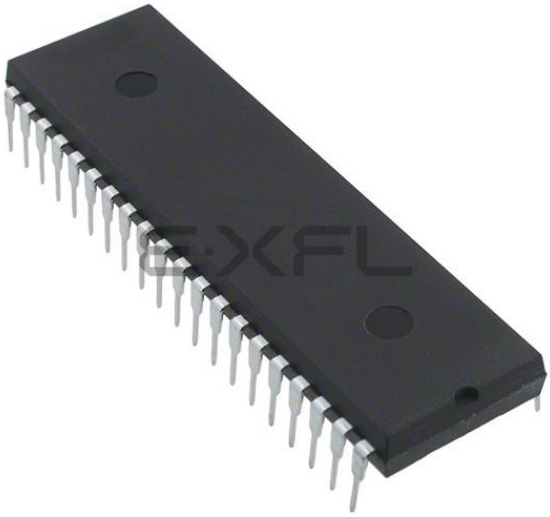


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	AVR
Core Size	8-Bit
Speed	8MHz
Connectivity	SPI, UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	35
Program Memory Size	16KB (8K x 16)
Program Memory Type	FLASH
EEPROM Size	512 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 5.5V
Data Converters	-
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Through Hole
Package / Case	40-DIP (0.600", 15.24mm)
Supplier Device Package	40-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atmega161-8pc

Pin Descriptions

VCC Supply voltage.

GND Ground.

Port A (PA7..PA0) Port A is an 8-bit bi-directional I/O port. Port pins can provide internal pull-up resistors (selected for each bit). The Port A output buffers can sink 20 mA and can drive LED displays directly. When pins PA0 to PA7 are used as inputs and are externally pulled low, they will source current if the internal pull-up resistors are activated. The Port A pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port A serves as a Multiplexed Address/Data port when using external memory interface.

Port B (PB7..PB0) Port B is an 8-bit bi-directional I/O port with internal pull-up resistors. The Port B output buffers can sink 20 mA. 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.

Port B also serves the functions of various special features of the ATmega161 as listed on page 92.

Port C (PC7..PC0) Port C is an 8-bit bi-directional I/O port with internal pull-up resistors. The Port C output buffers can sink 20 mA. 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.

Port C also serves as an address high output when using external memory interface.

Port D (PD7..PD0) Port D is an 8-bit bi-directional I/O port with internal pull-up resistors. The Port D output buffers can sink 20 mA. 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.

Port D also serves the functions of various special features of the ATmega161 as listed on page 101.

Port E (PE2..PE0) Port E is a 3-bit bi-directional I/O port with internal pull-up resistors. The Port E output buffers can sink 20 mA. As inputs, Port E pins that are externally pulled low will source current if the pull-up resistors are activated. The Port E pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port E also serves the functions of various special features of the ATmega161 as listed on page 107.

$\overline{\text{RESET}}$ Reset input. A low level on this pin for more than 500 ns will generate a Reset, even if the clock is not running. Shorter pulses are not guaranteed to generate a Reset.

XTAL1 Input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

XTAL2 Output from the inverting Oscillator amplifier.

In addition to the register operation, the conventional Memory Addressing modes can be used on the Register File. This is enabled by the fact that the Register File is assigned the 32 lowermost Data Space addresses ($00 - $1F), allowing them to be accessed as though they were ordinary memory locations.

The I/O memory space contains 64 addresses for CPU peripheral functions such as Control Registers, Timer/Counters, and other I/O functions. The I/O memory can be accessed directly or as the Data Space locations following those of the Register File, $20 - $5F.

The AVR uses a Harvard architecture concept – with separate memories and buses for program and data. The Program memory is executed with a two-stage pipeline. While one instruction is being executed, the next instruction is pre-fetched from the Program memory. This concept enables instructions to be executed in every clock cycle. The Program memory is Self-programmable Flash memory.

With the jump and call instructions, the whole 8K word address space is directly accessed. Most AVR instructions have a single 16-bit word format. Every Program memory address contains a 16- or 32-bit instruction.

During interrupts and subroutine calls, the return address Program Counter (PC) is stored on the Stack. The Stack is effectively allocated in the general data SRAM and, consequently, the Stack size is only limited by the total SRAM size and the usage of the SRAM. All user programs must initialize the SP (Stack Pointer) in the reset routine (before subroutines or interrupts are executed). The 16-bit Stack Pointer is read/write accessible in the I/O space.

The 1K byte data SRAM can be easily accessed through the five different Addressing modes supported in the AVR architecture.

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

The five different Addressing modes for the Data memory cover: Direct, Indirect with Displacement, Indirect, Indirect with Pre-decrement and Indirect with Post-increment. In the Register File, registers R26 to R31 feature the indirect Addressing Pointer Registers.

The direct addressing reaches the entire data space.

The Indirect with Displacement mode features a 63-address locations reach from the base address given by the Y- or Z-register.

When using Register Indirect Addressing modes with automatic pre-decrement and post-increment, the address registers X, Y, and Z are decremented and incremented.

The 32 general purpose working registers, 64 I/O Registers and the 1K byte of internal data SRAM in the ATmega161 are all accessible through all these Addressing modes.

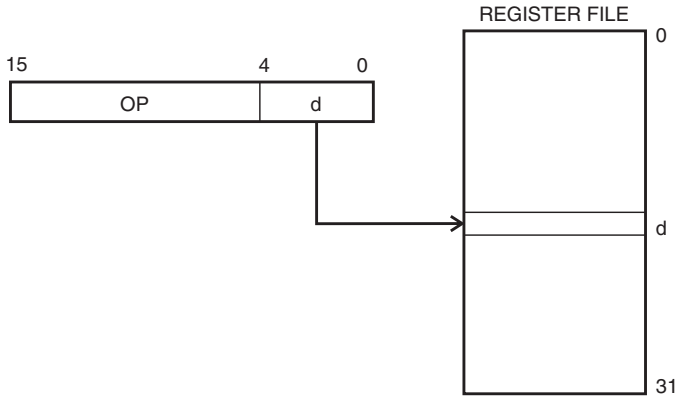
See the next section for a detailed description of the different Addressing modes.

Program and Data Addressing Modes

The ATmega161 AVR RISC microcontroller supports powerful and efficient Addressing modes for access to the Program memory (Flash) and Data memory (SRAM, Register File and I/O memory). This section describes the different Addressing modes supported by the AVR architecture. In the figures, OP means the operation code part of the instruction word. To simplify, not all figures show the exact location of the addressing bits.

Register Direct, Single Register Rd

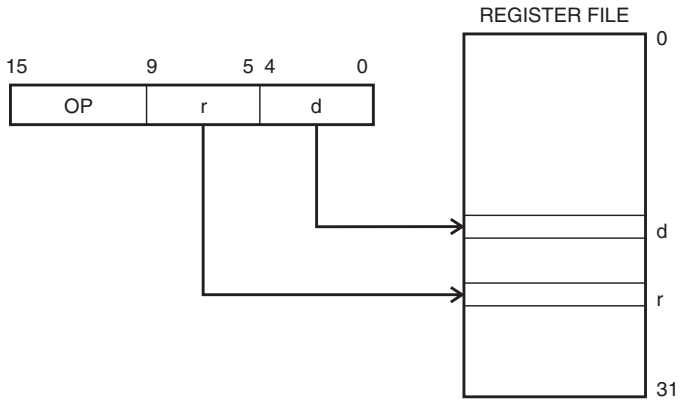
Figure 9. Direct Single Register Addressing



The operand is contained in register d (Rd).

