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.
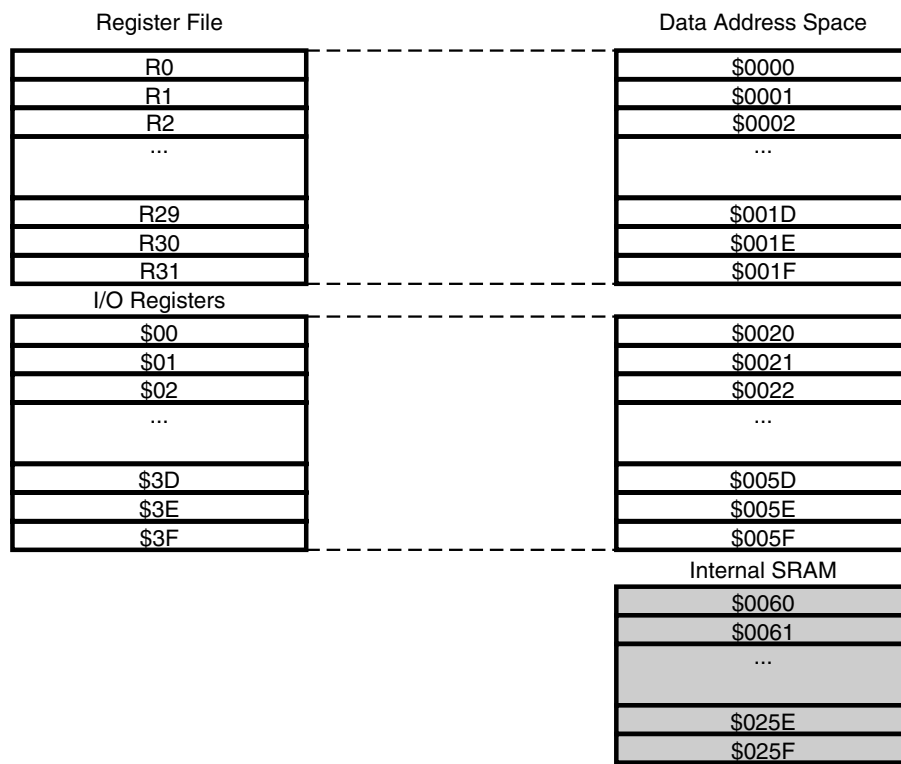
Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	AVR
Core Size	8-Bit
Speed	4MHz
Connectivity	SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	32
Program Memory Size	8KB (4K x 16)
Program Memory Type	FLASH
EEPROM Size	512 x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C
Mounting Type	Surface Mount
Package / Case	44-VFQFN Exposed Pad
Supplier Device Package	44-VQFN (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at90ls8535-4mi

SRAM Data Memory

Figure 8 shows how the AT90S8535 SRAM memory is organized.

Figure 8. SRAM Organization

Register File		Data Address Space
R0		$0000
R1		$0001
R2		$0002
...		...
R29		$001D
R30		$001E
R31		$001F

I/O Registers		
$00		$0020
$01		$0021
$02		$0022
...		...
$3D		$005D
$3E		$005E
$3F		$005F

Internal SRAM
$0060
$0061
...
$025E
$025F

The lower 608 data memory locations address the Register file, the I/O memory and the internal data SRAM. The first 96 locations address the Register file + I/O memory, and the next 512 locations address the internal data SRAM.

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 63 address locations reached from the base address given by the Y- or Z-registers.

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 512 bytes of internal data SRAM in the AT90S8535 are all accessible through all these addressing modes.

See the next section for a detailed description of the different addressing modes.
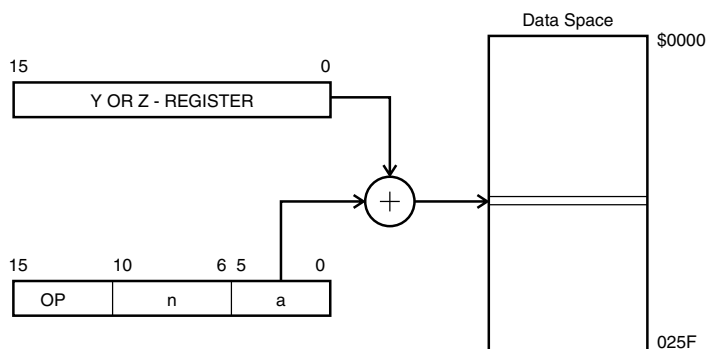
Data Direct

