

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	4MHz
Connectivity	SPI, UART/USART
Peripherals	PWM, WDT
Number of I/O	32
Program Memory Size	4KB (2K x 16)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C
Mounting Type	Surface Mount
Package / Case	44-LCC (J-Lead)
Supplier Device Package	44-PLCC (16.6x16.6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at90s4414-4ji

ALU - Arithmetic Logic Unit

The high-performance AVR ALU operates in direct connection with all the 32 general purpose working registers. Within a single clock cycle, ALU operations between registers in the register file are executed. The ALU operations are divided into three main categories - arithmetic, logical and bit-functions.

In-System Programmable Flash Program Memory

The AT90S4414/8515 contains 4K/8K bytes on-chip In-System Programmable Flash memory for program storage. Since all instructions are 16-or 32-bit words, the Flash is organized as 2K x 16/4K x 16. The Flash memory has an endurance of at least 1000 write/erase cycles. The AT90S4414/8515 Program Counter (PC) is 11/12 bits wide, thus addressing the 2048/4096 program memory addresses.

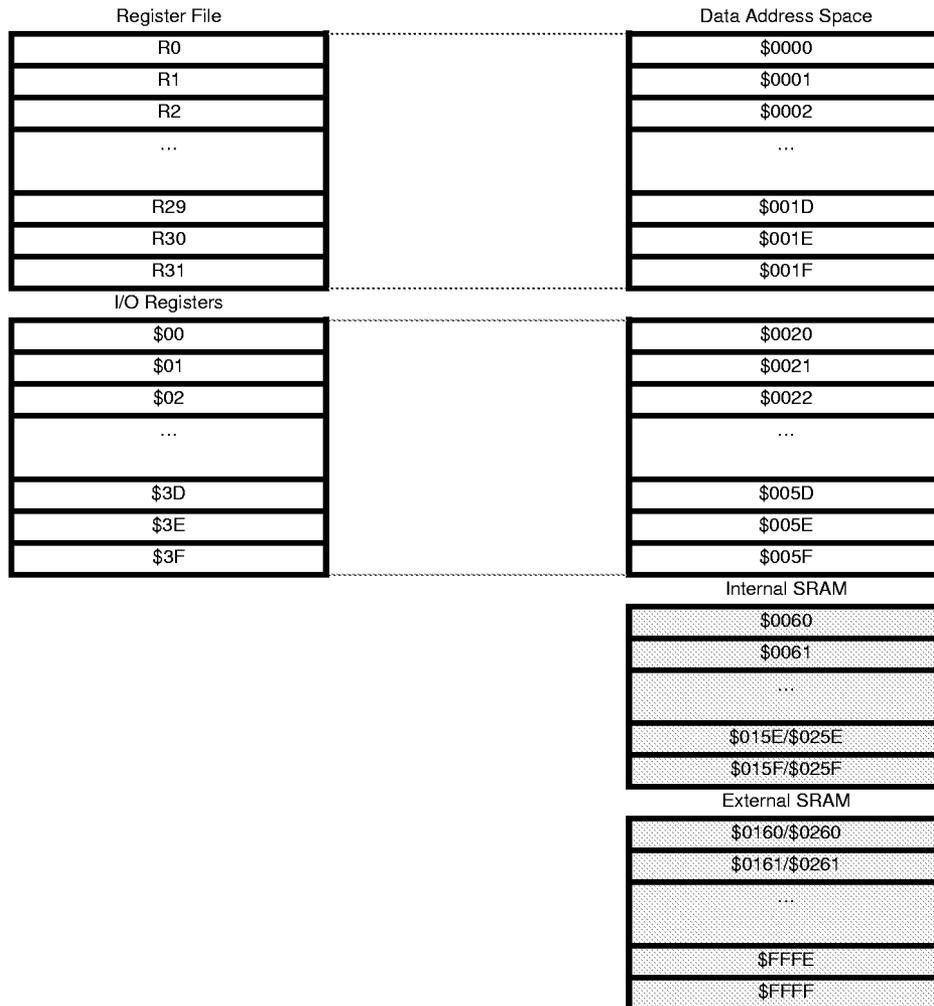
See page 77 for a detailed description on Flash data downloading.

See page 10 for the different program memory addressing modes.

SRAM Data Memory - Internal and External

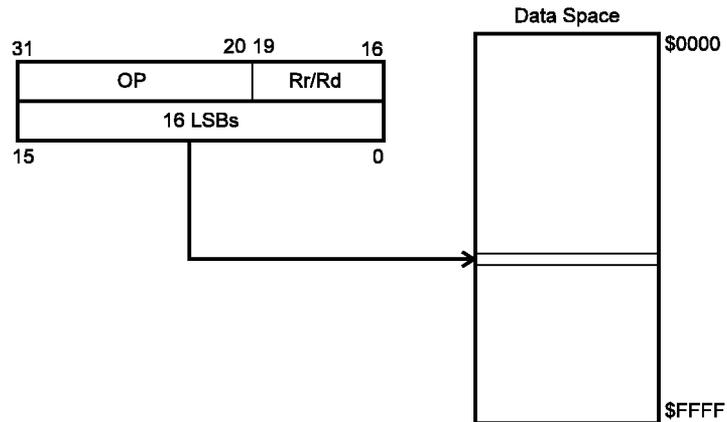
The following figure shows how the AT90S4414/8515 SRAM Memory is organized:

Figure 8. SRAM Organization



Data Direct

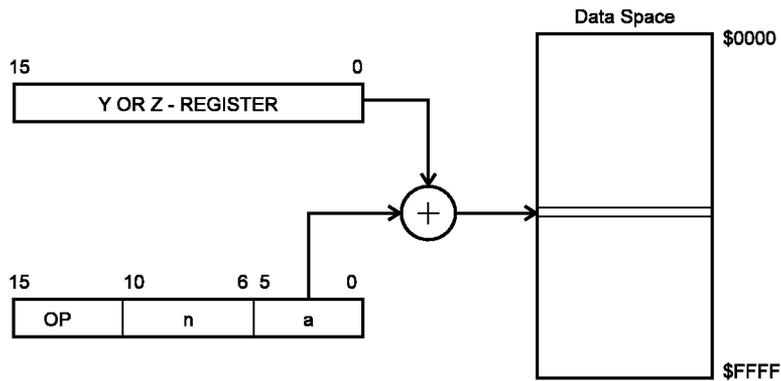
Figure 12. Direct Data Addressing



A 16-bit Data Address is contained in the 16 LSBs of a two-word instruction. Rd/Rr specify the destination or source register.

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 6 bits of the instruction word.

I/O Memory

The I/O space definition of the AT90S4414/8515 is shown in the following table:

