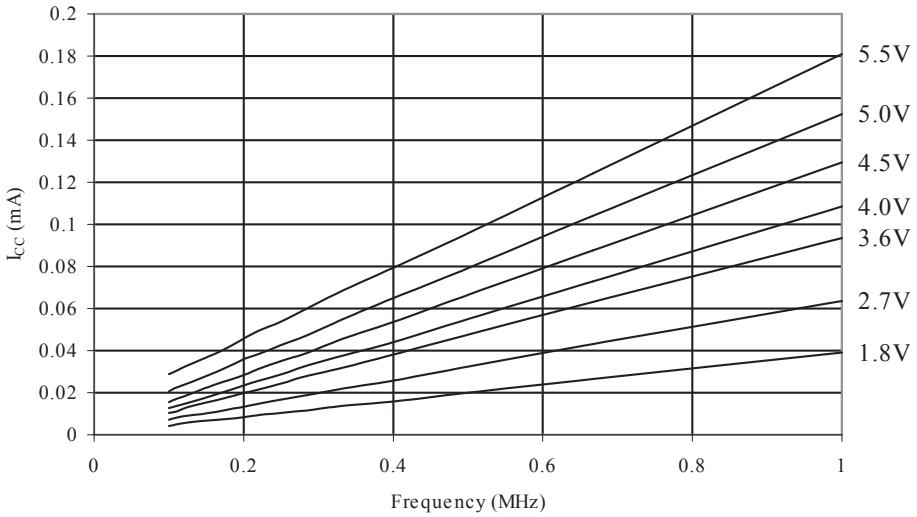


Figure 34-7. ATmega328P: Idle Supply Current vs. Frequency (1MHz - 20MHz)

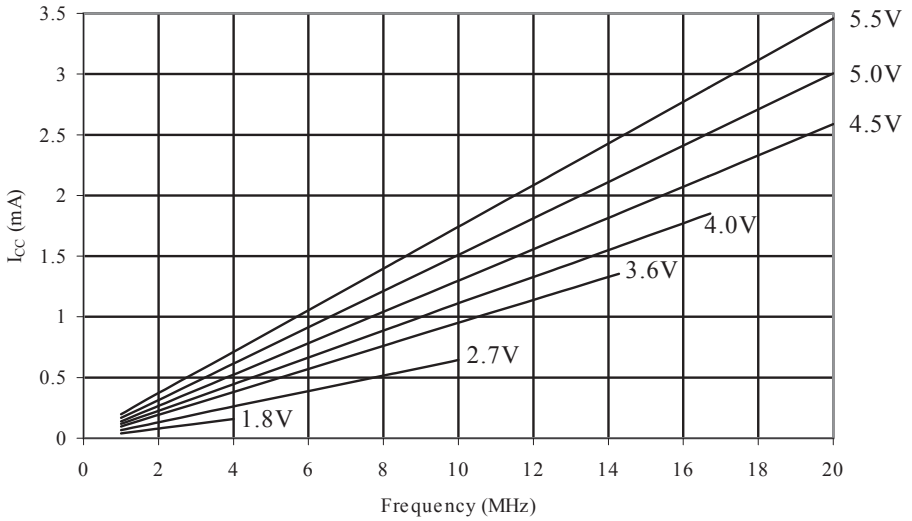


Figure 34-8. ATmega328P: Idle Supply Current vs. V_{CC} (Internal RC Oscillator, 128kHz)

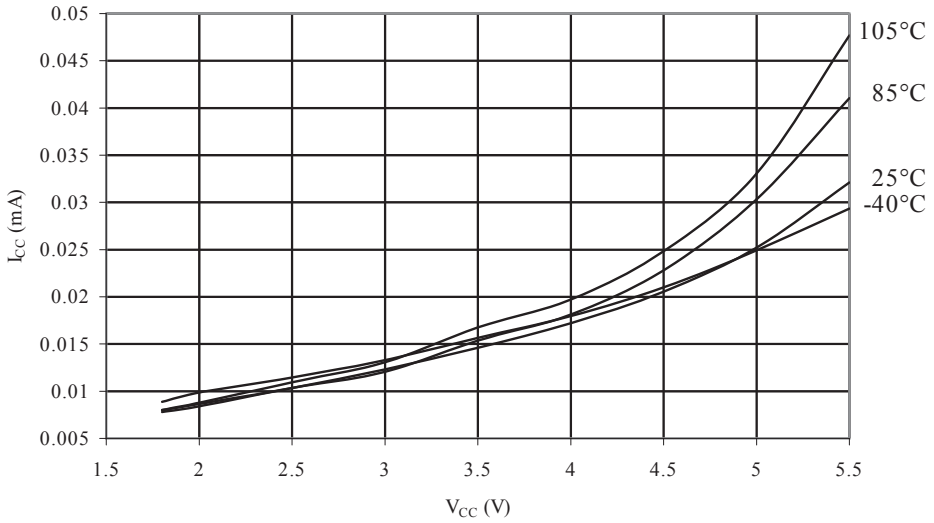


Figure 34-9. ATmega328P: Idle Supply Current vs. V_{CC} (Internal RC Oscillator, 1MHz)

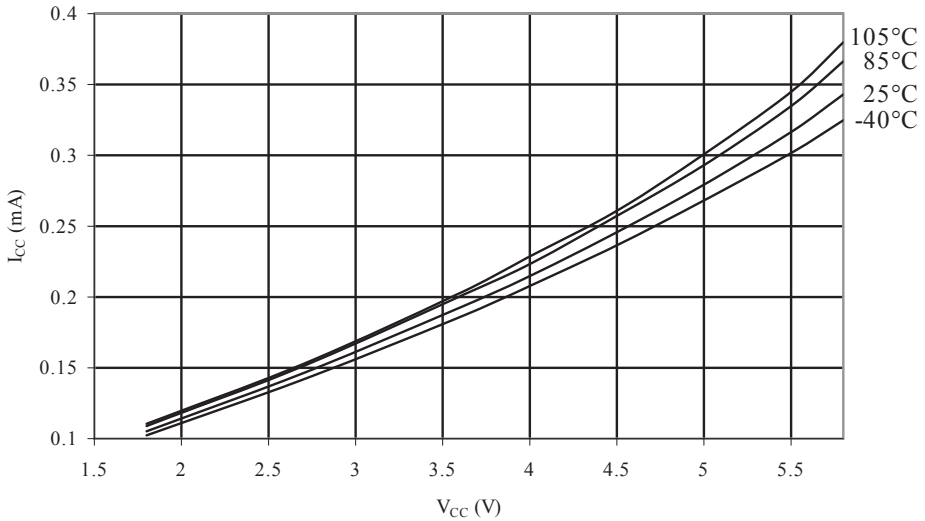
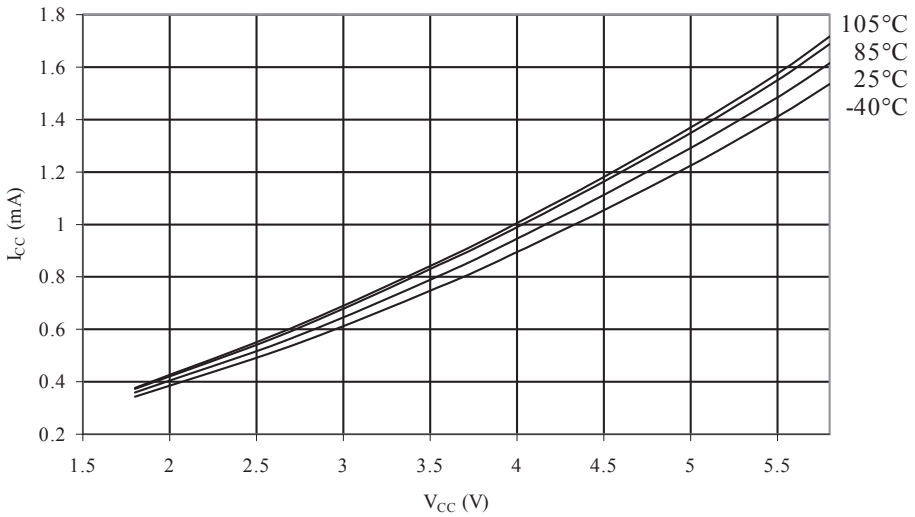


Figure 34-10. ATmega328P: Idle Supply Current vs. V_{CC} (Internal RC Oscillator, 8MHz)



34.1.3. Supply Current of IO Modules

The tables and formulas below can be used to calculate the additional current consumption for the different I/O modules in Active and Idle mode. The enabling or disabling of the I/O modules are controlled by the Power Reduction Register. See “Power Reduction Register” for details.