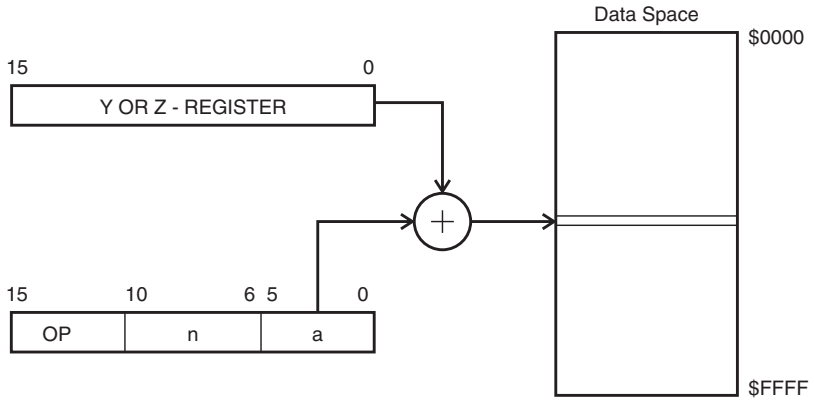
Register Direct, Two Registers Rd and Rr

Figure 10. Direct Register Addressing, Two Registers



Data Indirect with Displacement

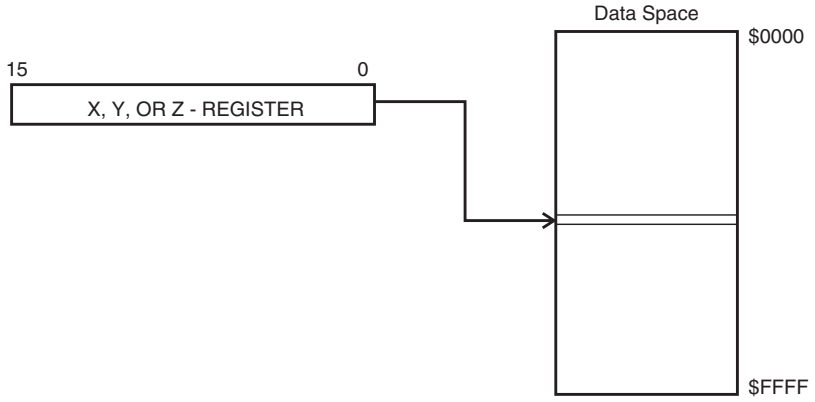
Figure 13. Data Indirect with Displacement



Operand address is the result of the Y- or Z-register contents added to the address contained in six bits of the instruction word.

Data Indirect

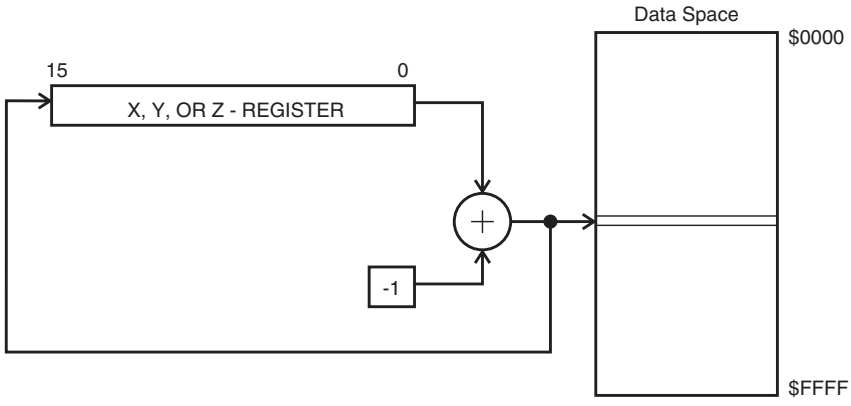
Figure 14. Data Indirect Addressing



Operand address is the contents of the X-, Y-, or Z-register.

Data Indirect with Pre-decrement

Figure 15. Data Indirect Addressing with Pre-decrement



The X-, Y-, or Z-register is decremented before the operation. Operand address is the decremented contents of the X-, Y-, or Z-register.

Table 3. Reset Characteristics (V_{CC} = 5.0V)[1]

Symbol	Parameter	Min	Typ	Max	Units
V_{POT}	Power-on Reset Threshold Voltage (rising)	1.0	1.4	1.8	V
	Power-on Reset Threshold Voltage (falling)[1]	0.4	0.6	0.8	V
V_{RST}	$\overline{\text{RESET}}$ Pin Threshold Voltage			0.85 V_{CC}	V

Note: 1. The Power-on Reset will not work unless the supply voltage has been below V_{POT} (falling).

Table 4. Reset Delay Selections[3]

CKSEL [2:0]	Start-up Time, V_{CC} = 2.7V, SUT Unprogrammed	Start-up Time, V_{CC} = 4.0V, SUT Programmed	Recommended Usage[1]
000	4.2 ms + 6 CK	5.8 ms + 6 CK	External Clock, Fast Rising Power
001	30 µs + 6 CK	10 µs + 6 CK	External Clock[2]
010	67 ms + 16K CK	92 ms + 16K CK	Crystal Oscillator, Slowly Rising Power
011	4.2 ms + 16K CK	5.8 ms + 16K CK	Crystal Oscillator, Fast Rising Power
100	30 µs + 16K CK	10 µs + 16K CK	Crystal Oscillator[2]
101	67 ms + 1K CK	92 ms + 1K CK	Ceramic Resonator/External Clock, Slowly Rising Power
110	4.2 ms + 1K CK	5.8 ms + 1K CK	Ceramic Resonator, Fast Rising Power
111	30 µs + 1K CK	10 µs + 1K CK	Ceramic Resonator[2]

Notes: 1. The CKSEL fuses control only the start-up time. The Oscillator is the same for all selections. On Power-up, the real-time part of the start-up time is increased with typ. 0.6 ms.
2. External Power-on Reset.
3. Table 4 shows the Start-up Times from Reset. From sleep, only the clock counting part of the start-up time is used. The Watchdog Oscillator is used for timing the real-time part of the start-up time. The number WDT Oscillator cycles used for each time-out is shown in Table 5.

Table 5. Number of Watchdog Oscillator Cycles

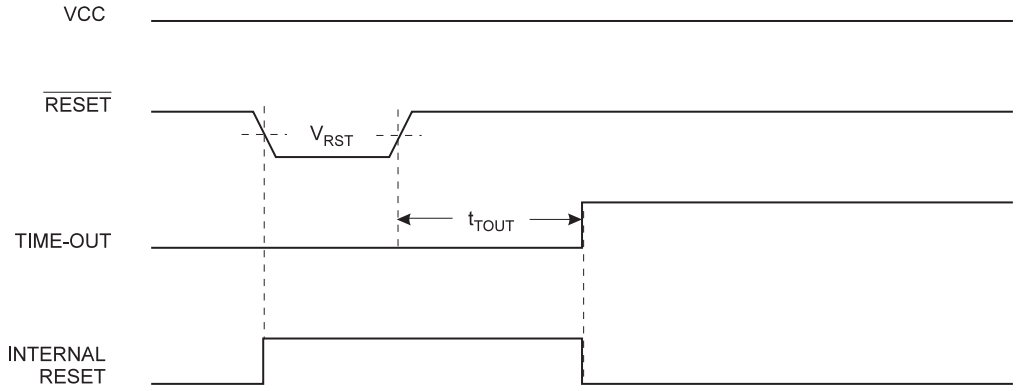
SUT	Time-out	Number of Cycles
Unprogrammed	4.2 ms (at V_{CC} = 2.7V)	1K
Unprogrammed	67 ms (at V_{CC} = 2.7V)	16K
Programmed	5.8 ms (at V_{CC} = 4.0V)	4K
Programmed	92 ms (at V_{CC} = 4.0V)	64K

The frequency of the Watchdog Oscillator is voltage-dependent as shown in the Electrical Characteristics section. The device is shipped with CKSEL = 010.

External Reset

An External Reset is generated by a low level on the $\overline{\text{RESET}}$ pin. Reset pulses longer than 500 ns will generate a Reset, even if the clock is not running. Shorter pulses are not guaranteed to generate a Reset. When the applied signal reaches the Reset Threshold Voltage (V_{RST}) on its positive edge, the delay timer starts the MCU after the Time-out period (t_{TOUT}) has expired.