Table 2. AT90S4414/8515 I/O Space

Address Hex	Name	Function
\$3F (\$5F)	SREG	Status Register
\$3E (\$5E)	SPH	Stack Pointer High
\$3D (\$5D)	SPL	Stack Pointer Low
\$3B (\$5B)	GIMSK	General Interrupt Mask register
\$3A (\$5A)	GIFR	General Interrupt Flag Register
\$39 (\$59)	TIMSK	Timer/Counter Interrupt Mask register
\$38 (\$58)	TIFR	Timer/Counter Interrupt Flag register
\$35 (\$55)	MCUCR	MCU general Control Register
\$33 (\$53)	TCCR0	Timer/Counter0 Control Register
\$32 (\$52)	TCNT0	Timer/Counter0 (8-bit)
\$2F (\$4F)	TCCR1A	Timer/Counter1 Control Register A
\$2E (\$4E)	TCCR1B	Timer/Counter1 Control Register B
\$2D (\$4D)	TCNT1H	Timer/Counter1 High Byte
\$2C (\$4C)	TCNT1L	Timer/Counter1 Low Byte
\$2B (\$4B)	OCR1AH	Timer/Counter1 Output Compare Register A High Byte
\$2A (\$4A)	OCR1AL	Timer/Counter1 Output Compare Register A Low Byte
\$29 (\$49)	OCR1BH	Timer/Counter1 Output Compare Register B High Byte
\$28 (\$48)	OCR1BL	Timer/Counter1 Output Compare Register B Low Byte
\$25 (\$45)	ICR1H	T/C 1 Input Capture Register High Byte
\$24 (\$44)	ICR1L	T/C 1 Input Capture Register Low Byte
\$21 (\$41)	WDTCR	Watchdog Timer Control Register
\$1F (\$3E)	EEARH	EEPROM Address Register High Byte (AT90S8515)
\$1E (\$3E)	EEARL	EEPROM Address Register Low Byte
\$1D (\$3D)	EEDR	EEPROM Data Register
\$1C (\$3C)	EEDR	EEPROM Control Register
\$1B (\$3B)	PORTA	Data Register, Port A
\$1A (\$3A)	DDRA	Data Direction Register, Port A
\$19 (\$39)	PINA	Input Pins, Port A
\$18 (\$38)	PORTB	Data Register, Port B
\$17 (\$37)	DDRB	Data Direction Register, Port B
\$16 (\$36)	PINB	Input Pins, Port B
\$15 (\$35)	PORTC	Data Register, Port C
\$14 (\$34)	DDRC	Data Direction Register, Port C
\$13 (\$33)	PINC	Input Pins, Port C



Table 2. AT90S4414/8515 I/O Space (Continued)

Address Hex	Name	Function
\$12 (\$32)	PORTD	Data Register, Port D
\$11 (\$31)	DDRD	Data Direction Register, Port D
\$10 (\$30)	PIND	Input Pins, Port D
\$0F (\$2F)	SPDR	SPI I/O Data Register
\$0E (\$2E)	SPSR	SPI Status Register
\$0D (\$2D)	SPCR	SPI Control Register
\$0C (\$2C)	UDR	UART I/O Data Register
\$0B (\$2B)	USR	UART Status Register
\$0A (\$2A)	UCR	UART Control Register
\$09 (\$29)	UBRR	UART Baud Rate Register
\$08 (\$28)	ACSR	Analog Comparator Control and Status Register

Note: Reserved and unused locations are not shown in the table

All AT90S4414/8515 I/Os and peripherals are placed in the I/O space. The I/O locations are accessed by the IN and OUT instructions transferring data between the 32 general purpose working registers and the I/O space. I/O registers within the address range \$00 - \$1F are directly bit-accessible using the SBI and CBI instructions. In these registers, the value of single bits can be checked by using the SBIS and SBIC instructions. Refer to the instruction set chapter for more details. When using the I/O specific commands IN, OUT the I/O addresses \$00 - \$3F must be used. When addressing I/O registers as SRAM, \$20 must be added to this address. All I/O register addresses throughout this document are shown with the SRAM address in parentheses.

For compatibility with future devices, reserved bits should be written to zero if accessed. Reserved I/O memory addresses should never be written.

Some of the status flags are cleared by writing a logical one to them. Note that the CBI and SBI instructions will operate on all bits in the I/O register, writing a one back into any flag read as set, thus clearing the flag. The CBI and SBI instructions work with registers \$00 to \$1F only.

The I/O and peripherals control registers are explained in the following chapters.

Status Register - SREG

The AVR status register - SREG - at I/O space location \$3F (\$5F) is defined as:

Bit	7	6	5	4	3	2	1	0	
\$3F (\$5F)	I	T	H	S	V	N	Z	C	SREG
Read/Write	R/W								
Initial value	0	0	0	0	0	0	0	0	

- **Bit 7 - I: Global Interrupt Enable**

The global interrupt enable bit must be set (one) for the interrupts to be enabled. The individual interrupt enable control is then performed in separate control registers. If the global interrupt enable bit is cleared (zero), none of the interrupts are enabled independent of the individual interrupt enable settings. The I-bit is cleared by hardware after an interrupt has occurred, and is set by the RETI instruction to enable subsequent interrupts.

- **Bit 6 - T: Bit Copy Storage**

The bit copy instructions BLD (Bit Load) and BST (Bit Store) use the T bit as source and destination for the operated bit. A bit from a register in the register file can be copied into T by the BST instruction, and a bit in T can be copied into a bit in a register in the register file by the BLD instruction.

Table 3. Reset and Interrupt Vectors

Vector No.	Program Address	Source	Interrupt Definition
1	\$000	RESET	External Reset, Power-on Reset and Watchdog Reset
2	\$001	INT0	External Interrupt Request 0
3	\$002	INT1	External Interrupt Request 1
4	\$003	TIMER1 CAPT	Timer/Counter1 Capture Event
5	\$004	TIMER1 COMPA	Timer/Counter1 Compare Match A
6	\$005	TIMER1 COMPB	Timer/Counter1 Compare Match B
7	\$006	TIMER1 OVF	Timer/Counter1 Overflow
8	\$007	TIMER0, OVF	Timer/Counter0 Overflow
9	\$008	SPI, STC	Serial Transfer Complete
10	\$009	UART, RX	UART, Rx Complete
11	\$00A	UART, UDRE	UART Data Register Empty
12	\$00B	UART, TX	UART, Tx Complete
13	\$00C	ANA_COMP	Analog Comparator

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

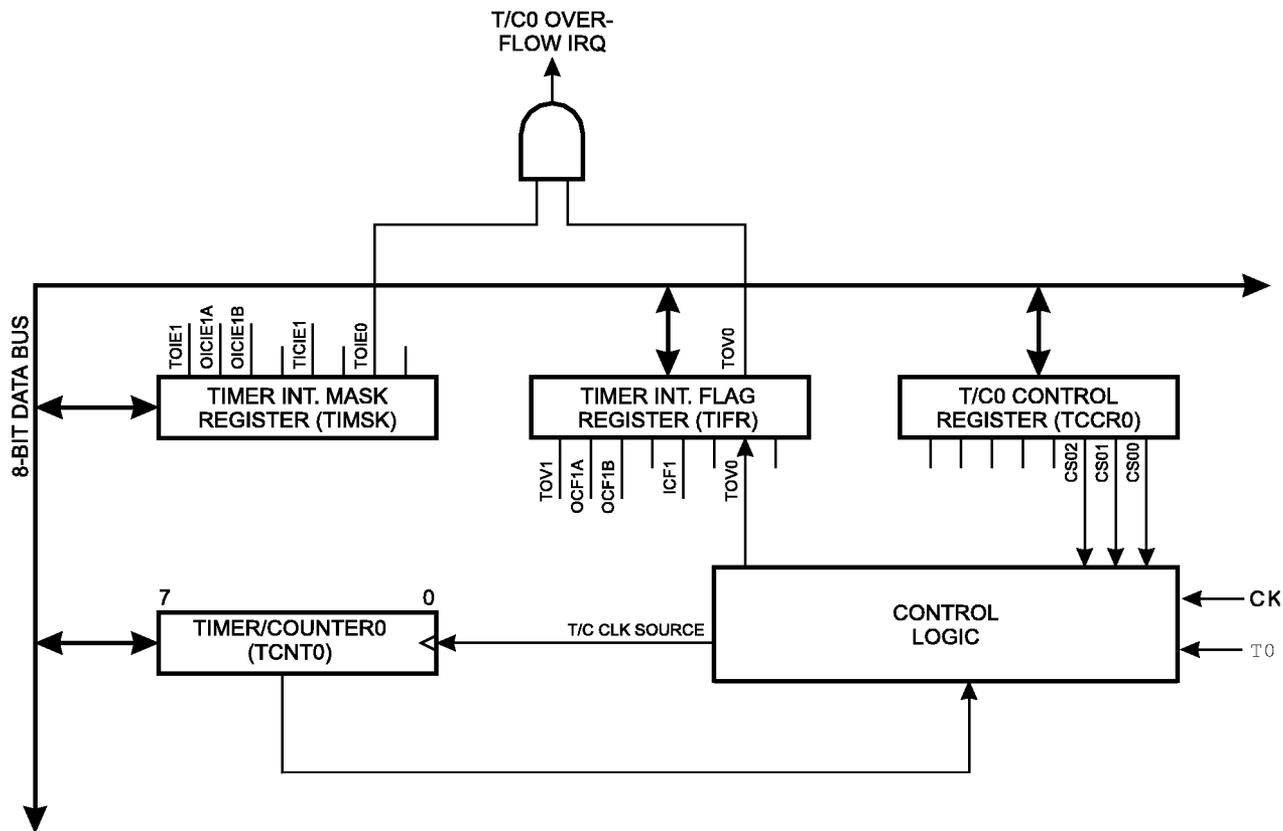
```

Address      Labels      Code      Comments
$000
$001
$002
$003
$004
$005
$006
$007
$008
$009
$00a
$00b
$00c
;
$00d      MAIN:      ldi r16,high(RAMEND); Main program start
$00e
$00f
$010
$011
<instr> xxx

...