Figure 12. Direct Data Addressing
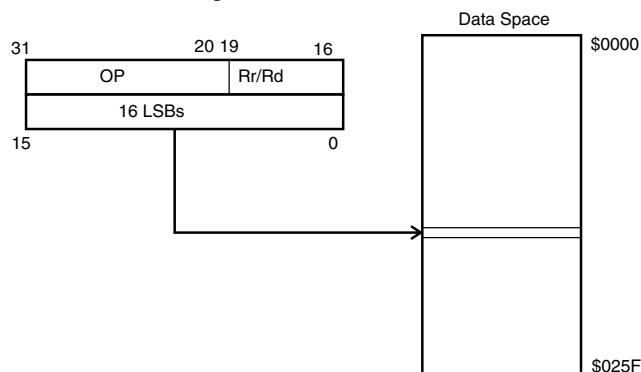
A 16-bit data address is contained in the 16 LSBs of a 2-word instruction. Rd/Rr specify the destination or source register.
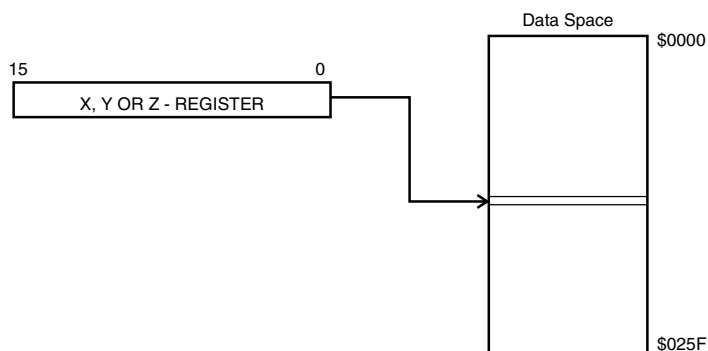
Data Indirect with Displacement

Figure 13. Data Indirect with Displacement

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

Data Indirect

Figure 14. Data Indirect Addressing

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

- **Bit 5 – ICF1: Input Capture Flag 1**

The ICF1 bit is set (one) to flag an input capture event, indicating that the Timer/Counter1 value has been transferred to the Input Capture Register (ICR1). ICF1 is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, ICF1 is cleared by writing a logical "1" to the flag. When the SREG I-bit and TICIE1 (Timer/Counter1 Input Capture Interrupt Enable) and ICF1 are set (one), the Timer/Counter1 Capture Interrupt is executed.

- **Bit 4 – OCF1A: Output Compare Flag 1A**

The OCF1A bit is set (one) when compare match occurs between the Timer/Counter1 and the data in OCR1A (Output Compare Register 1A). OCF1A is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, OCF1A is cleared by writing a logical "1" to the flag. When the I-bit in SREG and OCIE1A (Timer/Counter1 Compare Match InterruptA Enable) and the OCF1A are set (one), the Timer/Counter1 Compare A Match Interrupt is executed.

- **Bit 3 – OCF1B: Output Compare Flag 1B**

The OCF1B bit is set (one) when compare match occurs between the Timer/Counter1 and the data in OCR1B (Output Compare Register 1B). OCF1B is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, OCF1B is cleared by writing a logical "1" to the flag. When the I-bit in SREG and OCIE1B (Timer/Counter1 Compare Match InterruptB Enable) and the OCF1B are set (one), the Timer/Counter1 Compare Match B Interrupt is executed.

- **Bit 2 – TOV1: Timer/Counter1 Overflow Flag**

The TOV1 is set (one) when an overflow occurs in Timer/Counter1. TOV1 is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, TOV1 is cleared by writing a logical "1" to the flag. When the I-bit in SREG and TOIE1 (Timer/Counter1 Overflow Interrupt Enable) and TOV1 are set (one), the Timer/Counter1 Overflow Interrupt is executed. In up/down PWM mode, this bit is set when Timer/Counter1 advances from $0000.

- **Bit 1 – Res: Reserved Bit**

This bit is a reserved bit in the AT90S8535 and always reads zero.