Table 34-1. ATmega328P: Additional Current Consumption for the different I/O modules (absolute values)

PRR bit	Typical numbers (µA)		
	V_{CC} = 2V, F = 1MHz	V_{CC} = 3V, F = 4MHz	V_{CC} = 5V, F = 8MHz
PRUSART0	3.20	22.17	100.25
PRTWI	7.34	46.55	199.25
PRTIM2	7.34	50.79	224.25
PRTIM1	6.19	41.25	176.25
PRTIM0	1.89	14.28	61.13
PRSPI	6.94	43.84	186.50
PRADC	8.66	61.80	295.38

Table 34-2. ATmega328P: Additional Current Consumption (percentage) in Active and Idle mode

PRR bit	Additional Current consumption compared to Active with external clock (see Figure 34-1 and Figure 34-2)	Additional Current consumption compared to Idle with external clock (see Figure 34-6 and Figure 34-7)
PRUSART0	1.4%	7.8%
PRTWI	3.0%	16.6%
PRTIM2	3.3%	17.8%
PRTIM1	2.7%	14.5%
PRTIM0	0.9%	4.8%

PRR bit	Additional Current consumption compared to Active with external clock (see Figure 34-1 and Figure 34-2)	Additional Current consumption compared to Idle with external clock (see Figure 34-6 and Figure 34-7)
PRSPI	2.9%	15.7%
PRADC	4.1%	22.1%

It is possible to calculate the typical current consumption based on the numbers from the above table for other V_{CC} and frequency settings.

Related Links

PRR on page 71

34.1.3.1. Example

Calculate the expected current consumption in idle mode with TIMER1, ADC, and SPI enabled at V_{CC} = 2.0V and F = 1MHz. From Table *Additional Current Consumption (percentage) in Active and Idle mode* in the previous section, third column, we see that we need to add 14.5% for the TIMER1, 22.1% for the ADC, and 15.7% for the SPI module. Reading from Figure *Idle Supply Current vs. Low Frequency (0.1-1.0MHz)*, we find that the idle current consumption is ~0.045mA at V_{CC} = 2.0V and F = 1MHz. The total current consumption in idle mode with TIMER1, ADC, and SPI enabled, gives:

$$I_{CCtotal} \simeq 0.045\ \text{mA} \cdot (1 + 0.145 + 0.221 + 0.157) \simeq 0.069\ \text{mA}$$

34.1.4. Power-down Supply Current

Figure 34-11. ATmega328P: Power-Down Supply Current vs. V_{CC} (Watchdog Timer Disabled)

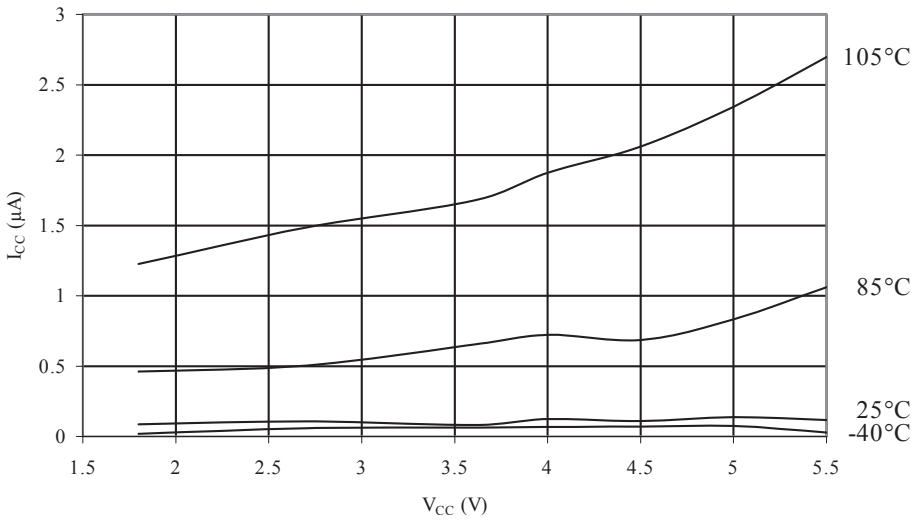
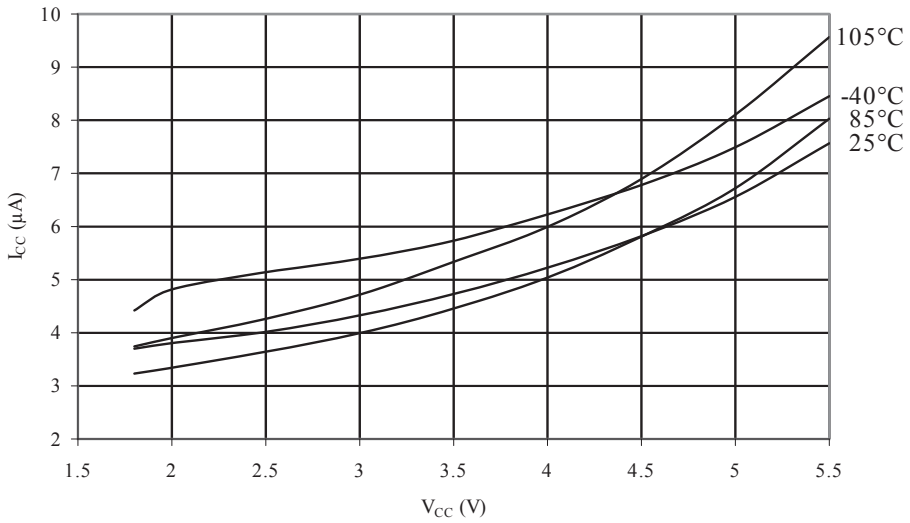
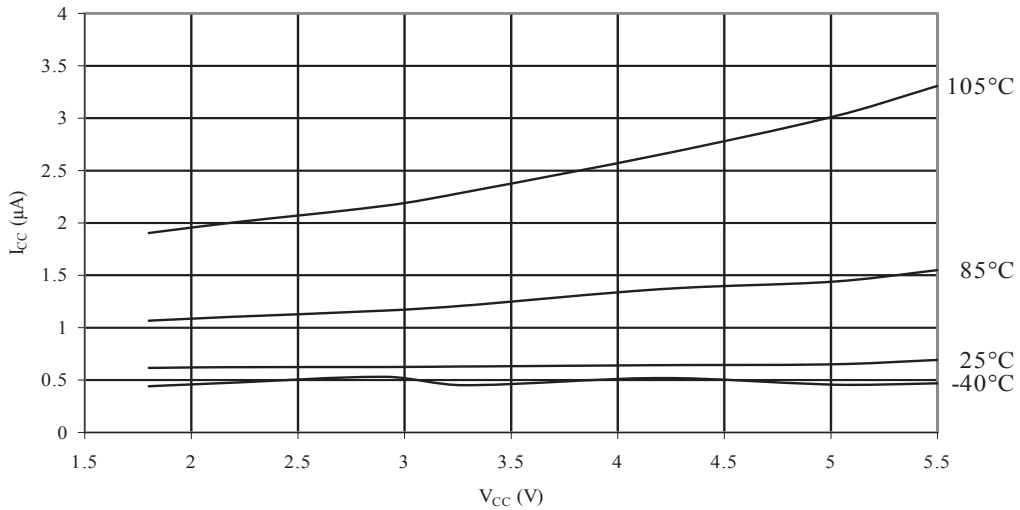


Figure 34-12. ATmega328P: Power-Down Supply Current vs. V_{CC} (Watchdog Timer Enabled)



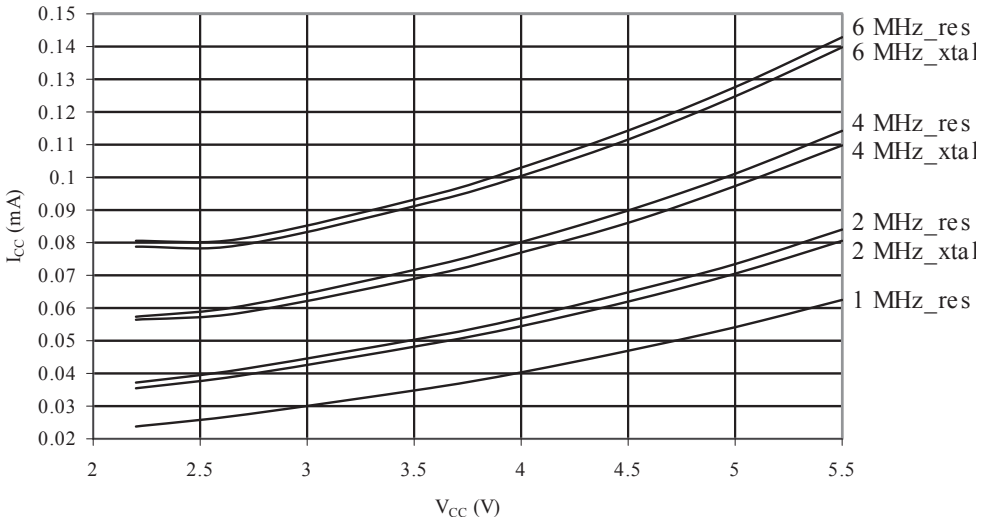
34.1.5. Power-save Supply Current

Figure 34-13. ATmega328P: Power-Save Supply Current vs. V_{CC} (Watchdog Timer Disabled and 32kHz Crystal Oscillator Running)