```

Figure 29. Timer/Counter0 Block Diagram



Timer/Counter0 Control Register - TCCR0

Bit	7	6	5	4	3	2	1	0	
\$33 (\$53)	-	-	-	-	-	CS02	CS01	CS00	TCCR0
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

• **Bits 7..3 - Res: Reserved bits**

These bits are reserved bits in the AT90S4414/8515 and always read as zero.

• **Bits 2,1,0 - CS02, CS01, CS00: Clock Select0, bit 2,1 and 0**

The Clock Select0 bits 2,1 and 0 define the prescaling source of Timer/Counter0.

Table 8. Clock 0 Prescale Select

CS02	CS01	CS00	Description
0	0	0	Stop, the Timer/Counter0 is stopped.
0	0	1	CK
0	1	0	CK/8
0	1	1	CK/64
1	0	0	CK/256
1	0	1	CK/1024
1	1	0	External Pin T0, falling edge
1	1	1	External Pin T0, rising edge

Table 9. Compare 1 Mode Select

COM1X1	COM1X0	Description
0	0	Timer/Counter1 disconnected from output pin OC1X
0	1	Toggle the OC1X output line.
1	0	Clear the OC1X output line (to zero).
1	1	Set the OC1X output line (to one).

Note: X = A or B

In PWM mode, these bits have a different function. Refer to Table 13 for a detailed description.

• **Bits 3..2 - Res: Reserved bits**

These bits are reserved bits in the AT90S4414/8515 and always read zero.

• **Bits 1..0 - PWM11, PWM10: Pulse Width Modulator Select Bits 1 and 0**

These bits select PWM operation of Timer/Counter1 as specified in Table 10. This mode is described on page 36.

Table 10. PWM Mode Select

PWM11	PWM10	Description
0	0	PWM operation of Timer/Counter1 is disabled
0	1	Timer/Counter1 is an 8-bit PWM
1	0	Timer/Counter1 is a 9-bit PWM
1	1	Timer/Counter1 is a 10-bit PWM

Timer/Counter1 Control Register B - TCCR1B

Bit	7	6	5	4	3	2	1	0	
\$2E (\$4E)	ICNC1	ICES1	-	-	CTC1	CS12	CS11	CS10	TCCR1B
Read/Write	R/W	R/W	R	R	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

• **Bit 7 - ICNC1: Input Capture1 Noise Canceler (4 CKs)**

When the ICNC1 bit is cleared (zero), the input capture trigger noise canceler function is disabled. The input capture is triggered at the first rising/falling edge sampled on the ICP - input capture pin - as specified. When the ICNC1 bit is set (one), four successive samples are measures on the ICP - input capture pin, and all samples must be high/low according to the input capture trigger specification in the ICES1 bit. The actual sampling frequency is XTAL clock frequency.

• **Bit 6 - ICES1: Input Capture1 Edge Select**

While the ICES1 bit is cleared (zero), the Timer/Counter1 contents are transferred to the Input Capture Register - ICR1 - on the falling edge of the input capture pin - ICP. While the ICES1 bit is set (one), the Timer/Counter1 contents are transferred to the Input Capture Register - ICR1 - on the rising edge of the input capture pin - ICP.

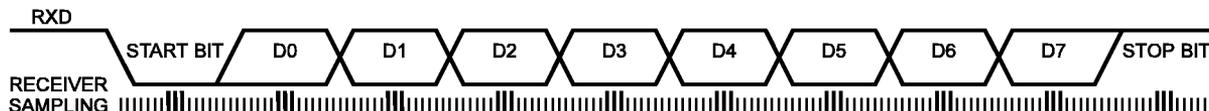
• **Bits 5, 4 - Res: Reserved bits**

These bits are reserved bits in the AT90S4414/8515 and always read zero.

The receiver front-end logic samples the signal on the RXD pin at a frequency 16 times the baud rate. While the line is idle, one single sample of logical zero 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 RXD pin at samples 8, 9 and 10. If two or more of these three samples are found to be logical ones, 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 40.

Figure 40. Sampling Received Data



When the stop bit enters the receiver, the majority of the three samples must be one to accept the stop bit. If two or more samples are logical zeros, the Framing Error (FE) flag in the UART Status Register (USR) is set. Before reading the UDR register, the user should always check the FE bit to detect Framing Errors.

Whether or not a valid stop bit is detected at the end of a character reception cycle, the data is transferred to UDR and the RXC flag in USR is set. UDR is in fact two physically separate registers, one for transmitted data and one for received data. When UDR is read, the Receive Data register is accessed, and when UDR is written, the Transmit Data register is accessed. If 9 bit data word is selected (the CHR9 bit in the UART Control Register, UCR is set), the RXB8 bit in UCR is loaded with bit 9 in the Transmit shift register when data is transferred to UDR.

If, after having received a character, the UDR register has not been read since the last receive, the OverRun (OR) flag in UCR is set. This means that the last data byte shifted into to the shift register could not be transferred to UDR and has been lost. The OR bit is buffered, and is updated when the valid data byte in UDR is read. Thus, the user should always check the OR bit after reading the UDR register in order to detect any overruns if the baud rate is high or CPU load is high.

When the RXEN bit in the UCR register is cleared (zero), the receiver is disabled. This means that the PD0 pin can be used as a general I/O pin. When RXEN is set, the UART Receiver will be connected to PD0, which is forced to be an input pin regardless of the setting of the DDD0 bit in DDRD. When PD0 is forced to input by the UART, the PORTD0 bit can still be used to control the pull-up resistor on the pin.

When the CHR9 bit in the UCR register is set, transmitted and received characters are 9-bit long plus start and stop bits. The 9th data bit to be transmitted is the TXB8 bit in UCR register. This bit must be set to the wanted value before a transmission is initiated by writing to the UDR register. The 9th data bit received is the RXB8 bit in the UCR register.

UART Control

UART I/O Data Register - UDR

Bit	7	6	5	4	3	2	1	0	
\$0C (\$2C)	MSB							LSB	UDR
Read/Write	R/W								
Initial value	0	0	0	0	0	0	0	0	

The UDR register is actually two physically separate registers sharing the same I/O address. When writing to the register, the UART Transmit Data register is written. When reading from UDR, the UART Receive Data register is read.

- **Bit 2 - ACIC: Analog Comparator Input Capture Enable**

When set (one), this bit enables the Input Capture function in Timer/Counter1 to be triggered by the analog comparator. The comparator output is in this case directly connected to the Input Capture front-end logic, making the comparator utilize the noise canceler and edge select features of the Timer/Counter1 Input Capture interrupt. When cleared (zero), no connection between the analog comparator and the Input Capture function is given. To make the comparator trigger the Timer/Counter1 Input Capture interrupt, the TICIE1 bit in the Timer Interrupt Mask Register (TIMSK) must be set (one).

- **Bits 1,0 - ACIS1, ACIS0: Analog Comparator Interrupt Mode Select**

These bits determine which comparator events that trigger the Analog Comparator interrupt. The different settings are shown in Table 19.