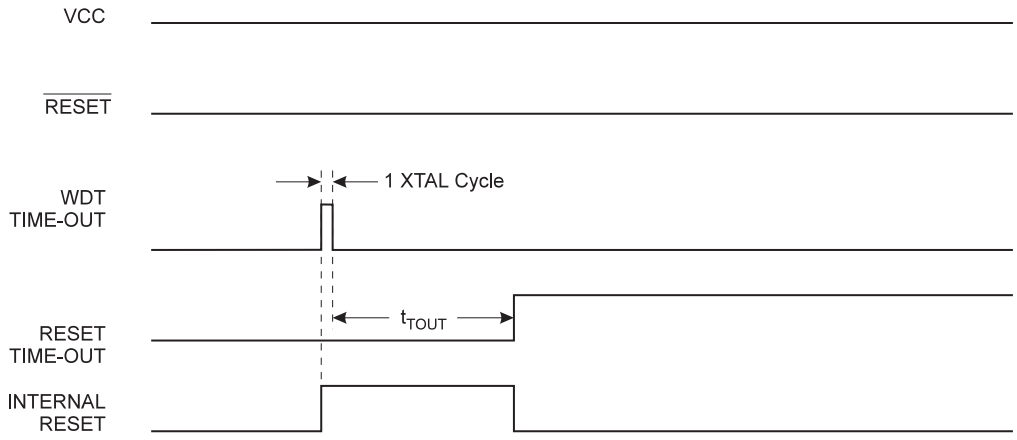
Figure 27. External Reset during Operation



Watchdog Reset

When the Watchdog times out, it will generate a short reset pulse of 1 XTAL cycle duration. On the falling edge of this pulse, the delay timer starts counting the Time-out period (t_{TOUT}). Refer to page 58 for details on operation of the Watchdog.

Figure 28. Watchdog Reset during Operation



MCU Status Register – MCUSR

The MCU Status Register provides information on which reset source caused an MCU Reset.

Bit	7	6	5	4	3	2	1	0	
$34 ($54)	–	–	–	–	WDRF	–	EXTRF	PORF	MCUSR
Read/Write	R	R	R	R	R/W	R	R/W	R/W	
Initial Value	0	0	0	0	See Bit Description				

- **Bits 7..4 – Res: Reserved Bits**

These bits are reserved bits in the ATmega161 and always read as zero.

- **Bit 3 – WDRF: Watchdog Reset Flag**

This bit is set if a Watchdog Reset occurs. The bit is cleared by a Power-on Reset or by writing a logical “0” to the Flag.

- **Bit 2 – Res: Reserved Bit**

This bit are reserved bit in the ATmega161 and always read as zero.

- **Bit 1 – EXTRF: External Reset Flag**

This bit is set if an External Reset occurs. The bit is cleared by a Power-on Reset or by writing a logical “0” to the Flag.

- **Bit 0 – PORF: Power-on Reset Flag**

This bit is set if a Power-on Reset occurs. The bit is cleared only by writing a logical “0” to the Flag.

To make use of the Reset Flags to identify a reset condition, the user should read and then clear 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.

Interrupt Handling

The ATmega161 has two 8-bit Interrupt Mask Control Registers; GIMSK (General Interrupt Mask Register) and TIMSK (Timer/Counter Interrupt Mask Register).

When an interrupt occurs, the Global Interrupt Enable I-bit is cleared (zero) and all interrupts are disabled. The user software can set (one) the I-bit to enable nested interrupts. The I-bit is set (one) when a Return from Interrupt instruction (RETI) is executed.

When the Program Counter is vectored to the actual Interrupt Vector in order to execute the interrupt handling routine, hardware clears the corresponding flag that generated the interrupt. Some of the Interrupt Flags can also be cleared by writing a logical “1” to the Flag bit position(s) to be cleared.

If an interrupt condition occurs when the corresponding interrupt enable bit is cleared (zero), the Interrupt Flag will be set and remembered until the interrupt is enabled or the Flag is cleared by software.

If one or more interrupt conditions occur when the global interrupt enable bit is cleared (zero), the corresponding Interrupt Flag(s) will be set and remembered until the global interrupt enable bit is set (one), and will be executed by order of priority.

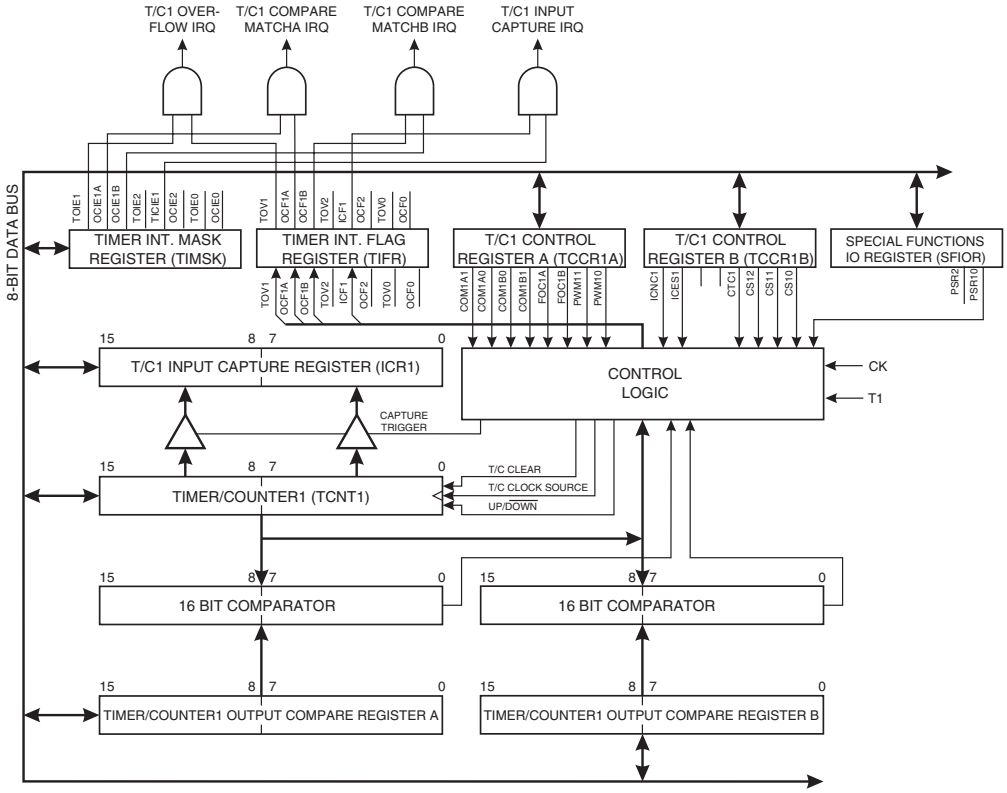
Note that external level interrupt does not have a flag and will only be remembered for as long as the interrupt condition is present.

- Description of wake-up from Power-save mode when the timer is clocked asynchronously: When the interrupt condition is met, the wake-up process is started on the following cycle of the timer clock, that is, the timer is always advanced by at least 1 before the processor can read the counter value. The Interrupt Flags are updated three processor cycles after the processor clock has started. During these cycles, the processor executes instructions, but the interrupt condition is not readable and the interrupt routine has not started yet.
- During asynchronous operation, the synchronization of the Interrupt Flags for the asynchronous timer takes three processor cycles plus one timer cycle. The timer is therefore advanced by at least 1 before the processor can read the timer value, causing the setting of the Interrupt Flag. The output compare pin is changed on the timer clock and is not synchronized to the processor clock.

Timer/Counter1

Figure 35 shows the block diagram for Timer/Counter1.

Figure 35. Timer/Counter1 Block Diagram



The 16-bit Timer/Counter1 can select clock source from CK, prescaled CK or an external pin. In addition, it can be stopped as described in “Timer/Counter1 Control Register B – TCCR1B”. The different Status Flags (Overflow, Compare Match, and Capture Event) are found in the Timer/Counter Interrupt Flag Register (TIFR). Control signals are found in the Timer/Counter1 Control Registers (TCCR1A and TCCR1B). The interrupt enable/disable settings for Timer/Counter1 are found in the Timer/Counter Interrupt Mask Register (TIMSK).