- **Bit 0 – TOV0: Timer/Counter0 Overflow Flag**

The bit TOV0 is set (one) when an overflow occurs in Timer/Counter0. TOV0 is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, TOV0 is cleared by writing a logical "1" to the flag. When the SREG I-bit and TOIE0 (Timer/Counter0 Overflow Interrupt Enable) and TOV0 are set (one), the Timer/Counter0 Overflow Interrupt is executed. In up/down PWM mode, this bit is set when Timer/Counter1 advances from $0000.

External Interrupts

The external interrupts are triggered by the INT1 and INT0 pins. Observe that, if enabled, the interrupts will trigger even if the INT0/INT1 pins are configured as outputs. This feature provides a way of generating a software interrupt. 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 MCU Control Register (MCUCR). When the external interrupt is enabled and is configured as level-triggered, the interrupt will trigger as long as the pin is held low.

The external interrupts are set up as described in the specification for the MCU Control Register (MCUCR).

Table 8. Interrupt 1 Sense Control

ISC11	ISC10	Description
0	0	The low level of INT1 generates an interrupt request.
0	1	Reserved
1	0	The falling edge of INT1 generates an interrupt request.
1	1	The rising edge of INT1 generates an interrupt request.

The value on the INT pin is sampled before detecting edges. If edge interrupt is selected, pulses that last longer than one CPU 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. If enabled, a level-triggered interrupt will generate an interrupt request as long as the pin is held low.

- **Bit 1, 0 – ISC01, ISC00: Interrupt Sense Control 0 Bits 1 and 0**

The External Interrupt 0 is activated by the external pin INT0 if the SREG I-flag and the corresponding interrupt mask is set. The level and edges on the external INT0 pin that activate the interrupt are defined in Table 9.

Table 9. Interrupt 0 Sense Control

ISC01	ISC00	Description
0	0	The low level of INT0 generates an interrupt request.
0	1	Reserved
1	0	The falling edge of INT0 generates an interrupt request.
1	1	The rising edge of INT0 generates an interrupt request.

The value on the INT pin is sampled before detecting edges. If edge interrupt is selected, pulses that last longer than one CPU 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. If enabled, a level-triggered interrupt will generate an interrupt request as long as the pin is held low.

Sleep Modes

To enter any of the three sleep modes, the SE bit in MCUCR must be set (one) and a SLEEP instruction must be executed. The SM0 and SM1 bits in the MCUCR register select which sleep mode (Idle, Power-down or Power Save) will be activated by the SLEEP instruction. See Table 7.

If an enabled interrupt occurs while the MCU is in a sleep mode, the MCU wakes up, executes the interrupt routine and resumes execution from the instruction following SLEEP. The contents of the register file, SRAM and I/O memory are unaltered. If a reset occurs during Sleep Mode, the MCU wakes up and executes from the Reset vector.

Idle Mode

When the SM1/SM0 bits are set to 00, the SLEEP instruction makes the MCU enter the Idle Mode, stopping the CPU but allowing SPI, UARTs, Analog Comparator, ADC, Timer/Counters, Watchdog and the interrupt system to continue operating. This enables the MCU to wake up from external triggered interrupts as well as internal ones like the Timer Overflow and UART Receive 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. When the MCU wakes up from Idle Mode, the CPU starts program execution immediately.

Power-down Mode

When the SM1/SM0 bits are set to 10, the SLEEP instruction makes the MCU enter the Power-down mode. In this mode, the external oscillator is stopped while the external interrupts and the Watchdog (if enabled) continue operating. Only an external reset, a Watchdog reset (if enabled) or an external level interrupt can wake up the MCU.

Note that when a level-triggered interrupt is used for wake-up from power-down, the low level must be held for a time longer than the reset delay Time-out period t_{TOUT}.

When waking up from Power-down mode, 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 equal to the reset period, as shown in Table 3 on page 22.

If the wake-up condition disappears before the MCU wakes up and starts to execute, e.g., a low-level on is not held long enough, the interrupt causing the wake-up will not be executed.

Power Save Mode

When the SM1/SM0 bits are 11, the SLEEP instruction makes the MCU enter the Power Save Mode. This mode is identical to Power-down, with one exception: If Timer/Counter2 is clocked asynchronously, i.e., the AS2 bit in ASSR is set, Timer/Counter2 will run during sleep. In addition to the power-down wake-up sources, the device can also wake up from either a Timer Overflow or Output Compare event from Timer/Counter2 if the corresponding Timer/Counter2 interrupt enable bits are set in TIMSK and the global interrupt enable bit in SREG is set.