Table 19. ACIS1/ACIS0 Settings

ACIS1	ACIS0	Interrupt Mode
0	0	Comparator Interrupt on Output Toggle
0	1	Reserved
1	0	Comparator Interrupt on Falling Output Edge
1	1	Comparator Interrupt on Rising Output Edge

Note: When changing the ACIS1/ACIS0 bits, The Analog Comparator Interrupt must be disabled by clearing its Interrupt Enable bit in the ACSR register. Otherwise an interrupt can occur when the bits are changed.

Interface to External SRAM

The interface to the SRAM consists of:

- Port A: Multiplexed low-order address bus and data bus
- Port C: High-order address bus
- The ALE-pin: Address latch enable
- The \overline{RD} and \overline{WR} -pin: Read and write strobes.

The external data SRAM is enabled by setting the SRE - External SRAM enable bit of the MCUCR - MCU control register, and will override the setting of the data direction register DDRA. When the SRE bit is cleared (zero), the external data SRAM is disabled, and the normal pin and data direction settings are used. When SRE is cleared (zero), the address space above the internal SRAM boundary is not mapped into the internal SRAM, as in AVR parts not having interface to the external SRAM.

When ALE goes from high to low, there is a valid address on Port A. ALE is low during a data transfer. \overline{RD} and \overline{WR} are active when accessing the external SRAM only.

When the external SRAM is enabled, the ALE signal may have short pulses when accessing the internal RAM, but the ALE signal is stable when accessing the external SRAM.

Figure 42 sketches how to connect an external SRAM to the AVR using 8 latches which are transparent when G is high.

Default, the external SRAM access is a three-cycle scheme as depicted in Figure 43. When one extra wait state is needed in the access cycle, set the SRW bit (one) in the MCUCR register. The resulting access scheme is shown in Figure 44. In both cases, note that PORTA is data bus in one cycle only. As soon as the data access finishes, PORTA becomes a low order address bus again.

For details in the timing for the SRAM interface, please refer to Figure 68, Table 38, Table 39, Table 40, and Table 41 in section "Absolute Maximum Ratings*" on page 81.

Figure 50. Port B Schematic Diagram (Pin PB6)

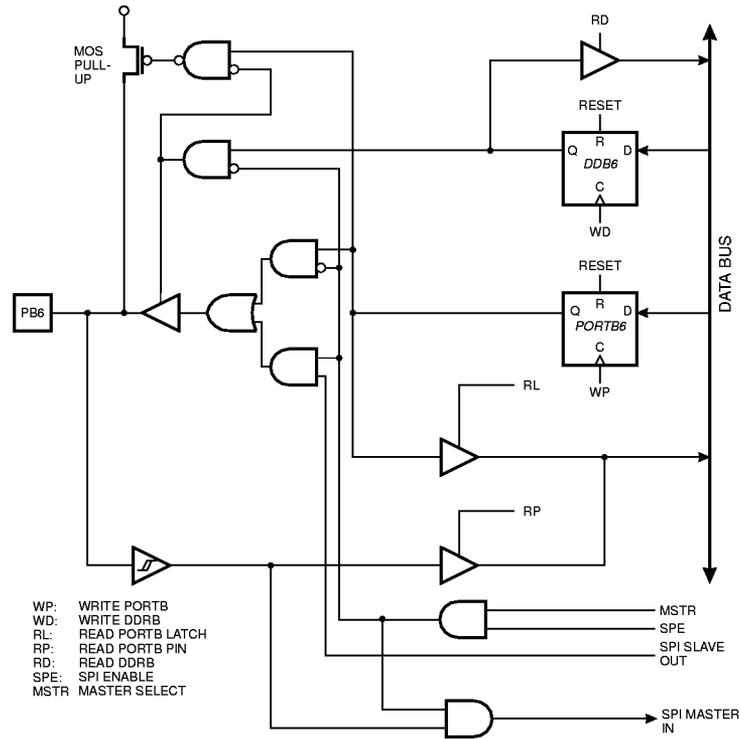
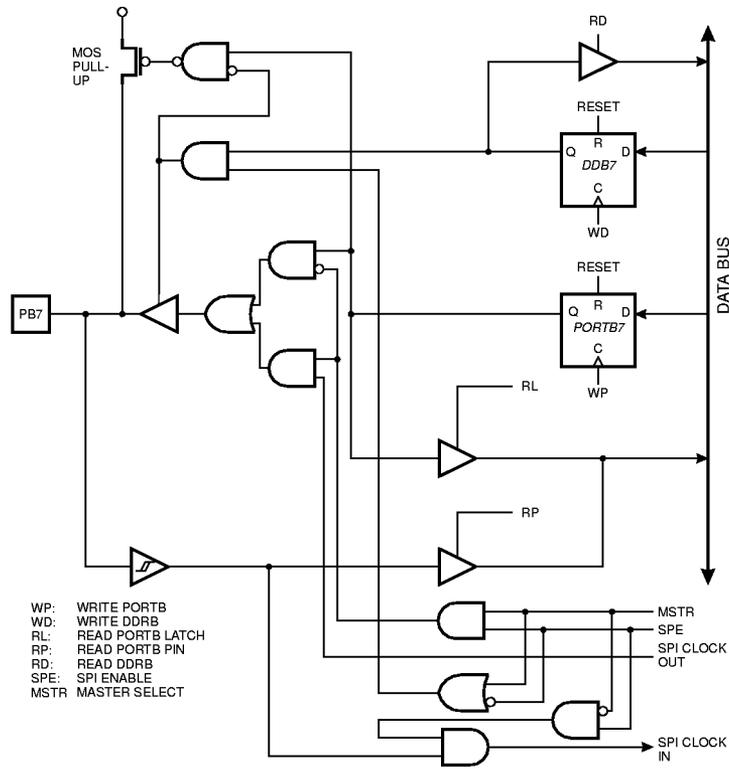


Figure 51. Port B Schematic Diagram (Pin PB7)



Port C

Port C is an 8-bit bi-directional I/O port. Three I/O memory address locations are allocated for the Port C, one each for the Data Register - PORTC, \$15(\$35), Data Direction Register - DDRC, \$14(\$34) and the Port C Input Pins - PINC, \$13(\$33). The Port C 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 C output buffers can sink 20mA and thus drive LED displays directly. When pins PC0 to PC7 are used as inputs and are externally pulled low, they will source current if the internal pull-up resistors are activated.

The Port C pins have alternate functions related to the optional external data SRAM. Port C can be configured to be the high-order address byte during accesses to external data memory. When Port C is set to the alternate function by the SRE - External SRAM Enable - bit in the MCUCR - MCU Control Register, the alternate settings override the data direction register.

Port C Data Register - PORTC

Bit	7	6	5	4	3	2	1	0	
\$15 (\$35)	PORTC7 PORTC6 PORTC5 PORTC4 PORTC3 PORTC2 PORTC1 PORTC0								PORTC
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 C Data Direction Register - DDRC

Bit	7	6	5	4	3	2	1	0	
\$14 (\$34)	DDC7 DDC6 DDC5 DDC4 DDC3 DDC2 DDC1 DDC0								DDRC
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 C Input Pins Address - PINC

Bit	7	6	5	4	3	2	1	0	
\$13 (\$33)	PINC7 PINC6 PINC5 PINC4 PINC3 PINC2 PINC1 PINC0								PINC
Read/Write	R	R	R	R	R	R	R	R	
Initial value	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	

The Port C Input Pins address - PINC - is not a register, and this address enables access to the physical value on each Port C pin. When reading PORTC, the Port C Data Latch is read, and when reading PINC, the logical values present on the pins are read.

PortC as General Digital I/O

All 8 pins in Port C have equal functionality when used as digital I/O pins.

PCn, General I/O pin: The DDCn bit in the DDRC register selects the direction of this pin, if DDCn is set (one), PCn is configured as an output pin. If DDCn is cleared (zero), PCn is configured as an input pin. If PORTCn is set (one) when the pin configured as an input pin, the MOS pull up resistor is activated. To switch the pull up resistor off, PORTCn has to be cleared (zero) or the pin has to be configured as an output pin. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not active.