When Timer/Counter1 is externally clocked, the external signal is synchronized with the Oscillator frequency of the CPU. To assure proper sampling of the external clock, the minimum time between two external clock transitions must be at least one internal CPU

The Timer/Counter1 is realized as an up or up/down (in PWM mode) counter with read and write access. If Timer/Counter1 is written to and a clock source is selected, the Timer/Counter1 continues counting in the timer clock cycle after it is preset with the written value.

Timer/Counter1 Output Compare Register – OCR1AH AND OCR1AL

Bit	15	14	13	12	11	10	9	8	
$2B ($4B)	MSB								OCR1AH
$2A ($4A)								LSB	OCR1AL
	7	6	5	4	3	2	1	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

Timer/Counter1 Output Compare Register – OCR1BH AND OCR1BL

Bit	15	14	13	12	11	10	9	8	
$29 ($49)	MSB								OCR1BH
$28 ($48)								LSB	OCR1BL
	7	6	5	4	3	2	1	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

The Output Compare Registers are 16-bit read/write registers.

The Timer/Counter1 Output Compare Registers contain the data to be continuously compared with Timer/Counter1. Actions on compare matches are specified in the Timer/Counter1 Control and Status Registers. A software write to the Timer/Counter Register blocks compare matches in the next Timer/Counter clock cycle. This prevents immediate interrupts when initializing the Timer/Counter.

A compare match will set the Compare Interrupt Flag in the CPU clock cycle following the compare event.

Since the Output Compare Registers (OCR1A and OCR1B) are 16-bit registers, a temporary register (TEMP) is used when OCR1A/B are written to ensure that both bytes are updated simultaneously. When the CPU writes the High byte, OCR1AH or OCR1BH, the data is temporarily stored in the TEMP Register. When the CPU writes the Low byte, OCR1AL or OCR1BL, the TEMP Register is simultaneously written to OCR1AH or OCR1BH. Consequently, the High byte OCR1AH or OCR1BH must be written first for a full 16-bit register write operation.

The TEMP Register is also used when accessing TCNT1 and ICR1. If the main program and interrupt routines perform access to registers using TEMP, interrupts must be disabled during access from the main program and interrupt routines.

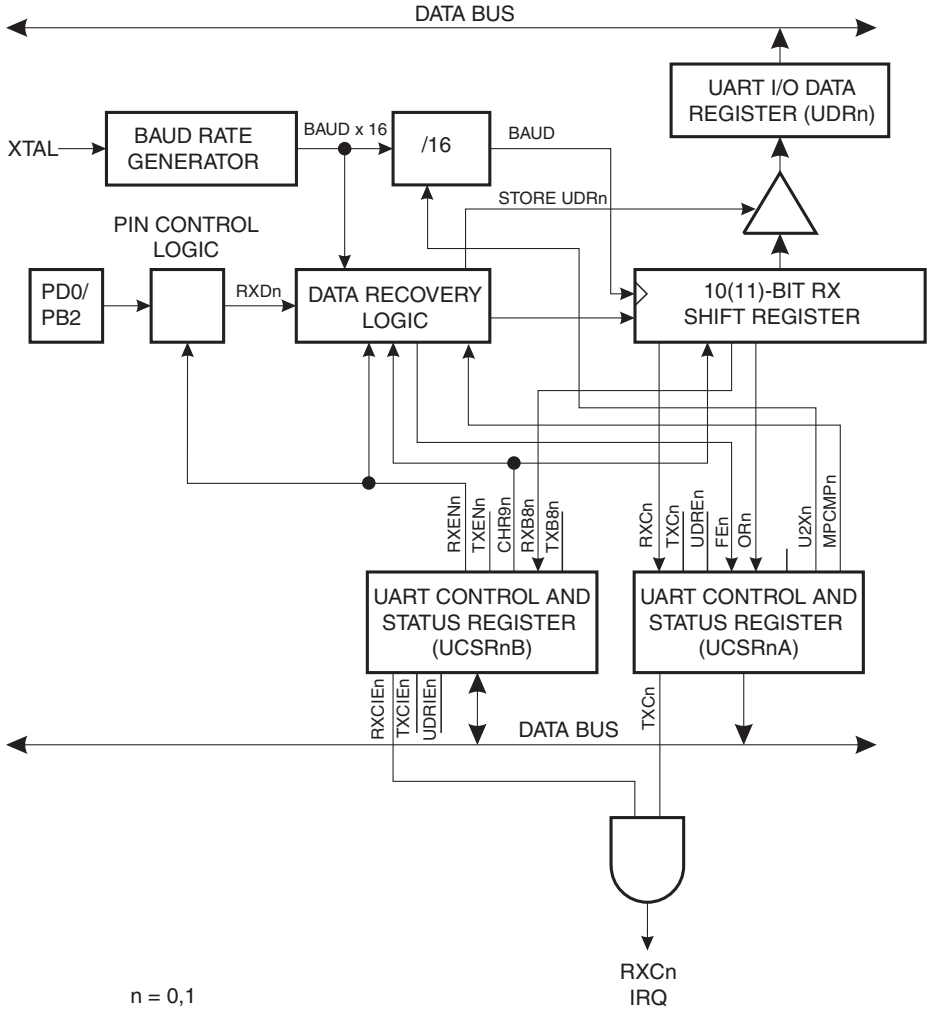
Timer/Counter1 Input Capture Register – ICR1H AND ICR1L

Bit	15	14	13	12	11	10	9	8	
$25 ($45)	MSB								ICR1H
$24 ($44)								LSB	ICR1L
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R	R	R	R	
	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

Data Reception

Figure 45 shows a block diagram of the UART Receiver.

Figure 45. UART Receiver



The Receiver front-end logic samples the signal on the RXDn pin at a frequency 16 times the baud rate. While the line is idle, one single sample of logical “0” will be interpreted as the falling edge of a start bit, and the start bit detection sequence is initiated. Let sample 1 denote the first zero-sample. Following the 1-to-0 transition, the Receiver samples the RXDn pin at samples 8, 9 and 10. If two or more of these three samples are found to be logical “1”s, the start bit is rejected as a noise spike and the Receiver starts looking for the next 1-to-0 transition.

If, however, a valid start bit is detected, sampling of the data bits following the start bit is performed. These bits are also sampled at samples 8, 9 and 10. The logical value found in at least two of the three samples is taken as the bit value. All bits are shifted into the Transmitter Shift Register as they are sampled. Sampling of an incoming character is shown in Figure 46. Note that the description above is not valid when the UART transmission speed is doubled. See “Double-speed Transmission” on page 78 for a detailed description.

UART1 Control and Status Registers – UCSR1A

Bit	7	6	5	4	3	2	1	0	
$02 ($22)	RXC1	TXC1	UDRE1	FE1	OR1	–	U2X1	MPCM1	UCSR1A
Read/Write	R	R/W	R	R	R	R	R/W	R/W	
Initial Value	0	0	1	0	0	0	0	0	

- **Bit 7 – RXC0/RXC1: UART Receive Complete**

This bit is set (one) when a received character is transferred from the Receiver Shift Register to UDRn. The bit is set regardless of any detected framing errors. When the RXCIEn bit in UCSRnB is set, the UART Receive Complete interrupt will be executed when RXCn is set (one). RXCn is cleared by reading UDRn. When interrupt-driven data reception is used, the UART Receive Complete Interrupt routine must read UDRn in order to clear RXCn; otherwise, a new interrupt will occur once the interrupt routine terminates.

- **Bit 6 – TXC0/TXC1: UART Transmit Complete**

This bit is set (one) when the entire character (including the stop bit) in the Transmit Shift Register has been shifted out and no new data has been written to UDRn. This Flag is especially useful in half-duplex communications interfaces, where a transmitting application must enter Receive mode and free the communications bus immediately after completing the transmission.

When the TXCIEn bit in UCSRnB is set, setting of TXCn causes the UART Transmit Complete interrupt to be executed. TXCn is cleared by hardware when executing the corresponding Interrupt Handling Vector. Alternatively, the TXCn bit is cleared (zero) by writing a logical “1” to the bit.

- **Bit 5 – UDRE0/UDRE1: UART Data Register Empty**

This bit is set (one) when a character written to UDRn is transferred to the Transmit Shift Register. Setting of this bit indicates that the Transmitter is ready to receive a new character for transmission.