34.1.6. Standby Supply Current

Figure 34-14. ATmega328P: Standby Supply Current vs. V_{CC} (Watchdog Timer Disabled)



34.1.7. Pin Pull-Up

Figure 34-15. ATmega328P: I/O Pin Pull-up Resistor Current vs. Input Voltage (V_{CC} = 1.8V)

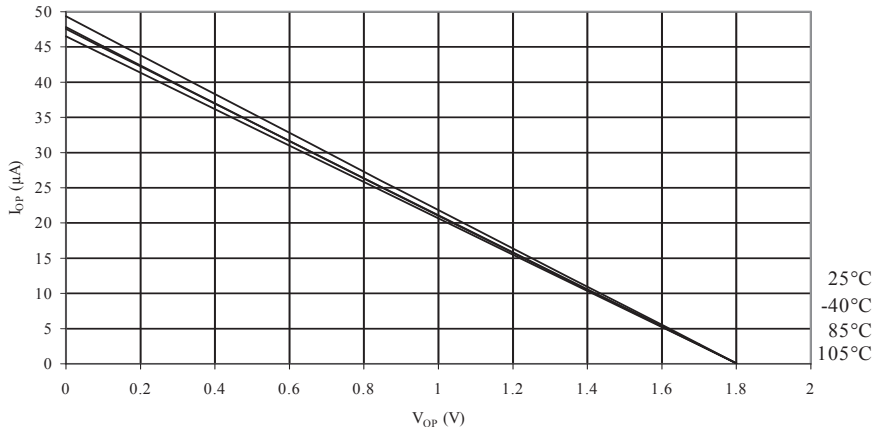


Figure 34-16. ATmega328P: I/O Pin Pull-up Resistor Current vs. Input Voltage (V_{CC} = 2.7V)

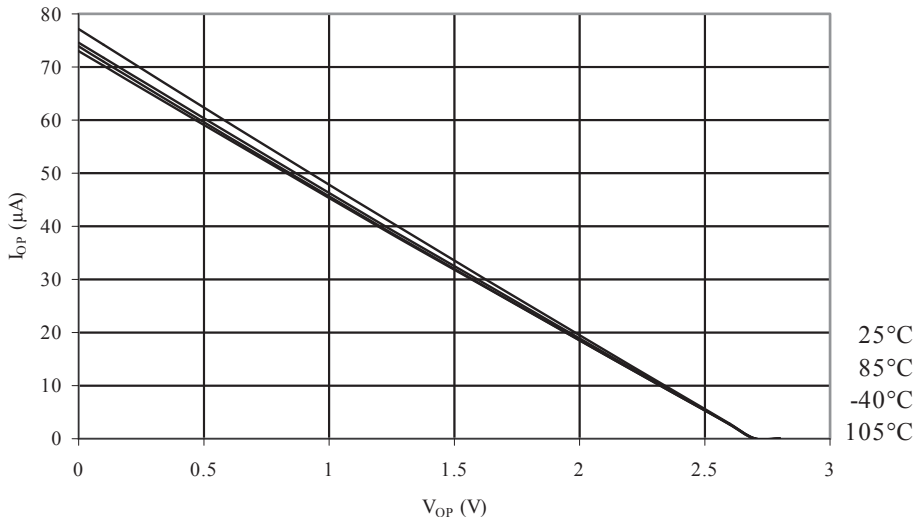


Figure 34-17. ATmega328P: I/O Pin Pull-up Resistor Current vs. Input Voltage (V_{CC} = 5V)

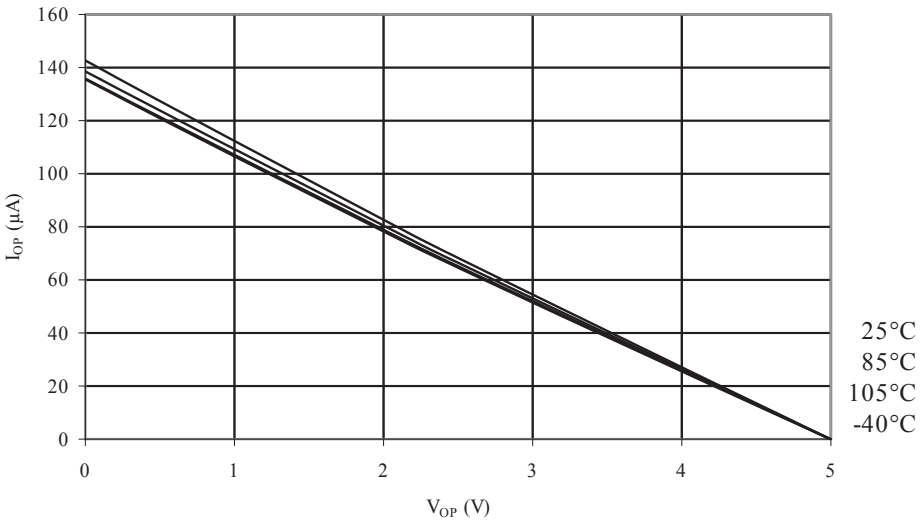


Figure 34-18. ATmega328P: Reset Pull-up Resistor Current vs. Reset Pin Voltage (V_{CC} = 1.8V)

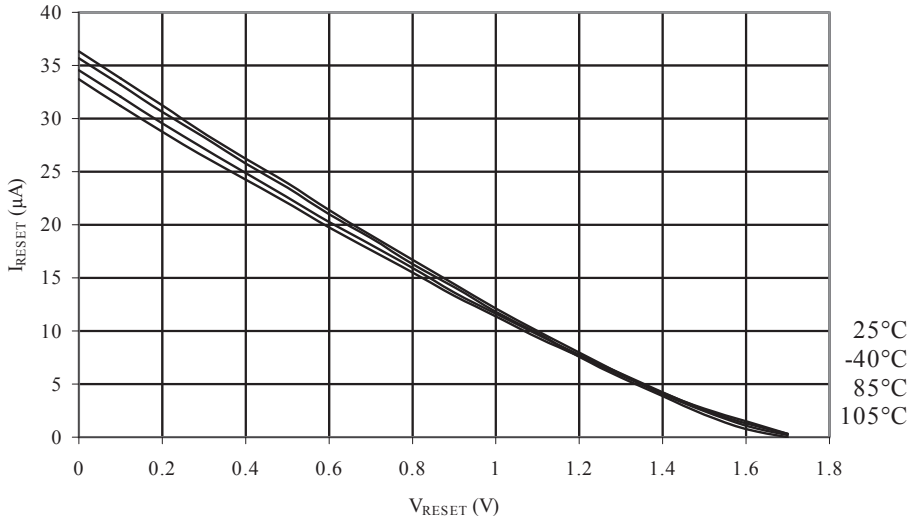


Figure 34-19. ATmega328P: Reset Pull-up Resistor Current vs. Reset Pin Voltage (V_{CC} = 2.7V)