Table 23. DDCn Effects on Port C Pins

DDCn	PORTCn	I/O	Pull up	Comment
0	0	Input	No	Tri-state (Hi-Z)
0	1	Input	Yes	PCn will source current if ext. pulled low.
1	0	Output	No	Push-Pull Zero Output
1	1	Output	No	Push-Pull One Output

n: 7...0, pin number

Figure 56. Port D Schematic Diagram (Pin PD4)

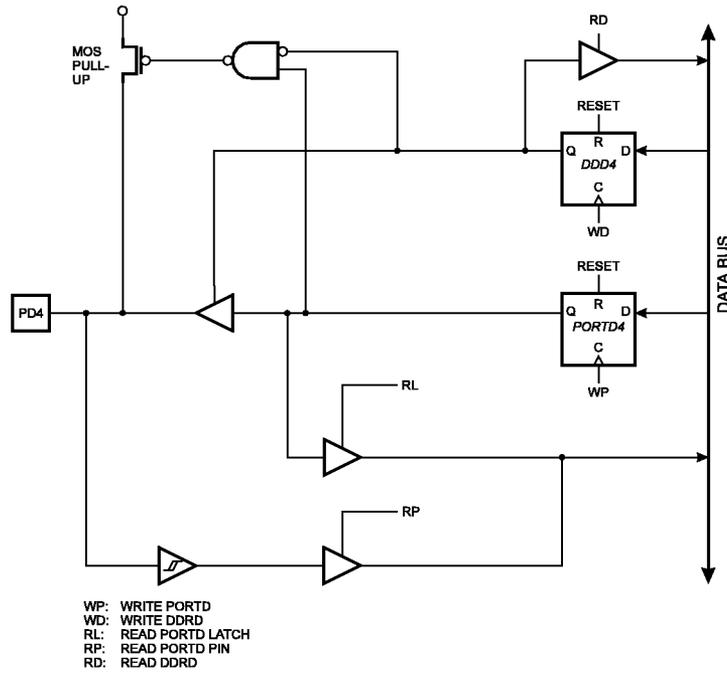
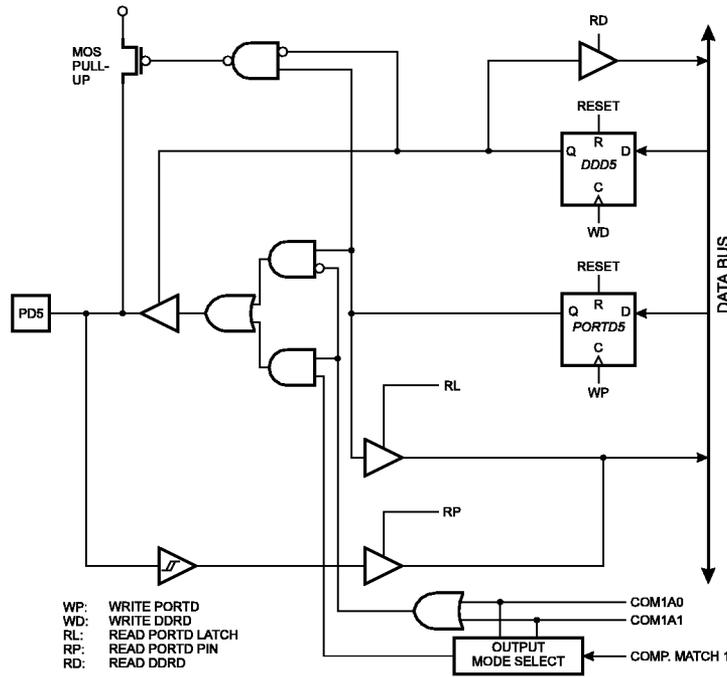


Figure 57. Port D Schematic Diagram (Pin PD5)



3. Wait until RDY/ $\overline{\text{BSY}}$ goes high to program the next byte.

(See Figure 62 for signal waveforms.)

The loaded command and address are retained in the device during programming. For efficient programming, the following should be considered.

- The command needs only be loaded once when writing or reading multiple memory locations.
- Address high byte needs only be loaded before programming a new 256 word page in the Flash.
- Skip writing the data value \$FF, that is the contents of the entire Flash and EEPROM after a Chip Erase.

These considerations also applies to EEPROM programming, and Flash, EEPROM and Signature bytes reading.

Figure 61. Programming the Flash Waveforms

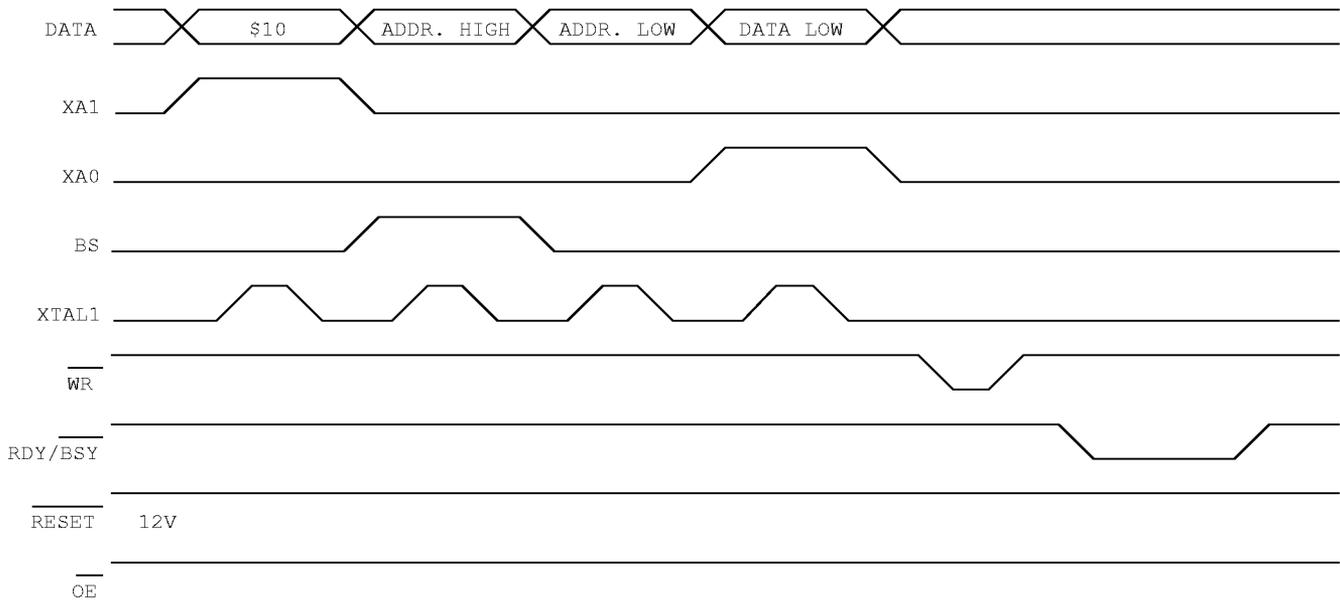
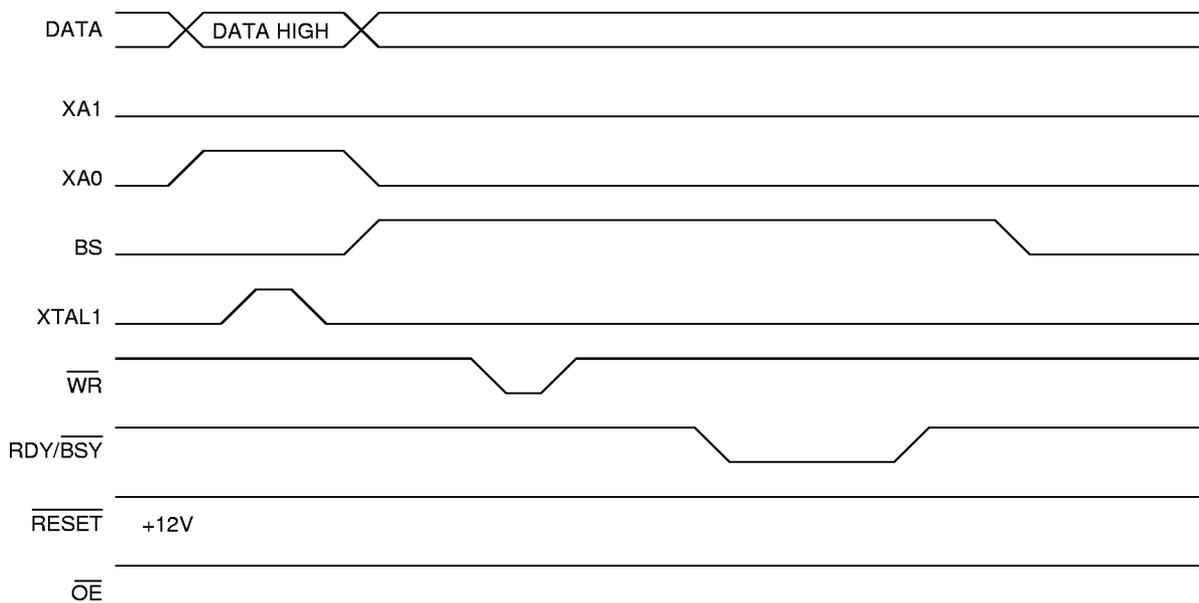