When the UDRIEn bit in UCSRnB is set, the UART Transmit Complete interrupt will be executed as long as UDREn is set and the global interrupt enable bit in SREG is set. UDREn is cleared by writing UDRn. When interrupt-driven data transmittal is used, the UART Data Register Empty Interrupt routine must write UDRn in order to clear UDREn, otherwise a new interrupt will occur once the interrupt routine terminates.

UDREn is set (one) during reset to indicate that the Transmitter is ready.

- **Bit 4 – FE0/FE1: Framing Error**

This bit is set if a Framing Error condition is detected, i.e., when the stop bit of an incoming character is zero.

The FEn bit is cleared when the stop bit of received data is one.

- **Bit 3 – OR0/OR1: OverRun**

This bit is set if an Overrun condition is detected, i.e., when a character already present in the UDRn Register is not read before the next character has been shifted into the Receiver Shift Register. The ORn bit is buffered, which means that it will be set once the valid data still in UDRn is read.

The ORn bit is cleared (zero) when data is received and transferred to UDRn.

Table 25. UBR Settings at Various Crystal Frequencies in Double-speed Mode

Baud Rate	1.0000 MHz	% Error	1.8432 MHz	% Error	2.0000 MHz	% Error
2400	UBR = **51**	0.2	UBR = **95**	0.0	UBR = **103**	0.2
4800	UBR = **25**	0.2	UBR = **47**	0.0	UBR = **51**	0.2
9600	UBR = **12**	0.2	UBR = **23**	0.0	UBR = **25**	0.2
14400	**UBR = 8**	3.7	UBR = **15**	0.0	UBR = 16	2.1
19200	**UBR = 6**	7.5	UBR = **11**	0.0	UBR = **12**	0.2
28800	**UBR = 3**	7.8	UBR = **7**	0.0	UBR = 8	3.7
38400	**UBR = 2**	7.8	UBR = **5**	0.0	UBR = 6	7.5
57600	**UBR = 1**	7.8	UBR = **3**	0.0	UBR = 3	7.8
76800	**UBR = 1**	22.9	UBR = **2**	0.0	UBR = 2	7.8
115200	**UBR = 0**	84.3	UBR = **1**	0.0	UBR = 1	7.8
230400	**-**	-	UBR = **0**	0.0	UBR = 0	84.3
Baud Rate	3.2768 MHz	% Error	3.6864 MHz	% Error	4.0000 MHz	% Error
2400	UBR = **170**	0.2	UBR = **191**	0.0	UBR = **207**	0.2
4800	UBR = **84**	0.4	UBR = **95**	0.0	UBR = **103**	0.2
9600	UBR = **42**	0.8	UBR = **47**	0.0	UBR = **51**	0.2
14400	UBR = 27	1.6	UBR = **31**	0.0	UBR = **34**	0.8
19200	UBR = 20	1.6	UBR = **23**	0.0	UBR = **25**	0.2
28800	UBR = 13	1.6	UBR = **15**	0.0	UBR = 16	2.1
38400	UBR = 10	3.1	UBR = **11**	0.0	UBR = **12**	0.2
57600	UBR = 6	1.6	UBR = **7**	0.0	UBR = 8	3.7
76800	UBR = 4	6.2	UBR = **5**	0.0	UBR = 6	7.5
115200	UBR = 3	12.5	UBR = **3**	0.0	UBR = 3	7.8
230400	UBR = 1	12.5	UBR = **1**	0.0	UBR = 1	7.8
460800	UBR = 0	12.5	UBR = **0**	0.0	UBR = 0	7.8
912600	-	-	-	-	UBR = 0	84.3
Baud Rate	7.3728 MHz	% Error	8.0000 MHz	% Error	9.2160 MHz	% Error
2400	UBR = **383**	0.0	UBR = **416**	0.1	UBR = **479**	0.0
4800	UBR = **191**	0.0	UBR = **207**	0.2	UBR = **239**	0.0
9600	UBR = **95**	0.0	UBR = **103**	0.2	UBR = **119**	0.0
14400	UBR = **63**	0.0	UBR = **68**	0.6	UBR = **79**	0.0
19200	UBR = **47**	0.0	UBR = **51**	0.2	UBR = **59**	0.0
28800	UBR = **31**	0.0	UBR = **34**	0.8	UBR = **39**	0.0
38400	UBR = **23**	0.0	UBR = **25**	0.2	UBR = **29**	0.0
57600	UBR = **15**	0.0	UBR = 16	2.1	UBR = **19**	0.0
76800	UBR = **11**	0.0	UBR = 12	0.2	UBR = **14**	0.0
115200	UBR = **7**	0.0	UBR = 8	3.7	UBR = **9**	0.0
230400	UBR = **3**	0.0	UBR = 3	7.8	UBR = **4**	0.0
460800	UBR = **1**	0.0	UBR = 1	7.8	UBR = 2	20.0
912600	UBR = **0**	0.0	UBR = 0	7.8	UBR = 0	20.0

Port B

Port B is an 8-bit bi-directional I/O port.

Three I/O memory address locations are allocated for the Port B, one each for the Data Register – PORTB, $18($38), Data Direction Register – DDRB, $17($37) and the Port B Input Pins – PINB, $16($36). The Port B Input Pins address is read-only, while the Data Register and the Data Direction Register are read/write.

All port pins have individually selectable pull-up resistors. The Port B output buffers can sink 20 mA and thus drive LED displays directly. When pins PB0 to PB7 are used as inputs and are externally pulled low, they will source current if the internal pull-up resistors are activated.

The Port B pins with alternate functions are shown in Table 30.

Table 30. Port B Pin Alternate Functions(1)

Port Pin	Alternate Functions
PB0	OC0 (Timer/Counter0 Compare Match Output)/T0 (Timer/Counter0 External Counter Input)
PB1	OC2 (Timer/Counter2 Compare Match Output)/T1 (Timer/Counter1 External Counter Input)
PB2	RXD1 (UART1 Input Line)/AIN0 (Analog Comparator Positive Input)
PB3	TXD1 (UART1 Output Line)/AIN1 (Analog Comparator Negative Input)
PB4	$\overline{SS}$ (SPI Slave Select Input)
PB5	MOSI (SPI Bus Master Output/Slave Input)
PB6	MISO (SPI Bus Master Input/Slave Output)
PB7	SCK (SPI Bus Serial Clock)

Note: 1. When the pins are used for the alternate function, the DDRB and PORTB Registers have to be set according to the alternate function description.

Port B Data Register – PORTB

Bit	7	6	5	4	3	2	1	0	
$18 ($38)	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	PORTB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Port B Data Direction Register – DDRB

Bit	7	6	5	4	3	2	1	0	
$17 ($37)	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	DDRB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Port B Input Pins Address – PINB

Bit	7	6	5	4	3	2	1	0	
$16 ($36)	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	PINB
Read/Write	R	R	R	R	R	R	R	R	
Initial Value	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	

The Port B Input Pins address (PINB) is not a register; this address enables access to the physical value on each Port B pin. When reading PORTB, the Port B Data Latch is read and when reading PINB, the logical values present on the pins are read.

Parallel Programming

This section describes how to parallel program and verify Flash Program memory, EEPROM Data memory, Lock bits and Fuse bits in the ATmega161. Pulses are assumed to be at least 500 ns unless otherwise noted.

Signal Names

In this section, some pins of the ATmega161 are referenced by signal names describing their functionality during parallel programming (see Figure 75 and Table 42). Pins not described in the following table are referenced by pin name.

The XA1/XA0 pins determine the action executed when the XTAL1 pin is given a positive pulse. The bit codings are shown in Table 43.

When pulsing $\overline{WR}$ or $\overline{OE}$, the command loaded determines the action executed. The command is a byte where the different bits are assigned functions as shown in Table 44.