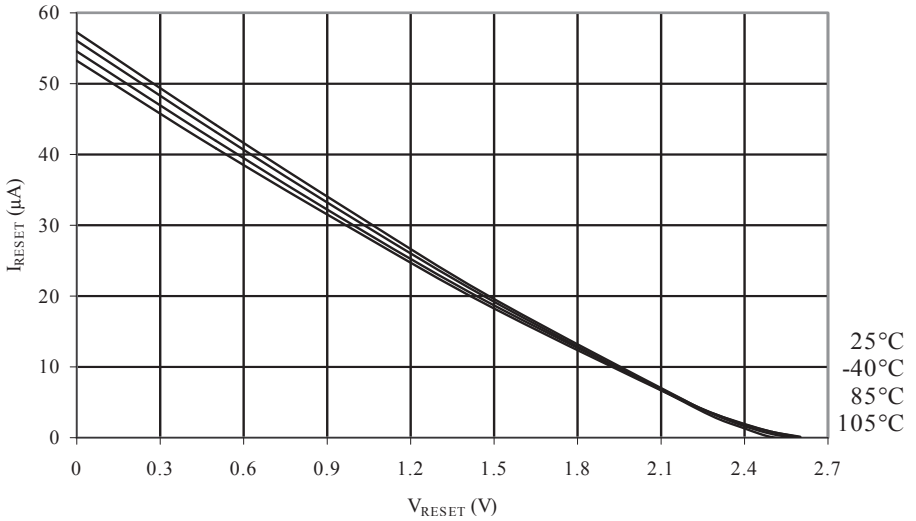
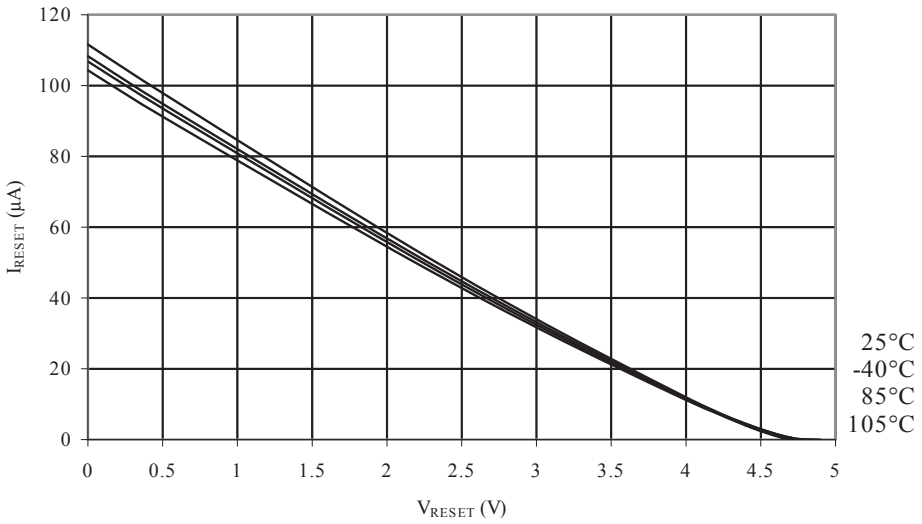


Figure 34-20. ATmega328P: Reset Pull-up Resistor Current vs. Reset Pin Voltage (V_{CC} = 5V)



34.1.8. Pin Driver Strength

Figure 34-21. ATmega328P: I/O Pin Output Voltage vs. Sink Current (V_{CC} = 3V)

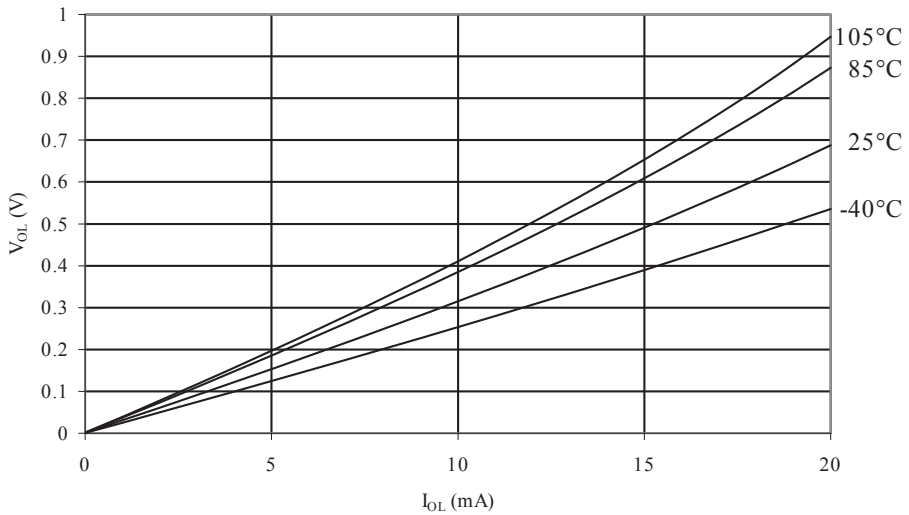


Figure 34-22. ATmega328P: I/O Pin Output Voltage vs. Sink Current (V_{CC} = 5V)

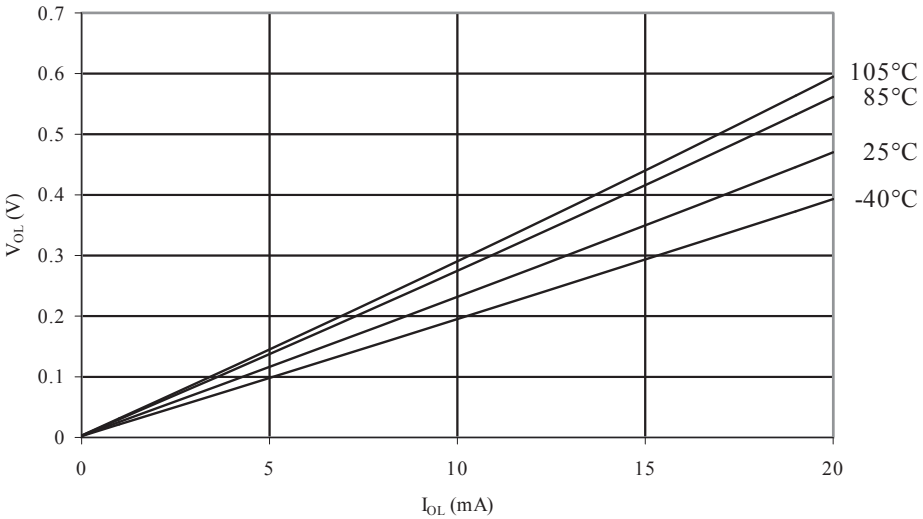


Figure 34-23. ATmega328P: I/O Pin Output Voltage vs. Source Current (Vcc = 3V)

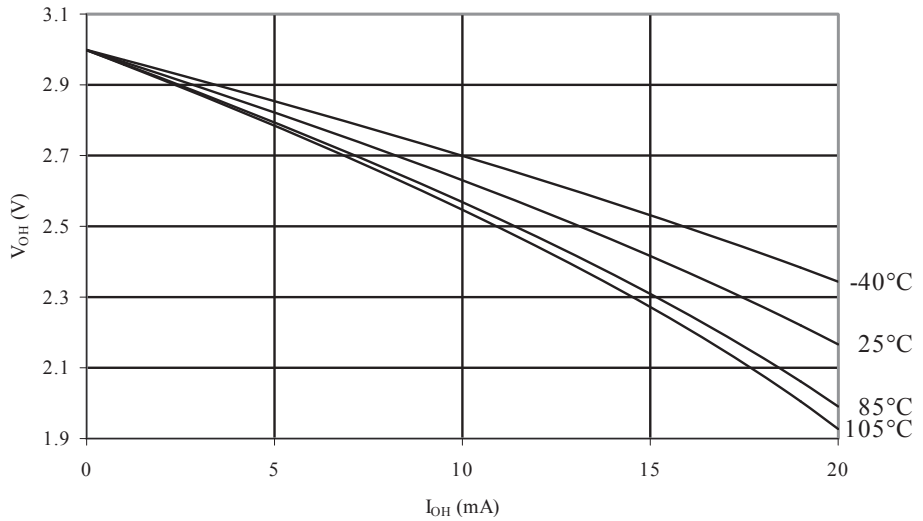
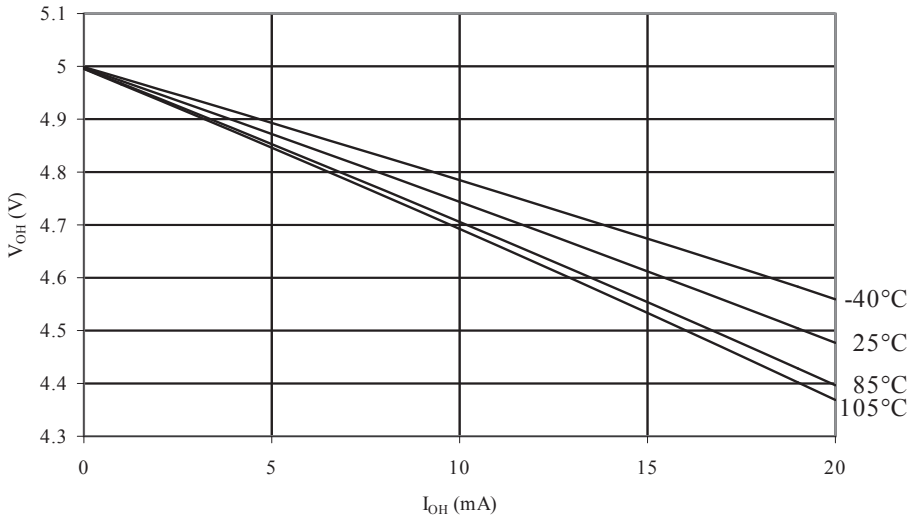


Figure 34-24. ATmega328P: I/O Pin Output Voltage vs. Source Current (V_{CC} = 5V)



34.1.9. Pin Threshold and Hysteresis

Figure 34-25. ATmega328P: I/O Pin Input Threshold Voltage vs. V_{CC} (V_{IH}, I/O Pin read as '1')