When waking up from Power Save Mode by an external interrupt, two instruction cycles are executed before the interrupt flags are updated. When waking up by the asynchronous timer, three instruction cycles are executed before the flags are updated. During these cycles, the processor executes instructions, but the interrupt condition is not readable and the interrupt routine has not started yet.

When waking up from Power Save Mode by an asynchronous timer interrupt, the part will wake up even if global interrupts are disabled. To ensure that the part executes the interrupt routine when waking up, also set the global interrupt enable bit in SREG.

If the asynchronous timer is not clocked asynchronously, Power-down mode is recommended instead of Power Save Mode because the contents of the registers in the asynchronous timer should be considered undefined after wake-up in Power Save Mode, even if AS2 is 0.

- **Bit 3 – CTC1: Clear Timer/Counter1 on Compare Match**

When the CTC1 control bit is set (one), the Timer/Counter1 is reset to $0000 in the clock cycle after a compareA match. If the CTC1 control bit is cleared, Timer/Counter1 continues counting and is unaffected by a compare match. Since the compare match is detected in the CPU clock cycle following the match, this function will behave differently when a prescaling higher than 1 is used for the timer. When a prescaling of 1 is used and the compareA register is set to C, the timer will count as follows if CTC1 is set:

... | C-2 | C-1 | C | 0 | 1 |...

When the prescaler is set to divide by 8, the timer will count like this:

... | C-2, C-2, C-2, C-2, C-2, C-2, C-2, C-2 | C-1, C-1, C-1, C-1, C-1, C-1, C-1, C-1 | C, 0, 0, 0, 0, 0, 0, 0 |1,1,1,1,1,1,1,1|...

In PWM mode, this bit has no effect.

- **Bits 2, 1, 0 – CS12, CS11, CS10: Clock Select1, Bits 2, 1 and 0**

The Clock Select1 bits 2, 1 and 0 define the prescaling source of Timer/Counter1.

Table 13. Clock 1 Prescale Select

CS12	CS11	CS10	Description
0	0	0	Stop, the Timer/Counter1 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 T1, falling edge
1	1	1	External Pin T1, rising edge

The Stop condition provides a Timer Enable/Disable function. The CK down divided modes are scaled directly from the CK oscillator clock. If the external pin modes are used, the corresponding setup must be performed in the actual Direction Control Register (cleared to zero gives an input pin).

Timer/Counter1 – TCNT1H AND TCNT1L

Bit	15	14	13	12	11	10	9	8	
$2D ($4D)	MSB								TCNT1H
$2C ($4C)								LSB	TCNT1L
	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	

This 16-bit register contains the prescaled value of the 16-bit Timer/Counter1. To ensure that both the high and low bytes are read and written simultaneously when the CPU accesses these registers, the access is performed using an 8-bit temporary register (TEMP). This temporary register is also used when accessing OCR1A, OCR1B and ICR1. If the main program and also interrupt routines perform access to registers using

TEMP, interrupts must be disabled during access from the main program (and from interrupt routines if interrupts are allowed from within interrupt routines).

- TCNT1 Timer/Counter1 Write:
  When the CPU writes to the high byte TCNT1H, the written data is placed in the TEMP register. Next, when the CPU writes the low byte TCNT1L, this byte of data is combined with the byte data in the TEMP register, and all 16 bits are written to the TCNT1 Timer/Counter1 register simultaneously. Consequently, the high byte TCNT1H must be accessed first for a full 16-bit register write operation.

- TCNT1 Timer/Counter1 Read:
  When the CPU reads the low byte TCNT1L, the data of the low byte TCNT1L is sent to the CPU and the data of the high byte TCNT1H is placed in the TEMP register. When the CPU reads the data in the high byte TCNT1H, the CPU receives the data in the TEMP register. Consequently, the low byte TCNT1L must be accessed first for a full 16-bit register read operation.

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 compare match only occurs if Timer/Counter1 counts to the OCR value. A software write that sets TCNT1 and OCR1A or OCR1B to the same value does not generate a compare match.