Figure 75. Parallel Programming

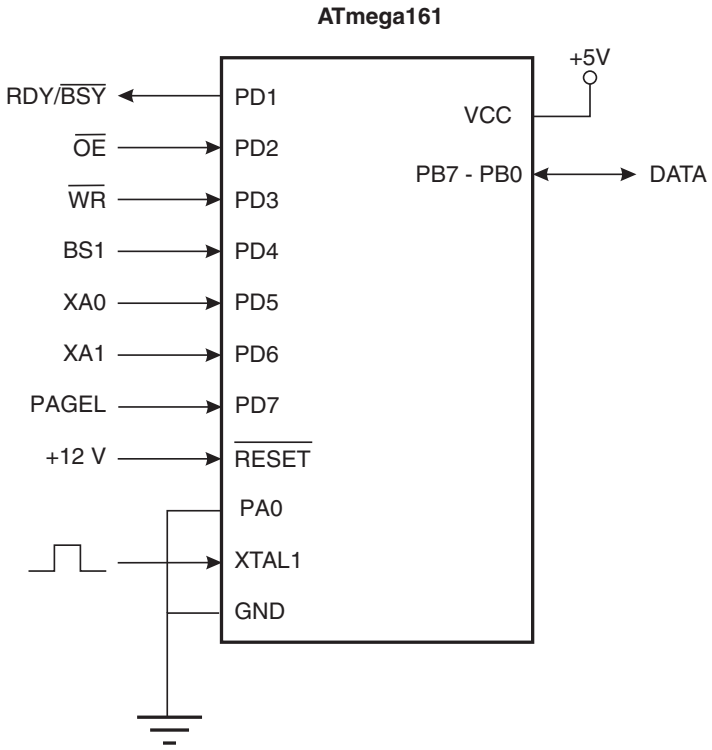


Table 42. Pin Name Mapping

Signal Name in Programming Mode	Pin Name	I/O	Function
RDY/$\overline{BSY}$	PD1	O	0: Device is busy programming; 1: Device is ready for new command
$\overline{OE}$	PD2	I	Output Enable (Active low)
$\overline{WR}$	PD3	I	Write Pulse (Active low)
BS1	PD4	I	Byte Select 1 (“0” selects Low byte, “1” selects High byte)
XA0	PD5	I	XTAL Action Bit 0
XA1	PD6	I	XTAL Action Bit 1

Reading the Fuse and Lock Bits

The algorithm for reading the Fuse and Lock bits is as follows (refer to “Programming the Flash” on page 120 for details on command loading):

1. A: Load Command “0000 0100”.
2. Set $\overline{OE}$ to “0”, and BS to “0”. The status of the Fuse bits can now be read at DATA (“0” means programmed).
   Bit 6 = BOOTRST Fuse bit
   Bit 5 = SPIEN Fuse bit
   Bit 4 = SUT Fuse bit
   Bit 3 = “1”. This bit is reserved and must be left unprogrammed (“1”).
   Bits 2 - 0 = CKSEL2..0 Fuse bits
3. Set $\overline{OE}$ to “0”, and BS to “1”. The status of the Lock bits can now be read at DATA (“0” means programmed).
   Bit 5 = Boot Lock Bit12
   Bit 4 = Boot Lock Bit11
   Bit 3 = Boot Lock Bit02
   Bit 2 = Boot Lock Bit01
   Bit 1 = Lock Bit2
   Bit 0 = Lock Bit1
4. Set $\overline{OE}$ to “1”.

Reading the Signature Bytes

The algorithm for reading the Signature bytes is as follows (refer to “Programming the Flash” on page 120 for details on command and address loading):

1. A: Load Command “0000 1000”.
2. C: Load Address Low Byte ($00 - $02).
   Set $\overline{OE}$ to “0”, and BS to “0”. The selected Signature byte can now be read at DATA.
3. Set $\overline{OE}$ to “1”.

Parallel Programming Characteristics

Figure 79. Parallel Programming Timing

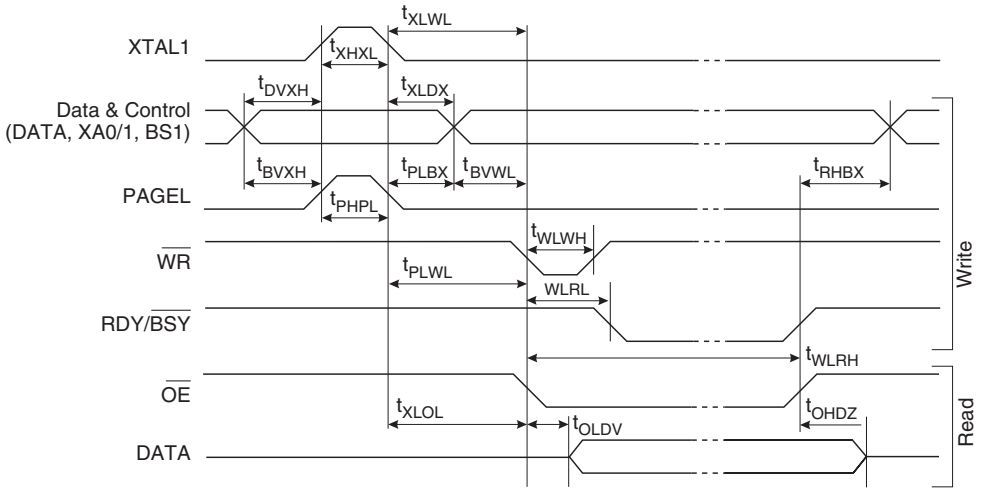


Table 48. Serial Programming Instruction Set[1]

Instruction	Instruction Format				Operation
	Byte 1	Byte 2	Byte 3	Byte 4	
Programming Enable	1010 1100	0101 0011	xxxx xxxx	xxxx xxxx	Enable Serial Programming after $\overline{\text{RESET}}$ goes low.
Chip Erase	1010 1100	100x xxxx	xxxx xxxx	xxxx xxxx	Chip Erase EEPROM and Flash.
Read Program Memory	0010 **H**000	xxxa **aaaa**	**bbbb bbbb**	**oooo oooo**	Read **H** (high or low) data **o** from Program memory at word address **a**:**b**.
Load Program Memory Page	0100 **H**000	xxxx xxxx	xx**bb bbbb**	**iiii iiii**	Write **H** (high or low) data **i** to Program memory page at word address **b**.
Write Program Memory Page	0100 1100	xxxa **aaaa**	**bb**xx xxxx	iiii iiii	Write Program memory page at address **a**:**b**.
Read EEPROM Memory	1010 0000	xxxx xxx**a**	**bbbb bbbb**	**oooo oooo**	Read data **o** from EEPROM memory at address **a**:**b**.
Write EEPROM Memory	1100 0000	xxxx xxx**a**	**bbbb bbbb**	**iiii iiii**	Write data **i** to EEPROM memory at address **a**:**b**.
Read Lock bits	0101 1000	xxxx xxxx	xxxx xxxx	xx***65 4321***	Read Lock bits. “0” = programmed, “1” = unprogrammed.
Write Lock bits	1010 1100	111x xxxx	xxxx xxxx	11***65 4321***	Write Lock bits. Set bits ***6*** - ***1*** = “0” to program Lock bits.
Read Signature Byte	0011 0000	xxxx xxxx	xxxx xx**bb**	**oooo oooo**	Read Signature byte **o** at address **b**.
Write Fuse bits	1010 1100	101x xxxx	xxxx xxxx	1**D**1**B 1987**	Set bits **D** - **B**, **9** - **7** = “0” to program, “1” to unprogram
Read Fuse bits	0101 0000	xxxx xxxx	xxxx xxxx	x**D**C**B 1987**	Read Fuse bits. “0” = programmed, “1” = unprogrammed

Note: 1. **a** = address high bits
b = address low bits
H = 0 – Low byte, 1 – High Byte
o = data out
i = data in
x = don’t care
1 = lock bit 1
2 = lock bit 2
3 = Boot Lock Bit1
4 = Boot Lock Bit2
5 = Boot Lock Bit11
6 = Boot Lock Bit12
7 = CKSEL0 Fuse
8 = CKSEL1 Fuse
9 = CKSEL2 Fuse
B = SUT Fuse
C = SPIEN Fuse
D = BOOTRST Fuse