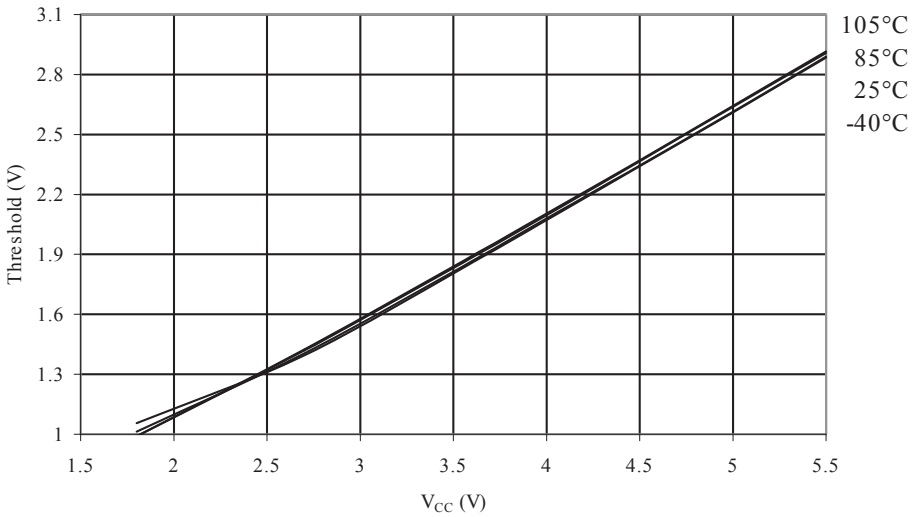


Figure 34-26. ATmega328P: I/O Pin Input Threshold Voltage vs. V_{CC} (V_{IL}, I/O Pin read as '0'

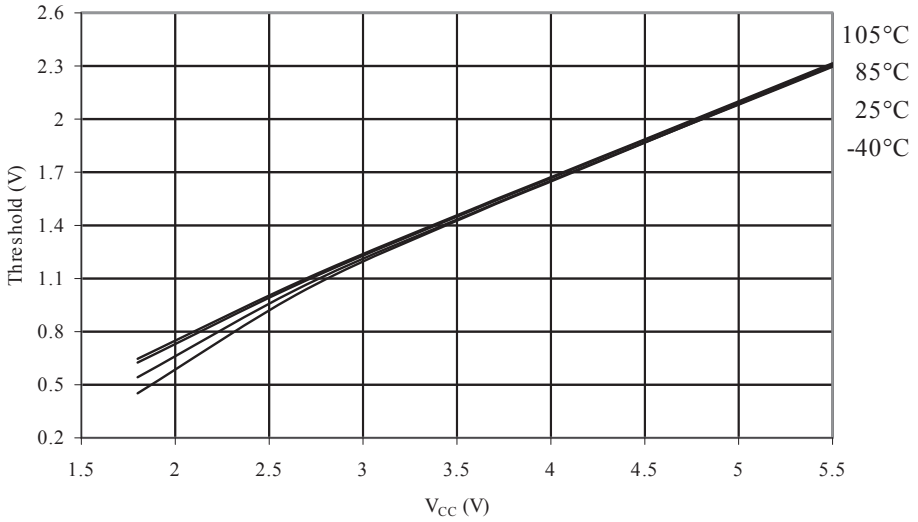


Figure 34-27. ATmega328P: I/O Pin Input Hysteresis vs. V_{CC}

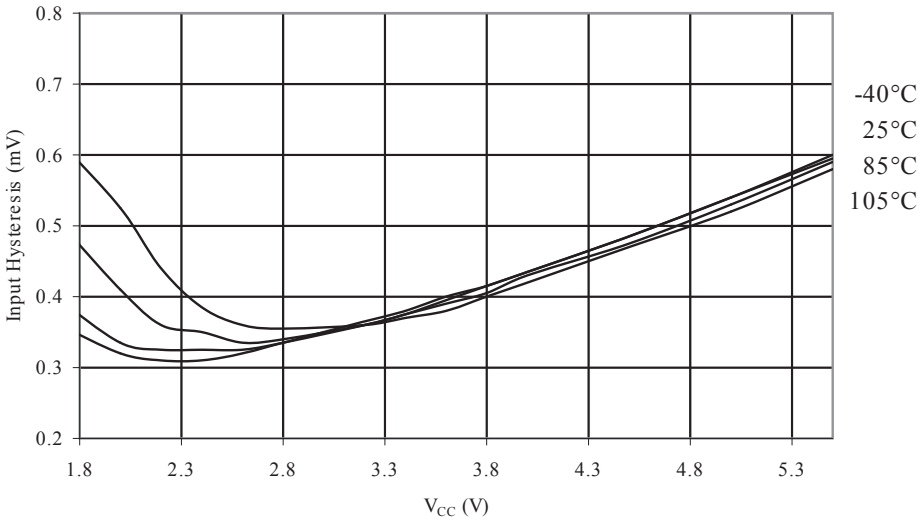


Figure 34-28. ATmega328P: Reset Input Threshold Voltage vs. V_{CC} (V_{IH}, I/O Pin read as '1')

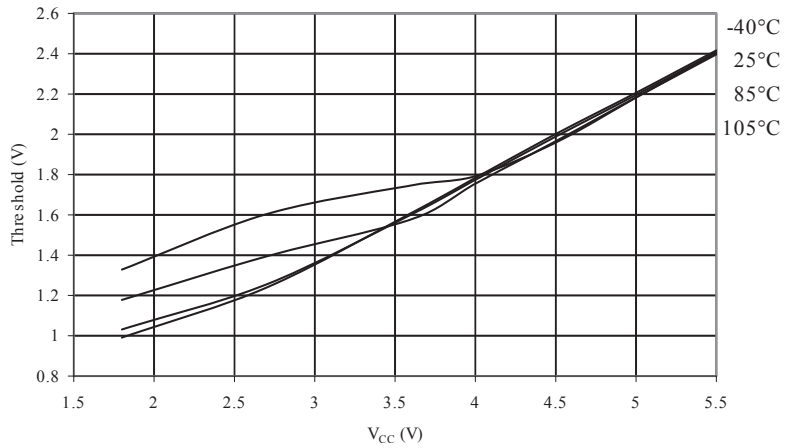


Figure 34-29. ATmega328P: Reset Input Threshold Voltage vs. V_{CC} (V_{IL}, I/O Pin read as '0')

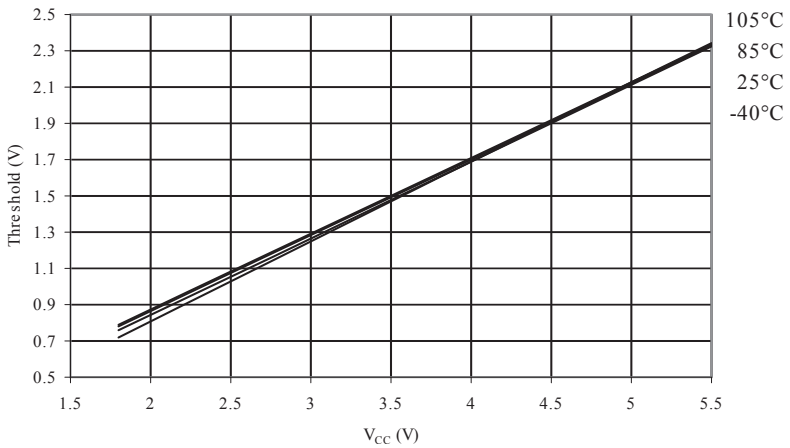
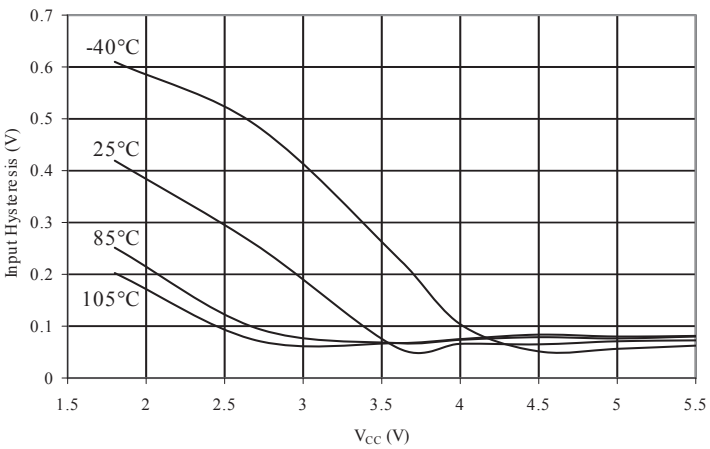