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,

39

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.

Timer/Counter1 Input Capture Register – ICR1H AND ICR1L

Bit	15	14	13	12	11	10	9	8	
$27 ($47)	MSB								ICR1H
$26 ($46)								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	

The Input Capture Register is a 16-bit read-only register.

When the rising or falling edge (according to the input capture edge setting [ICES1]) of the signal at the input capture pin (ICP) is detected, the current value of the Timer/Counter1 is transferred to the Input Capture Register (ICR1). At the same time, the input capture flag (ICF1) is set (one).

Since the Input Capture Register (ICR1) is a 16-bit register, a temporary register (TEMP) is used when ICR1 is read to ensure that both bytes are read simultaneously. When the CPU reads the low byte ICR1L, the data is sent to the CPU and the data of the high byte ICR1H is placed in the TEMP register. When the CPU reads the data in the high byte ICR1H, the CPU receives the data in the TEMP register. Consequently, the low-byte ICR1L must be accessed first for a full 16-bit register read operation.

The TEMP register is also used when accessing TCNT1, OCR1A and OCR1B. If the main program and interrupt routines perform access to registers using TEMP, interrupts must be disabled during access from the main program.

Timer/Counter1 In PWM Mode

When the PWM mode is selected, Timer/Counter1, the Output Compare Register1A (OCR1A) and the Output Compare Register1B (OCR1B) form a dual 8-, 9- or 10-bit, free-running, glitch-free and phase-correct PWM with outputs on the PD5(OC1A) and PD4(OC1B) pins. Timer/Counter1 acts as an up/down counter, counting up from $0000 to TOP (see Table 14), where it turns and counts down again to zero before the cycle is repeated. When the counter value matches the contents of the 10 least significant bits of OCR1A or OCR1B, the PD5(OC1A)/PD4(OC1B) pins are set or cleared according to the settings of the COM1A1/COM1A0 or COM1B1/COM1B0 bits in the Timer/Counter1 Control Register (TCCR1A). Refer to Table 15 for details.

Table 14. Timer TOP Values and PWM Frequency

PWM Resolution	Timer TOP value	Frequency
8-bit	$00FF (255)	$f_{TCK1}/510$
9-bit	$01FF (511)	$f_{TCK1}/1022$
10-bit	$03FF(1023)	$f_{TCK1}/2046$

Note that if the Compare Register contains the TOP value and the prescaler is not in use (CS12..CS10 = 001), the PWM output will not produce any pulse at all, because the

Timer/Counter Control Register (TCCR2). The interrupt enable/disable settings are found in the Timer/Counter Interrupt Mask Register (TIMSK).

This module features a high-resolution and a high-accuracy usage with the lower prescaling opportunities. Similarly, the high prescaling opportunities make this unit useful for lower speed functions or exact timing functions with infrequent actions.

The Timer/Counter supports an Output Compare function using the Output Compare Register (OCR2) as the data source to be compared to the Timer/Counter contents.The Output Compare function includes optional clearing of the counter on compare match and action on the Output Compare Pin, PD7(OC2), on compare match. Writing to PORTD7 does not set the OC2 value to a predetermined value.

Timer/Counter2 can also be used as an 8-bit Pulse Width Modulator. In this mode, Timer/Counter2 and the Output Compare Register serve as a glitch-free, stand-alone PWM with centered pulses. Refer to page 45 for a detailed description of this function.

Timer/Counter2 Control Register – TCCR2

Bit	7	6	5	4	3	2	1	0	
$25 ($45)	–	PWM2	COM21	COM20	CTC2	CS22	CS21	CS20	TCCR2
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – Res: Reserved Bit**

This bit is a reserved bit in the AT90S8535 and always reads as zero.

- **Bit 6 – PWM2: Pulse Width Modulator Enable**

When set (one), this bit enables PWM mode for Timer/Counter2. This mode is described on page 45.

- **Bits 5, 4 – COM21, COM20: Compare Output Mode, Bits 1 and 0**

The COM21 and COM20 control bits determine any output pin action following a compare match in Timer/Counter2. Output pin actions affect pin PD7(OC2). This is an alternative function to an I/O port and the corresponding direction control bit must be set (one) to control an output pin. The control configuration is shown in Table 17.