Figure 62. Programming the Flash Waveforms (continued)



Parallel Programming Characteristics

Figure 63. Parallel Programming Timing

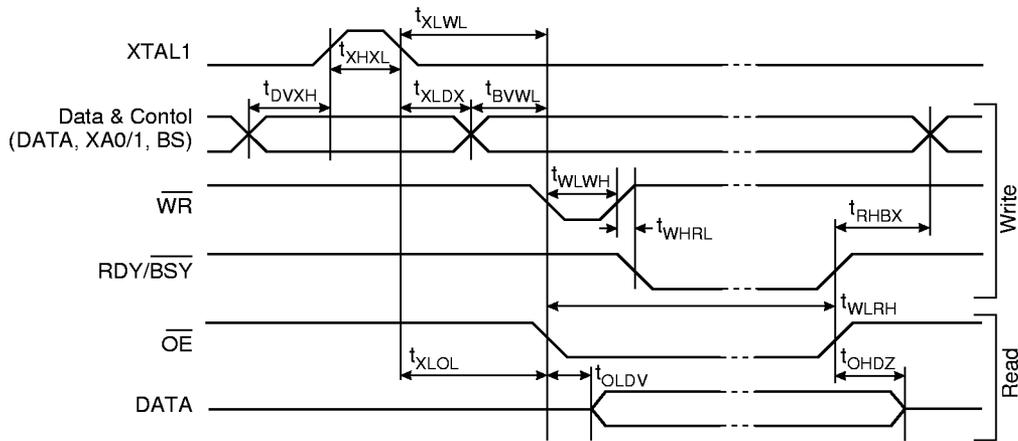


Table 31. Parallel Programming Characteristics $T_A = 25^\circ\text{C} \pm 10\%$, $V_{CC} = 5\text{V} \pm 10\%$

Symbol	Parameter	Min	Typ	Max	Units
V_{PP}	Programming Enable Voltage	11.5		12.5	V
I_{PP}	Programming Enable Current			250	μA
t_{DVXH}	Data and Control Setup before XTAL1 High	67			ns
t_{XHXL}	XTAL1 Pulse Width High	67			ns
t_{XLDX}	Data and Control Hold after XTAL1 Low	67			ns
t_{XLWL}	XTAL1 Low to \overline{WR} Low	67			ns
t_{BVWL}	BS Valid to \overline{WR} Low	67			ns
t_{RHBX}	BS Hold after RDY/BSY High	67			ns
t_{WLWH}	\overline{WR} Pulse Width Low ⁽¹⁾	67			ns
t_{WHRL}	\overline{WR} High to RDY/BSY Low ⁽²⁾		20		ns
t_{WLRH}	\overline{WR} Low to RDY/BSY High ⁽²⁾	0.5	0.7	0.9	ms
t_{XLOL}	XTAL1 Low to \overline{OE} Low	67			ns
t_{OLDV}	\overline{OE} Low to DATA Valid		20		ns
t_{OHDZ}	\overline{OE} High to DATA Tri-stated			20	ns
t_{WLWH_CE}	\overline{WR} Pulse Width Low for Chip Erase	5	10	15	ms
t_{WLWH_PFB}	\overline{WR} Pulse Width Low for Programming the Fuse Bits	1.0	1.5	1.8	ms

- Notes:
1. Use t_{WLWH_CE} for Chip Erase and t_{WLWH_PFB} for Programming the Fuse Bits.
 2. If t_{WLWH} is held longer than t_{WLRH} , no RDY/BSY pulse will be seen.

Serial Programming Algorithm

When writing serial data to the AT90S4414/8515, data is clocked on the rising edge of SCK.

When reading data from the AT90S4414/8515, data is clocked on the falling edge of SCK. See Figure 65, Figure 66 and Table 34 for timing details.

To program and verify the AT90S4414/8515 in the serial programming mode, the following sequence is recommended (See four byte instruction formats in Table 33):

1. Power-up sequence:

Apply power between V_{CC} and GND while \overline{RESET} and SCK are set to '0'. If a crystal is not connected across pins XTAL1 and XTAL2, apply a clock signal to the XTAL1 pin. In some systems, the programmer can not guarantee that SCK is held low during power-up. In this case, \overline{RESET} must be given a positive pulse of at least two XTAL1 cycles duration after SCK has been set to '0'.

2. Wait for at least 20 ms and enable serial programming by sending the Programming Enable serial instruction to the MOSI (PB5) pin.

3. The serial programming instructions will not work if the communication is out of synchronization. When in sync, the second byte (\$53) will echo back when issuing the third byte of the Programming Enable instruction. Whether the echo is correct or not, all 4 bytes of the instruction must be transmitted. If the \$53 did not echo back, give SCK a positive pulse and issue a new Programming Enable instruction. If the \$53 is not seen within 32 attempts, there is no functional device connected.

4. If a Chip Erase is performed (must be done to erase the Flash), wait t_{WD_ERASE} after the instruction, give \overline{RESET} a positive pulse, and start over from Step 2. See Table 35 on page 80 for t_{WD_ERASE} value.

5. The Flash or EEPROM array is programmed one byte at a time by supplying the address and data together with the appropriate Write instruction. An EEPROM memory location is first automatically erased before new data is written. Use Data Polling to detect when the next byte in the Flash or EEPROM can be written. If polling is not used, wait t_{WD_PROG} before transmitting the next instruction. See Table 36 on page 80 for t_{WD_PROG} value. In an erased device, no \$FFs in the data file(s) needs to be programmed.

6. Any memory location can be verified by using the Read instruction which returns the content at the selected address at the serial output MISO (PB6) pin.

7. At the end of the programming session, \overline{RESET} can be set high to commence normal operation.

8. Power-off sequence (if needed):

Set XTAL1 to "0" (if a crystal is not used).

Set \overline{RESET} to "1".

Turn V_{CC} power off

Data Polling EEPROM

When a byte is being programmed into the EEPROM, reading the address location being programmed will give the value P1 until the auto-erase is finished, and then the value P2. See Table 32 for P1 and P2 values.

At the time the device is ready for a new EEPROM byte, the programmed value will read correctly. This is used to determine when the next byte can be written. This will not work for the values P1 and P2, so when programming these values, the user will have to wait for at least the prescribed time t_{WD_PROG} before programming the next byte. See Table 35 for t_{WD_PROG} value. As a chip-erased device contains \$FF in all locations, programming of addresses that are meant to contain \$FF, can be skipped. This does not apply if the EEPROM is reprogrammed without first chip-erasing the device.