Figure 34-30. ATmega328P: Reset Pin Input Hysteresis vs. V_{CC}



34.1.10. BOD Threshold

Figure 34-31. ATmega328P: BOD Thresholds vs. Temperature (BODLEVEL is 1.8V)

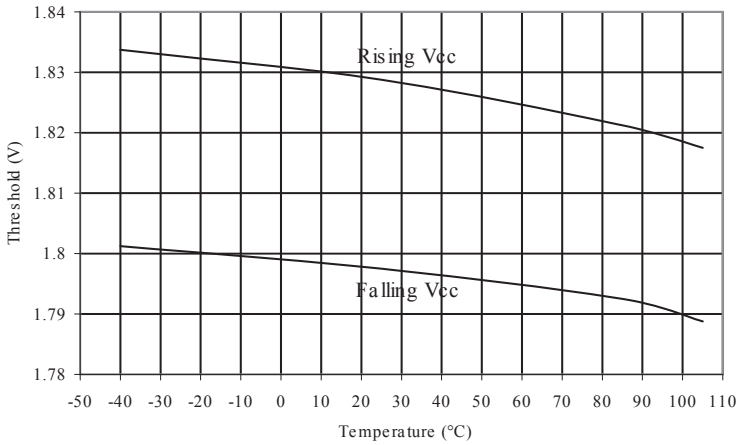


Figure 34-32. ATmega328P: BOD Thresholds vs. Temperature (BODLEVEL is 2.7V)

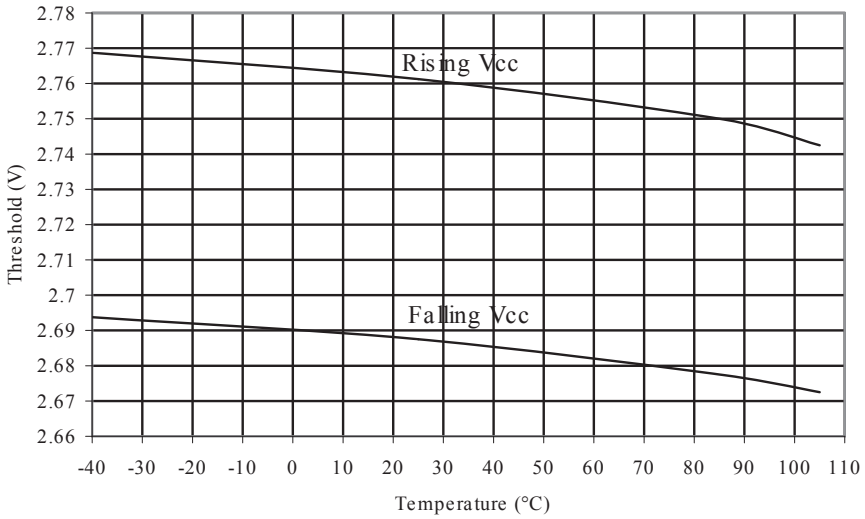


Figure 34-33. ATmega328P: BOD Thresholds vs. Temperature (BODLEVEL is 4.3V)

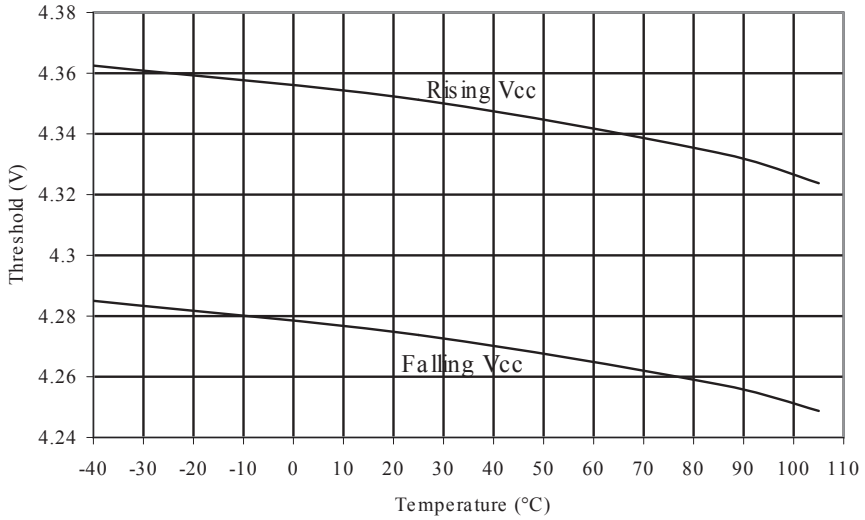
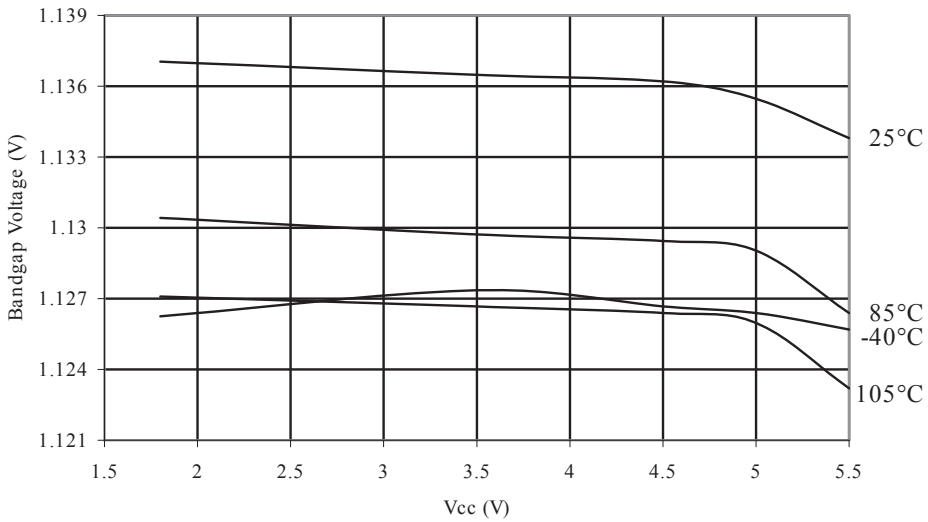