Table 17. Compare Mode Select

COM21	COM20	Description
0	0	Timer/Counter disconnected from output pin OC2
0	1	Toggle the OC2 output line.
1	0	Clear the OC2 output line (to zero).
1	1	Set the OC2 output line (to one).

Note: In PWM mode, these bits have a different function. Refer to Table 19 for a detailed description.

- **Bit 3 – CTC2: Clear Timer/Counter on Compare Match**

When the CTC2 control bit is set (one), Timer/Counter2 is reset to $00 in the CPU clock cycle after a compare match. If the control bit is cleared, Timer/Counter2 continues counting and is unaffected by a compare match. Since the compare match is detected in the CPU clock cycle following the match, this function will behave differently when a prescaling higher than 1 is used for the timer. When a prescaling of 1 is used and the compare2 register is set to C, the timer will count as follows if CTC2 is set:

... | C-2 | C-1 | C | 0 | 1 |...

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 "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 43.

Figure 43. Sampling Received Data
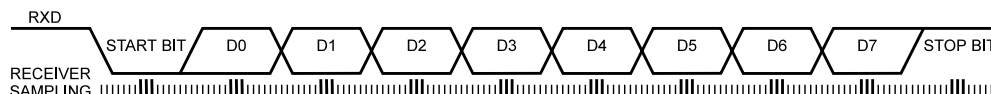
When the stop bit enters the receiver, the majority of the three samples must be one to accept the stop bit. If two or more samples are logical "0"s, the Framing Error (FE) flag in the UART Status Register (USR) is set. Before reading the UDR register, the user should always check the FE bit to detect 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 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 bits long, plus start and stop bits. The ninth 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 ninth data bit received is the RXB8 bit in the UCR register.

Analog-to-Digital Converter

Feature list

- **10-bit Resolution**
- **0.5 LSB Integral Non-linearity**
- **±2 LSB Absolute Accuracy**
- **65 - 260 µs Conversion Time**
- **Up to 15 kSPS at Maximum Resolution**
- **8 Multiplexed Input Channels**
- **Rail-to-Rail Input Range**
- **Free Running or Single Conversion Mode**
- **Interrupt on ADC Conversion Complete**
- **Sleep Mode Noise Canceler**

The AT90S8535 features a 10-bit successive approximation ADC. The ADC is connected to an 8-channel Analog Multiplexer that allows each pin of Port A to be used as an input for the ADC. The ADC contains a Sample and Hold Amplifier that ensures that the input voltage to the ADC is held at a constant level during conversion. A block diagram of the ADC is shown in Figure 45.

The ADC has two separate analog supply voltage pins, AVCC and AGND. AGND must be connected to GND and the voltage on AV_{CC} must not differ more than ±0.3V from V_{CC}. See "ADC Noise Canceling Techniques" on page 74 on how to connect these pins.

An external reference voltage must be applied to the AREF pin. This voltage must be in the range 2V - AV_{CC}.

Figure 45. Analog-to-Digital Converter Block Schematic
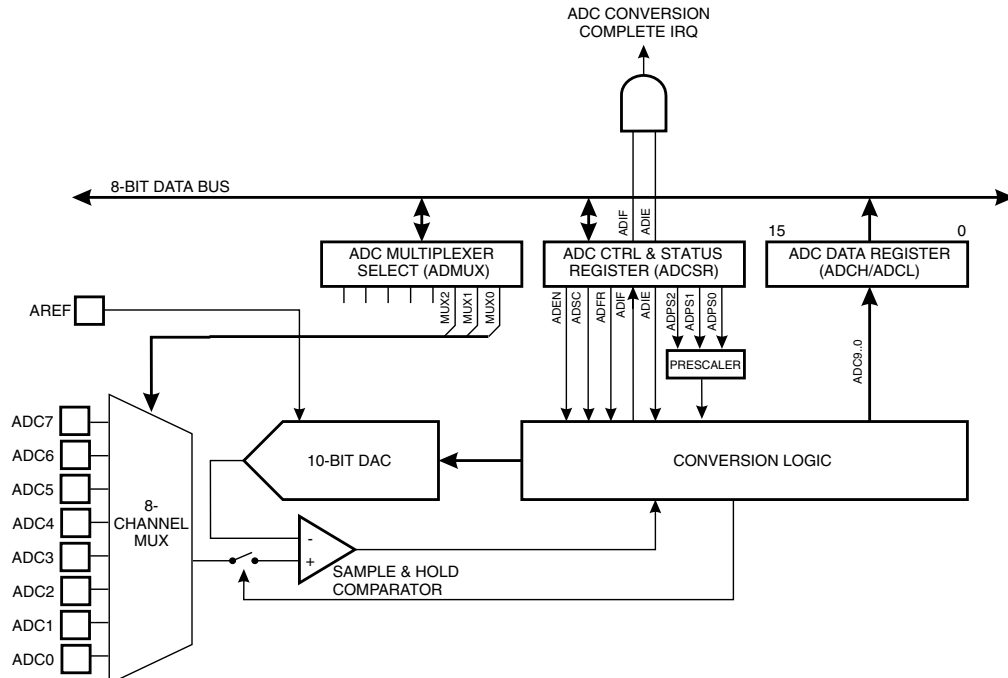
- **ADC9..0: ADC Conversion result**