Table 32. Read Back Value during EEPROM Polling

Part	P1	P2
AT90S4414	\$80	\$7F
AT90S8515	\$80	\$7F

Data Polling Flash

When a byte is being programmed into the Flash, reading the address location being programmed will give the value \$7F. At the time the device is ready for a new byte, the programmed value will read correctly. This is used to determine when the next byte can be written. This will not work for the value \$7F, so when programming this value, the user will have to wait for at least t_{WD_PROG} before programming the next byte. As a chip-erased device contains \$FF in all locations, programming of addresses that are meant to contain \$FF, can be skipped.

Figure 65. Serial Programming Waveforms

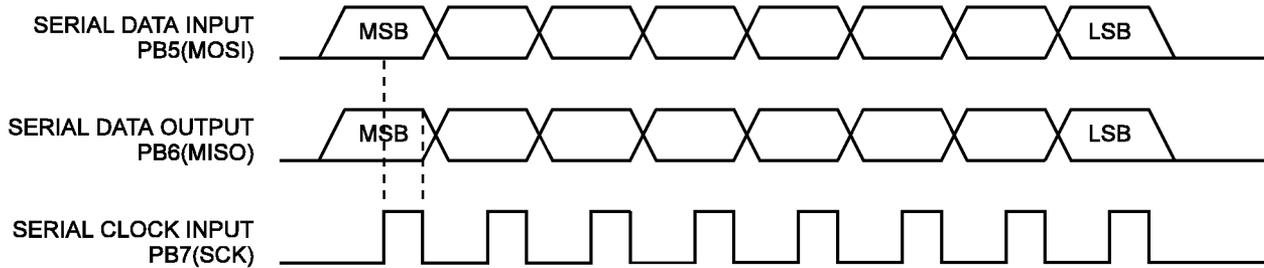


Table 33. Serial Programming Instruction Set

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 while RESET is low.
Chip Erase	1010 1100	100x xxxx	xxxx xxxx	xxxx xxxx	Chip Erase Flash and EEPROM memory arrays.
Read Program Memory	0010 H000	xxxx aaaa	bbbb bbbb	oooo oooo	Read H (high or low) data o from Program memory at word address a:b.
Write Program Memory	0100 H000	xxxx aaaa	bbbb bbbb	iiii iiii	Write H (high or low) data I to Program memory at word address a:b.
Read EEPROM Memory	1010 0000	xxxx xxxa	bbbb bbbb	oooo oooo	Read data o from EEPROM memory at address a:b.
Write EEPROM Memory	1100 0000	xxxx xxxa	bbbb bbbb	iiii iiii	Write data I to EEPROM memory at address a:b.
Write Lock Bits	1010 1100	111x x21x	xxxx xxxx	xxxx xxxx	Write Lock bits. Set bits 1,2='0' to program Lock bits.
Read Signature Bytes	0011 0000	xxxx xxxx	xxxx xxbb	oooo oooo	Read Signature Byte o at address b. ⁽¹⁾

Note: 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

Note: 1. The signature bytes are not readable in Lock mode 3, i.e. both Lock bits programmed.

External Data Memory Timing

Table 38. External Data Memory Characteristics, 4.0 - 6.0 Volts, No Wait State

	Symbol	Parameter	8 MHz Oscillator		Variable Oscillator		Unit
			Min	Max	Min	Max	
0	$1/t_{CLCL}$	Oscillator Frequency			0.0	8.0	MHz
1	t_{LHLL}	ALE Pulse Width	32.5		$0.5t_{CLCL}-30.0^{(1)}$		ns
2	t_{AVLL}	Address Valid A to ALE Low	22.5		$0.5t_{CLCL}-40.0^{(1)}$		ns
3a	t_{LLAX_ST}	Address Hold After ALE Low, ST/STD/STS Instructions	67.5		$0.5t_{CLCL}+5.0^{(2)}$		ns
3b	t_{LLAX_LD}	Address Hold after ALE Low, LD/LDD/LDS Instructions	15.0		15.0		ns
4	t_{AVLLC}	Address Valid C to ALE Low	22.5		$0.5t_{CLCL}-40.0^{(1)}$		ns
5	t_{AVRL}	Address Valid to RD Low	95.0		$1.0t_{CLCL}-30.0$		ns
6	t_{AVWL}	Address Valid to WR Low	157.5		$1.5t_{CLCL}-30.0^{(1)}$		ns
7	t_{LLWL}	ALE Low to WR Low	105.0	145	$1.0t_{CLCL}-20.0$	$1.0t_{CLCL}+20.0$	ns
8	t_{LLRL}	ALE Low to RD Low	42.5	82.5	$0.5t_{CLCL}-20.0^{(2)}$	$0.5t_{CLCL}+20.0^{(2)}$	ns
9	t_{DVRH}	Data Setup to RD High	60.0		60.0		ns
10	t_{RLDV}	Read Low to Data Valid		70.0		$1.0t_{CLCL}-55.0$	ns
11	t_{RHDX}	Data Hold After RD High	0.0		0.0		ns
12	t_{RLRH}	RD Pulse Width	105.0		$1.0t_{CLCL}-20.0$		ns
13	t_{DVWL}	Data Setup to WR Low	27.5		$0.5t_{CLCL}-35.0^{(2)}$		ns
14	t_{WHDX}	Data Hold After WR High	0.0		0.0		ns
15	t_{DVWH}	Data Valid to WR High	95.0		$1.0t_{CLCL}-30.0$		ns
16	t_{WLWH}	WR Pulse Width	42.5		$0.5t_{CLCL}-20.0^{(1)}$		ns

Table 39. External Data Memory Characteristics, 4.0 - 6.0 Volts, 1 Cycle Wait State

	Symbol	Parameter	8 MHz Oscillator		Variable Oscillator		Unit
			Min	Max	Min	Max	
0	$1/t_{CLCL}$	Oscillator Frequency			0.0	8.0	MHz
10	t_{RLDV}	Read Low to Data Valid		195.0		$2.0t_{CLCL}-55.0$	ns
12	t_{RLRH}	RD Pulse Width	230.0		$2.0t_{CLCL}-20.0$		ns
15	t_{DVWH}	Data Valid to WR High	220.0		$2.0t_{CLCL}-30.0$		ns
16	t_{WLWH}	WR Pulse Width	167.5		$1.5t_{CLCL}-20.0^{(2)}$		ns

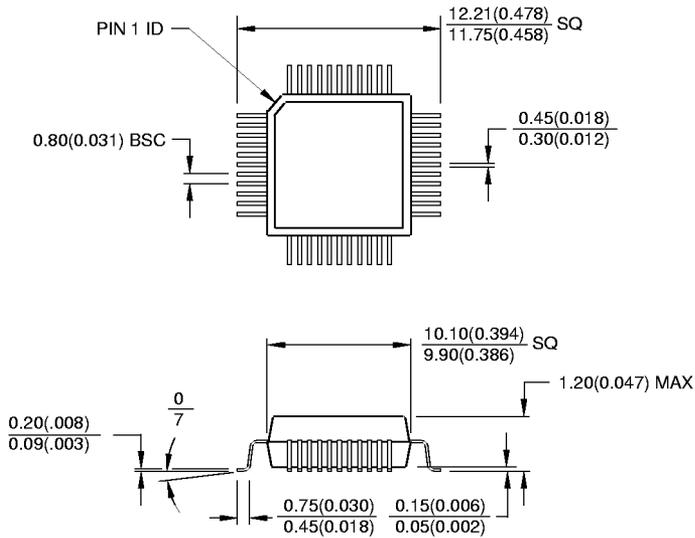
Notes: 1. This assumes 50% clock duty cycle. The half period is actually the high time of the external clock, XTAL1.
 2. This assumes 50% clock duty cycle. The half period is actually the low time of the external clock, XTAL1.