Figure 34-34. ATmega328P: Calibrated Bandgap Voltage vs. V_{CC}



34.1.11. Internal Oscillator Speed

Figure 34-35. ATmega328P: Watchdog Oscillator Frequency vs. Temperature

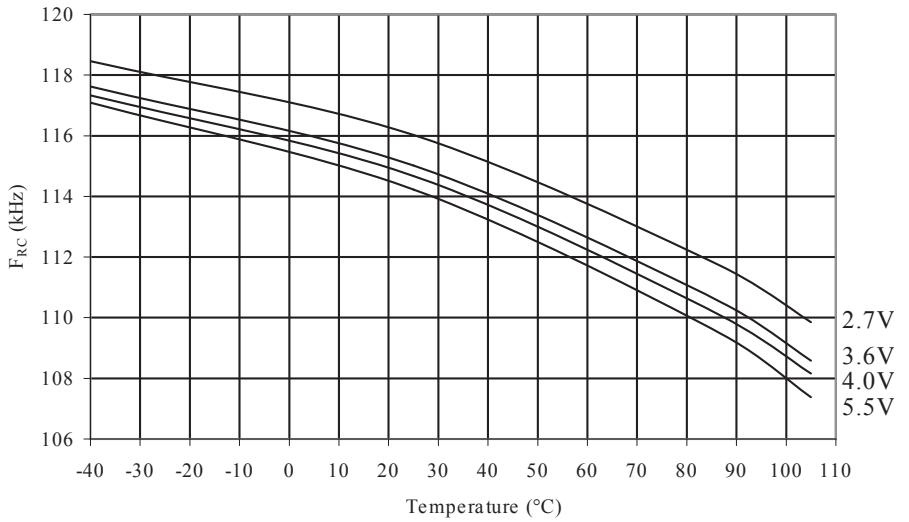


Figure 34-36. ATmega328P: Watchdog Oscillator Frequency vs. V_{CC}

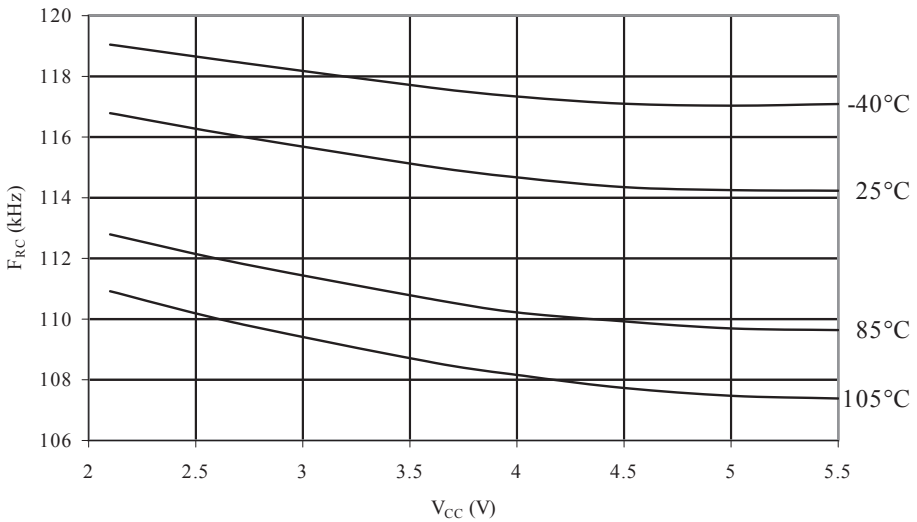


Figure 34-37. ATmega328P: Calibrated 8 MHz RC Oscillator Frequency vs. V_{CC}

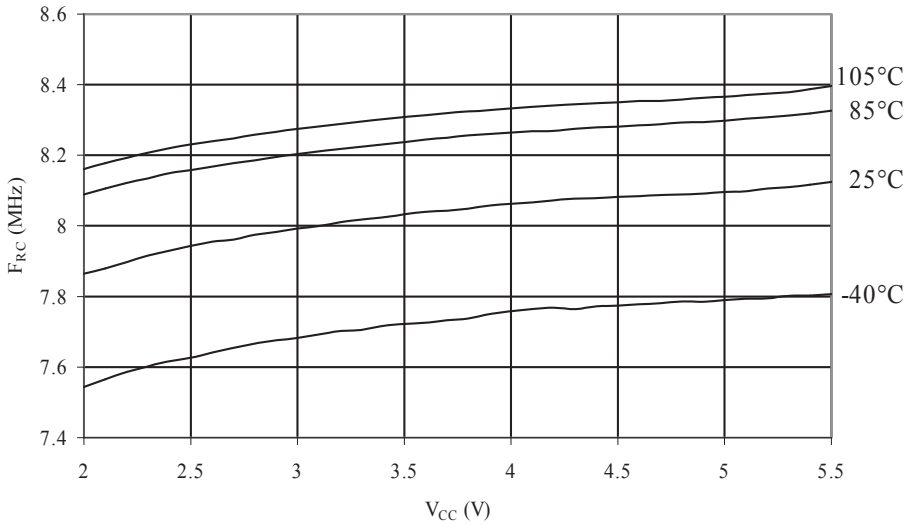


Figure 34-38. ATmega328P: Calibrated 8MHz RC Oscillator Frequency vs. Temperature

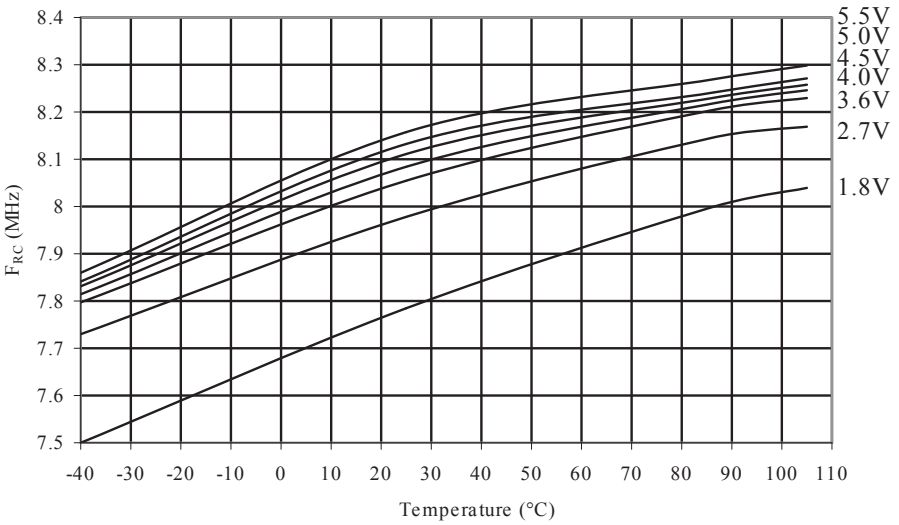
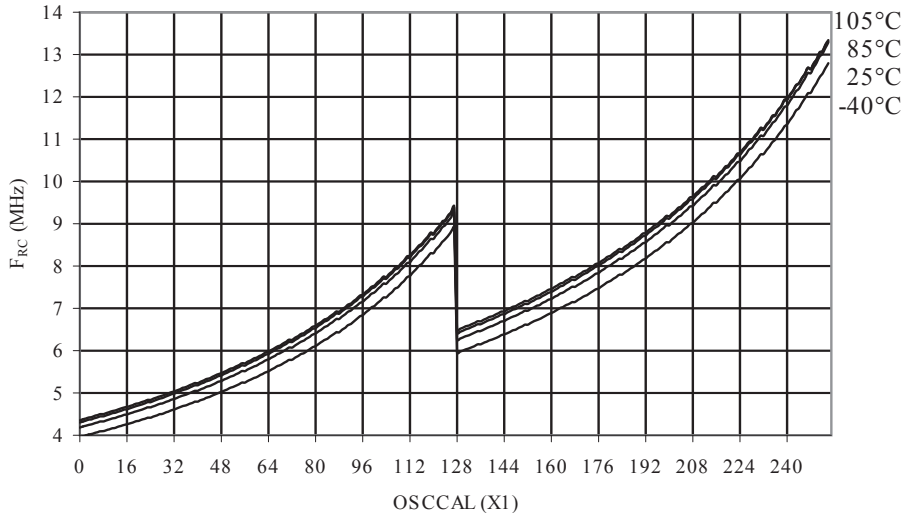


Figure 34-39. ATmega328P: Calibrated 8MHz RC Oscillator Frequency vs. OSCCAL Value



34.1.12. Current Consumption of Peripheral Units

Figure 34-40. ATmega328P: ADC Current vs. V_{CC} (AREF = AV_{CC})