Figure 91. Pull-up Resistor Current vs. Input Voltage

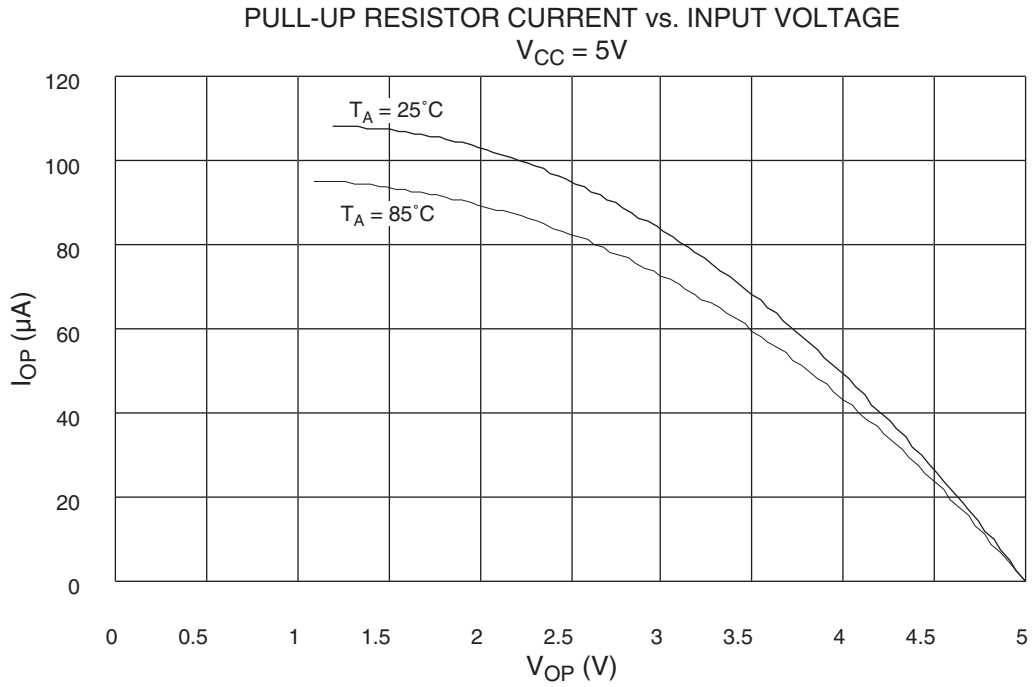


Figure 92. Pull-up Resistor Current vs. Input Voltage

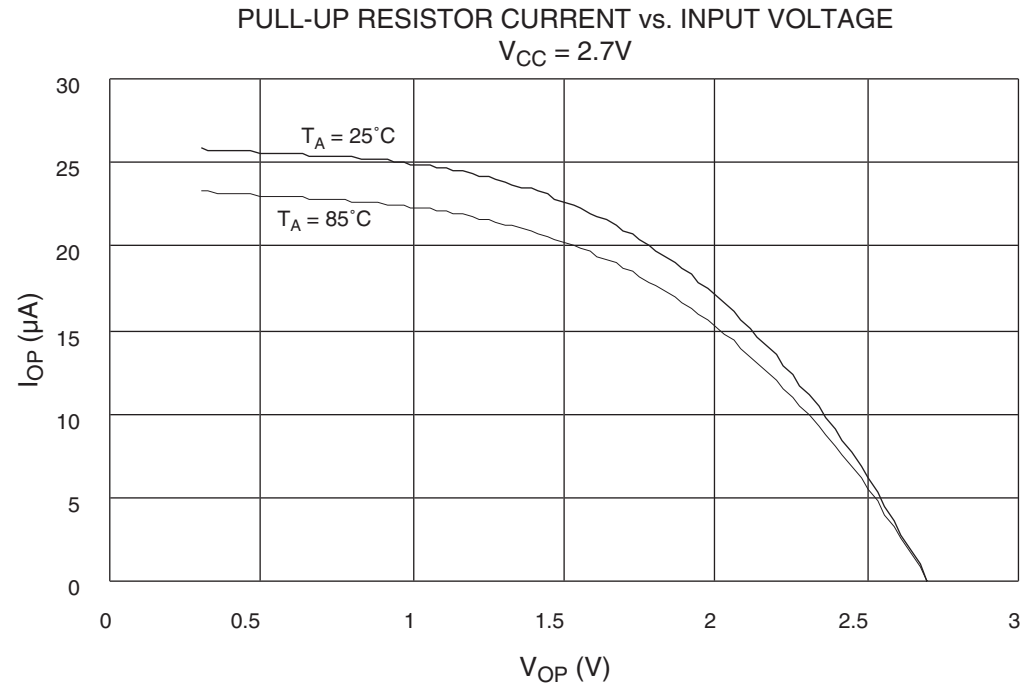


Figure 95. I/O Pin Sink Current vs. Output Voltage

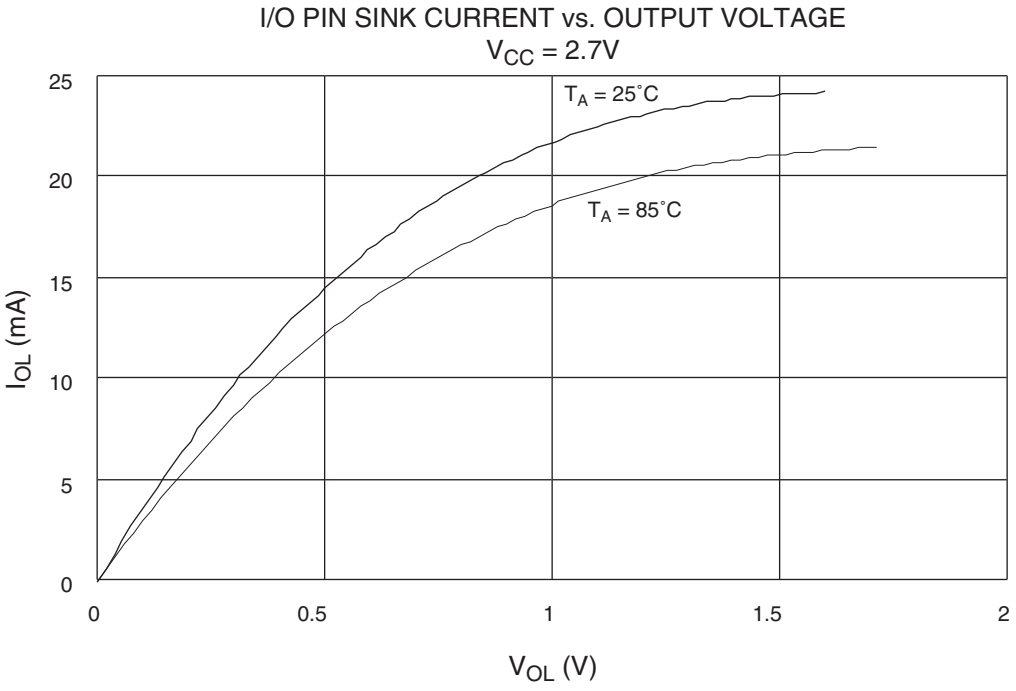
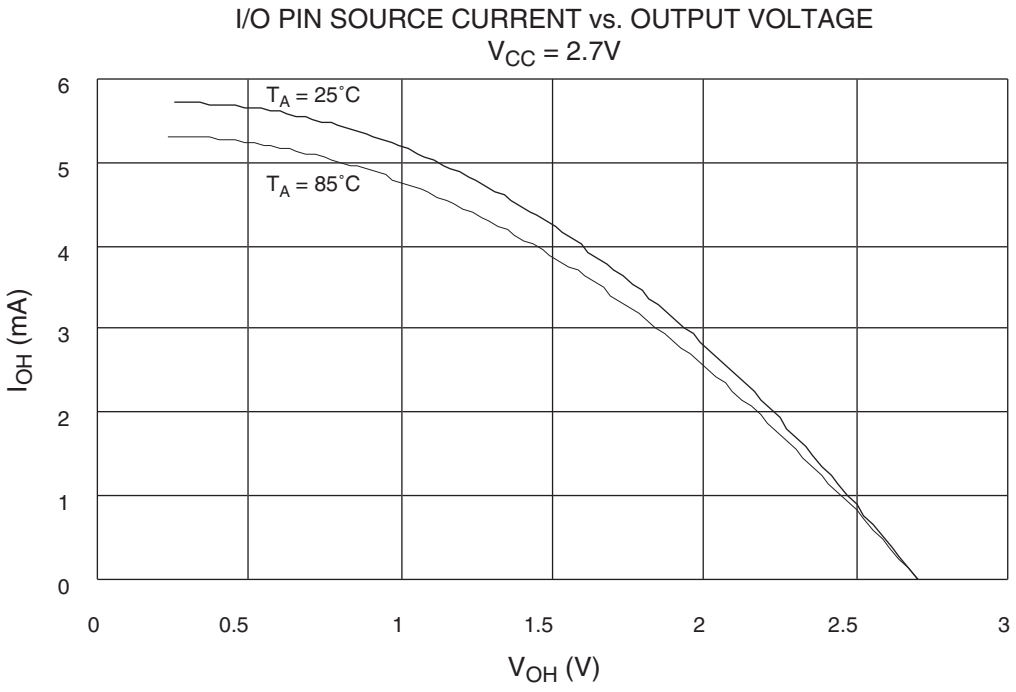