Instruction Set Summary (Continued)

Mnemonics	Operands	Description	Operation	Flags	#Clocks
DATA TRANSFER INSTRUCTIONS					
MOV	Rd, Rr	Move Between Registers	$Rd \leftarrow Rr$	None	1
LDI	Rd, K	Load Immediate	$Rd \leftarrow K$	None	1
LD	Rd, X	Load Indirect	$Rd \leftarrow (X)$	None	2
LD	Rd, X+	Load Indirect and Post-Inc.	$Rd \leftarrow (X), X \leftarrow X + 1$	None	2
LD	Rd, -X	Load Indirect and Pre-Dec.	$X \leftarrow X - 1, Rd \leftarrow (X)$	None	2
LD	Rd, Y	Load Indirect	$Rd \leftarrow (Y)$	None	2
LD	Rd, Y+	Load Indirect and Post-Inc.	$Rd \leftarrow (Y), Y \leftarrow Y + 1$	None	2
LD	Rd, -Y	Load Indirect and Pre-Dec.	$Y \leftarrow Y - 1, Rd \leftarrow (Y)$	None	2
LDD	Rd, Y+q	Load Indirect with Displacement	$Rd \leftarrow (Y + q)$	None	2
LD	Rd, Z	Load Indirect	$Rd \leftarrow (Z)$	None	2
LD	Rd, Z+	Load Indirect and Post-Inc.	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	None	2
LD	Rd, -Z	Load Indirect and Pre-Dec.	$Z \leftarrow Z - 1, Rd \leftarrow (Z)$	None	2
LDD	Rd, Z+q	Load Indirect with Displacement	$Rd \leftarrow (Z + q)$	None	2
LDS	Rd, k	Load Direct from SRAM	$Rd \leftarrow (k)$	None	2
ST	X, Rr	Store Indirect	$(X) \leftarrow Rr$	None	2
ST	X+, Rr	Store Indirect and Post-Inc.	$(X) \leftarrow Rr, X \leftarrow X + 1$	None	2
ST	-X, Rr	Store Indirect and Pre-Dec.	$X \leftarrow X - 1, (X) \leftarrow Rr$	None	2
ST	Y, Rr	Store Indirect	$(Y) \leftarrow Rr$	None	2
ST	Y+, Rr	Store Indirect and Post-Inc.	$(Y) \leftarrow Rr, Y \leftarrow Y + 1$	None	2
ST	-Y, Rr	Store Indirect and Pre-Dec.	$Y \leftarrow Y - 1, (Y) \leftarrow Rr$	None	2
STD	Y+q, Rr	Store Indirect with Displacement	$(Y + q) \leftarrow Rr$	None	2
ST	Z, Rr	Store Indirect	$(Z) \leftarrow Rr$	None	2
ST	Z+, Rr	Store Indirect and Post-Inc.	$(Z) \leftarrow Rr, Z \leftarrow Z + 1$	None	2
ST	-Z, Rr	Store Indirect and Pre-Dec.	$Z \leftarrow Z - 1, (Z) \leftarrow Rr$	None	2
STD	Z+q, Rr	Store Indirect with Displacement	$(Z + q) \leftarrow Rr$	None	2
STS	k, Rr	Store Direct to SRAM	$(k) \leftarrow Rr$	None	2
LPM		Load Program Memory	$R0 \leftarrow (Z)$	None	3
IN	Rd, P	In Port	$Rd \leftarrow P$	None	1
OUT	P, Rr	Out Port	$P \leftarrow Rr$	None	1
PUSH	Rr	Push Register on Stack	$STACK \leftarrow Rr$	None	2
POP	Rd	Pop Register from Stack	$Rd \leftarrow STACK$	None	2
BIT AND BIT-TEST INSTRUCTIONS					
SBI	P, b	Set Bit in I/O Register	$I/O(P, b) \leftarrow 1$	None	2
CBI	P, b	Clear Bit in I/O Register	$I/O(P, b) \leftarrow 0$	None	2
LSL	Rd	Logical Shift Left	$Rd(n+1) \leftarrow Rd(n), Rd(0) \leftarrow 0$	Z, C, N, V	1
LSR	Rd	Logical Shift Right	$Rd(n) \leftarrow Rd(n+1), Rd(7) \leftarrow 0$	Z, C, N, V	1
ROL	Rd	Rotate Left Through Carry	$Rd(0) \leftarrow C, Rd(n+1) \leftarrow Rd(n), C \leftarrow Rd(7)$	Z, C, N, V	1
ROR	Rd	Rotate Right Through Carry	$Rd(7) \leftarrow C, Rd(n) \leftarrow Rd(n+1), C \leftarrow Rd(0)$	Z, C, N, V	1
ASR	Rd	Arithmetic Shift Right	$Rd(n) \leftarrow Rd(n+1), n=0..6$	Z, C, N, V	1
SWAP	Rd	Swap Nibbles	$Rd(3..0) \leftarrow Rd(7..4), Rd(7..4) \leftarrow Rd(3..0)$	None	1
BSET	s	Flag Set	$SREG(s) \leftarrow 1$	SREG(s)	1
BCLR	s	Flag Clear	$SREG(s) \leftarrow 0$	SREG(s)	1
BST	Rr, b	Bit Store from Register to T	$T \leftarrow Rr(b)$	T	1
BLD	Rd, b	Bit load from T to Register	$Rd(b) \leftarrow T$	None	1
SEC		Set Carry	$C \leftarrow 1$	C	1
CLC		Clear Carry	$C \leftarrow 0$	C	1
SEN		Set Negative Flag	$N \leftarrow 1$	N	1
CLN		Clear Negative Flag	$N \leftarrow 0$	N	1
SEZ		Set Zero Flag	$Z \leftarrow 1$	Z	1
CLZ		Clear Zero Flag	$Z \leftarrow 0$	Z	1
SEI		Global Interrupt Enable	$I \leftarrow 1$	I	1
CLI		Global Interrupt Disable	$I \leftarrow 0$	I	1
SES		Set Signed Test Flag	$S \leftarrow 1$	S	1
CLS		Clear Signed Test Flag	$S \leftarrow 0$	S	1
SEV		Set Twos Complement Overflow.	$V \leftarrow 1$	V	1
CLV		Clear Twos Complement Overflow	$V \leftarrow 0$	V	1
SET		Set T in SREG	$T \leftarrow 1$	T	1
CLT		Clear T in SREG	$T \leftarrow 0$	T	1
SEH		Set Half Carry Flag in SREG	$H \leftarrow 1$	H	1
CLH		Clear Half Carry Flag in SREG	$H \leftarrow 0$	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

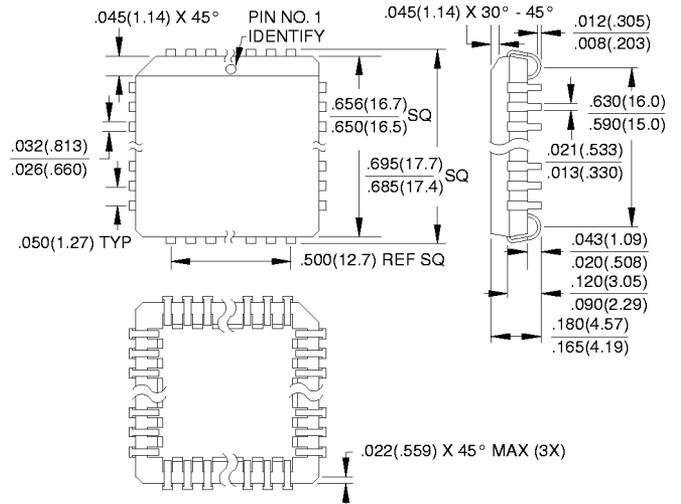
Packaging Information

44A, 44-lead, Thin (1.0 mm) Plastic Gull Wing Quad Flat Package (TQFP)
Dimensions in Millimeters and (Inches)*



*Controlling dimension: millimeters

44J, 44-lead, Plastic J-leaded Chip Carrier (PLCC)
Dimensions in Inches and (Millimeters)



40P6, 40-lead, 0.600" Wide,
Plastic Dual Inline Package (PDIP)
Dimensions in Inches and (Millimeters)
JEDEC STANDARD MS-011 AC