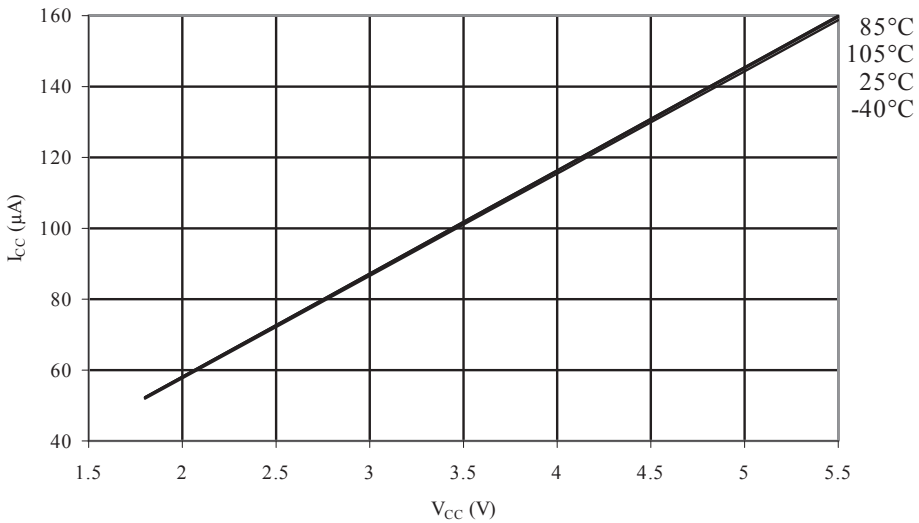


Figure 34-41. ATmega328P: Analog Comparator Current vs. V_{CC}

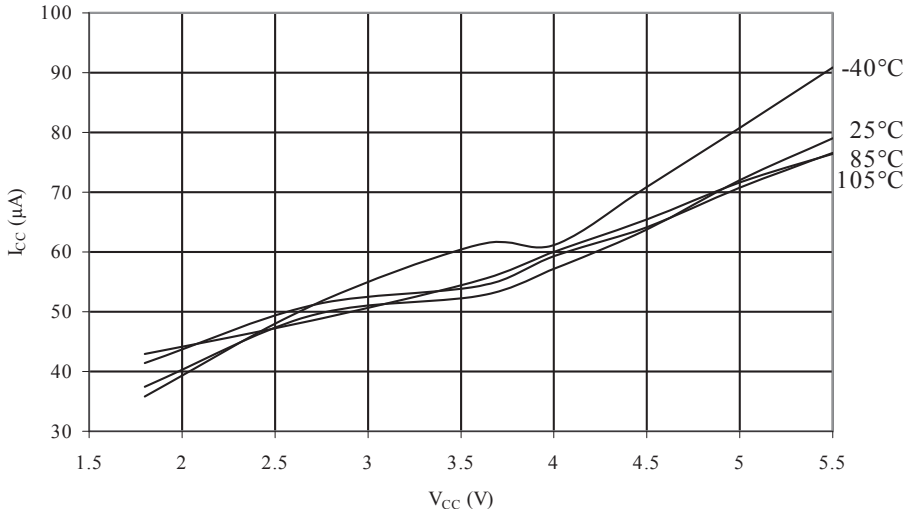


Figure 34-42. ATmega328P: AREF External Reference Current vs. V_{CC}

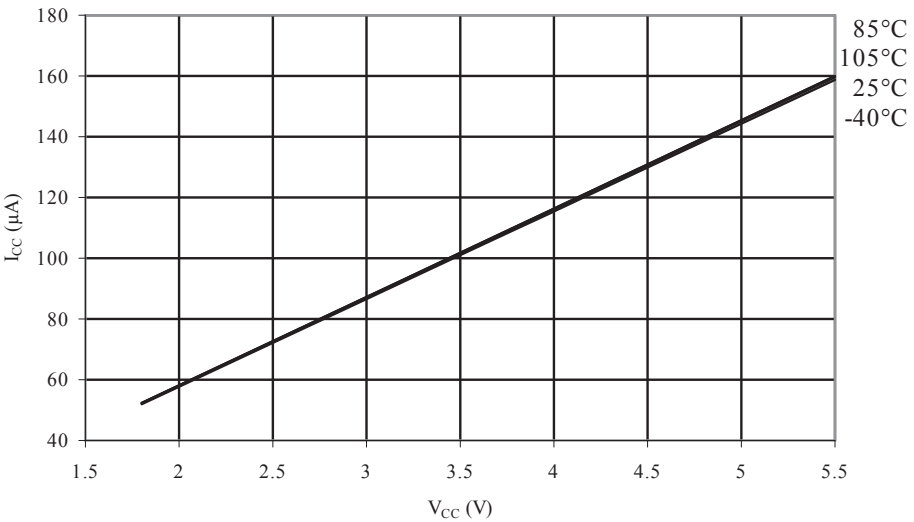


Figure 34-43. ATmega328P: Brownout Detector Current vs. V_{CC}

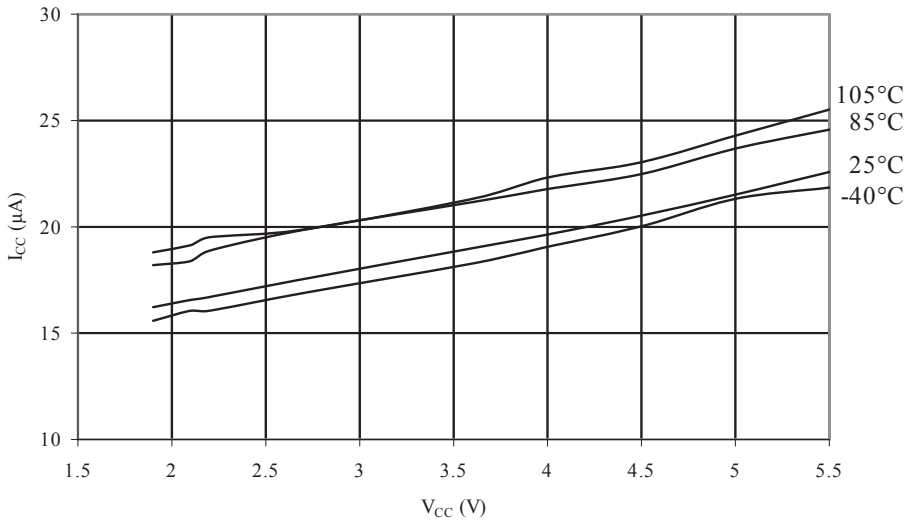
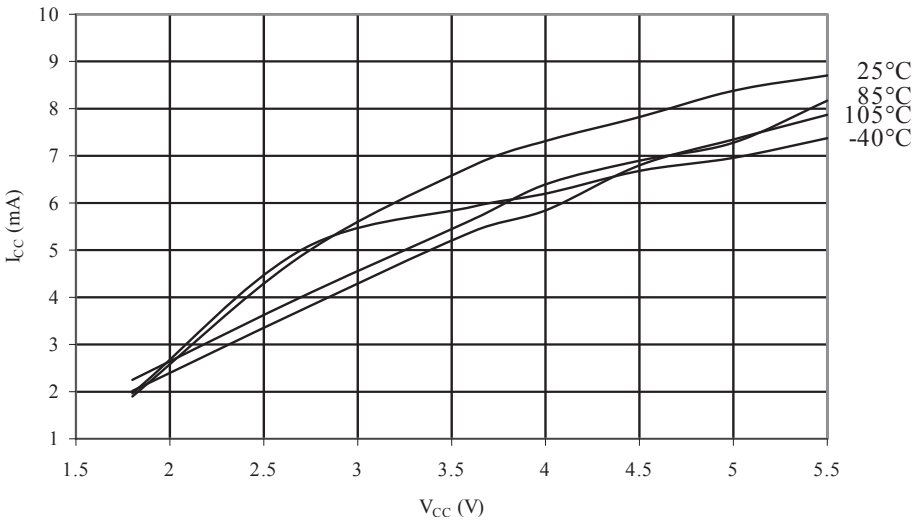


Figure 34-44. ATmega328P: Programming Current vs. V_{CC}



34.1.13. Current Consumption in Reset and Reset Pulsewidth

Figure 34-45. ATmega328P: Reset Supply Current vs. Low Frequency (0.1MHz - 1.0MHz)

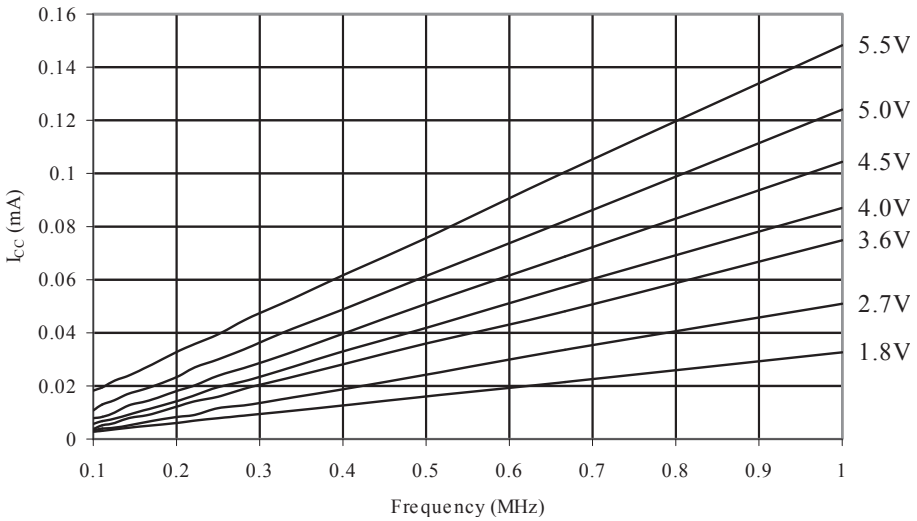


Figure 34-46. ATmega328P: Reset Supply Current vs. Frequency (1MHz - 20MHz)

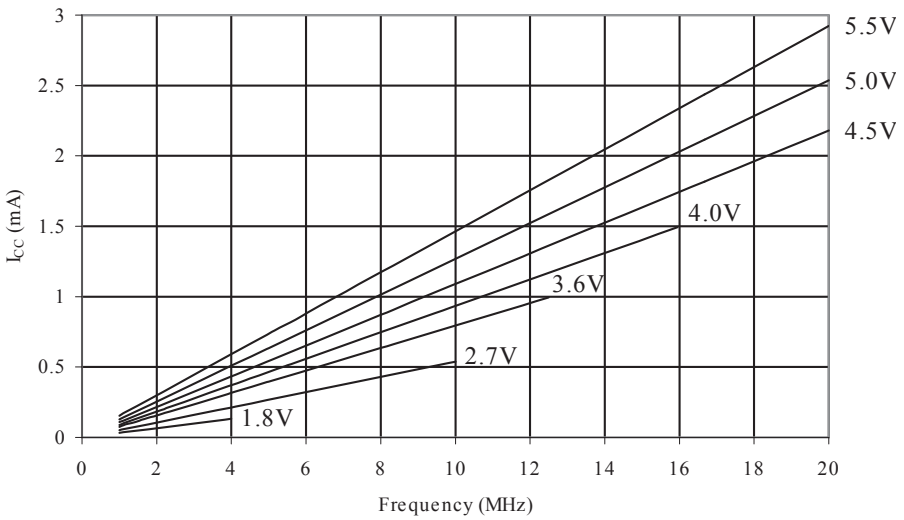


Figure 34-47. ATmega328P: Minimum Reset Pulse Width vs. V_{cc}