Figure 96. I/O Pin Source Current vs. Output Voltage



Register Summary

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
$3F ($5F)	SREG	I	T	H	S	V	N	Z	C	page 21
$3E ($5E)	SPH	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP8	page 22
$3D ($5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	page 22
$3C ($5C)	Reserved									
$3B ($5B)	GIMSK	INT1	INT0	INT2	-	-	-	-	-	page 30
$3A ($5A)	GIFR	INTF1	INTF0	INTF2						page 31
$39 ($59)	TIMSK	TOIE1	OCIE1A	OCIE1B	TOIE2	TICIE1	OCIE2	TOIE0	OCIE0	page 31
$38 ($58)	TIFR	TOV1	OCF1A	OCF1B	TOV2	ICF1	OCFI2	TOV0	OCIF0	page 33
$37 ($57)	SPMCR	-	-	-	-	LBSET	PGWRT	PGERS	SPMEN	page 114
$36 ($56)	EMCUCR	SM0	SRL2	SRL1	SRL0	SRW01	SRW00	SRW11	ISC2	page 36
$35 ($55)	MCUCR	SRE	SRW10	SE	SM1	ISC11	ISC10	ISC01	ISC00	page 34
$34 ($54)	MCUSR	-	-	-	-	WDRF	BORF	EXTRF	PORF	page 29
$33 ($53)	TCCR0	FOC0	PWM0	COM01	COM00	CTC0	CS02	CS01	CS00	page 42
$32 ($52)	TCNT0	Timer/Counter0 Counter Register								page 44
$31 ($51)	OCR0	Timer/Counter0 Output Compare Register								page 44
$30 ($50)	SFIOR	-	-	-	-	-	-	PSR2	PSR10	page 39
$2F ($4F)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	PWM11	PWM10	page 50
$2E ($4E)	TCCR1B	ICNC1	ICES1	-	-	CTC1	CS12	CS11	CS10	page 52
$2D ($4D)	TCNT1H	Timer/Counter1 - Counter Register High Byte								page 53
$2C ($4C)	TCNT1L	Timer/Counter1 - Counter Register Low Byte								page 53
$2B ($4B)	OCR1AH	Timer/Counter1 - Output Compare Register A High Byte								page 54
$2A ($4A)	OCR1AL	Timer/Counter1 - Output Compare Register A Low Byte								page 54
$29 ($49)	OCR1BH	Timer/Counter1 - Output Compare Register B High Byte								page 54
$28 ($48)	OCR1BL	Timer/Counter1 - Output Compare Register B Low Byte								page 54
$27 ($47)	TCCR2	FOC2	PWM2	COM21	COM20	CTC2	CS22	CS21	CS20	page 42
$26 ($46)	ASSR	-	-	-	-	AS20	TCON2UB	OCR2UB	TCR2UB	page 47
$25 ($45)	ICR1H	Timer/Counter1 - Input Capture Register High Byte								page 54
$24 ($44)	ICR1L	Timer/Counter1 - Input Capture Register Low Byte								page 54
$23 ($43)	TCNT2	Timer/Counter2 Counter Register								page 44
$22 ($42)	OCR2	Timer/Counter2 Output Compare Register								page 44
$21 ($41)	WDTCR	-	-	-	WDTOE	WDE	WDP2	WDP1	WDP0	page 58
$20 ($40)	UBRRHI	UBRR1[11:8]				UBRR0[11:8]				page 78
$1F ($3F)	EEARH	-	-	-	-	-	-	-	EEAR8	page 60
$1E ($3E)	EEARL	EEPROM Address Register Low Byte								page 60
$1D ($3D)	EEDR	EEPROM Data Register								page 60
$1C ($3C)	EECR	-	-	-	-	EERIE	EEMWE	EEWE	EERE	page 61
$1B ($3B)	PORTA	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	page 89
$1A ($3A)	DDRA	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	page 89
$19 ($39)	PINA	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0	page 89
$18 ($38)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	page 91
$17 ($37)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	page 91
$16 ($36)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	page 91
$15 ($35)	PORTC	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	page 98
$14 ($34)	DDRC	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	page 98
$13 ($33)	PINC	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	page 98
$12 ($32)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	page 100
$11 ($31)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	page 100
$10 ($30)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	page 100
$0F ($2F)	SPDR	SPI Data Register								page 68
$0E ($2E)	SPSR	SPIF	WCOL	-	-	-	-	-	SPI2X	page 67
$0D ($2D)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	page 66
$0C ($2C)	UDR0	UART0 I/O Data Register								page 73
$0B ($2B)	UCSR0A	RXC0	TXC0	UDRE0	FE0	OR0	-	U2X0	MPCM0	page 73
$0A ($2A)	UCSR0B	RXCIE0	TXCIE0	UDRIE0	RXEN0	TXEN0	CHR90	RXB80	TXB80	page 75
$09 ($29)	UBRR0	UART0 Baud Rate Register								page 78
$08 ($28)	ACSR	ACD	AINBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	page 81
$07 ($27)	PORTE	-	-	-	-	-	PORTE2	PORTE1	PORTE0	page 106
$06 ($26)	DDRE	-	-	-	-	-	DDE2	DDE1	DDE0	page 106
$05 ($25)	PINE	-	-	-	-	-	PINE2	PINE1	PINE0	page 106
$04 ($24)	Reserved									
$03 ($23)	UDR1	UART1 I/O Data Register								page 73
$02 ($22)	UCSR1A	RXC1	TXC1	UDRE1	FE1	OR1	-	U2X1	MPCM1	page 75
$01 ($21)	UCSR1B	RXCIE1	TXCIE1	UDRIE1	RXEN1	TXEN1	CHR91	RXB81	TXB81	page 73
$00 ($20)	UBRR1	UART1 Baud Rate Register								page 78

Instruction Set Summary (Continued)

Mnemonic	Operands	Description	Operation	Flags	# Clocks
SEZ		Set Zero Flag	Z ← 1	Z	1
CLZ		Clear Zero Flag	Z ← 0	Z	1
SEI		Global Interrupt Enable	I ← 1	I	1
CLI		Global Interrupt Disable	I ← 0	I	1
SES		Set Signed Test Flag	S ← 1	S	1
CLS		Clear Signed Test Flag	S ← 0	S	1
SEV		Set Two's Complement Overflow	V ← 1	V	1
CLV		Clear Two's Complement Overflow	V ← 0	V	1
SET		Set T in SREG	T ← 1	T	1
CLT		Clear T in SREG	T ← 0	T	1
SEH		Set Half-carry Flag in SREG	H ← 1	H	1
CLH		Clear Half-carry Flag in SREG	H ← 0	H	1
NOP		No Operation		None	1
SLEEP		Sleep	(see specific descr. for Sleep function)	None	3
WDR		Watchdog Reset	(see specific descr. for WDR/timer)	None	1