These bits represent the result from the conversion. $000 represents analog ground and $3FF represents the selected reference voltage minus one LSB.

Scanning Multiple Channels

Since change of analog channel always is delayed until a conversion is finished, the Free Running Mode can be used to scan multiple channels without interrupting the converter. Typically, the ADC Conversion Complete interrupt will be used to perform the channel shift. However, the user should take the following fact into consideration: The interrupt triggers once the result is ready to be read. In Free Running Mode, the next conversion will start immediately when the interrupt triggers. If ADMUX is changed after the interrupt triggers, the next conversion has already started and the old setting is used.

ADC Noise Canceling Techniques

Digital circuitry inside and outside the AT90S8535 generates EMI that might affect the accuracy of analog measurements. If conversion accuracy is critical, the noise level can be reduced by applying the following techniques:

1. The analog part of the AT90S8535 and all analog components in the application should have a separate analog ground plane on the PCB. This ground plane is connected to the digital ground plane via a single point on the PCB.

2. Keep analog signal paths as short as possible. Make sure analog tracks run over the analog ground plane and keep them well away from high-speed switching digital tracks.

3. The AV_{CC} pin on the AT90S8535 should be connected to the digital V_{CC} supply voltage via an LC network as shown in Figure 50.

4. Use the ADC noise canceler function to reduce induced noise from the CPU.

5. If some Port A pins are used as digital outputs, it is essential that these do not switch while a conversion is in progress.

Figure 50. ADC Power Connections
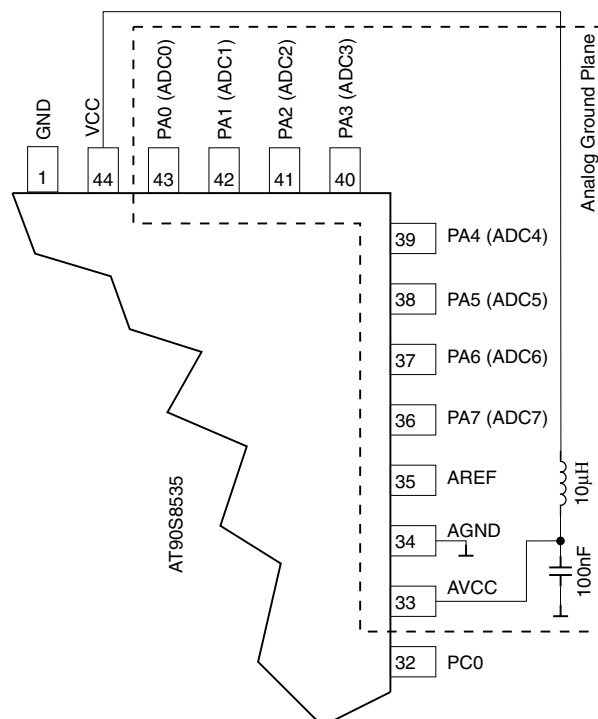
I/O Ports

All AVR ports have true read-modify-write functionality when used as general digital I/O ports. This means that the direction of one port pin can be changed without unintentionally changing the direction of any other pin with the SBI and CBI instructions. The same applies for changing drive value (if configured as output) or enabling/disabling of pull-up resistors (if configured as input).

Port A

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

Three I/O memory address locations are allocated for Port A, one each for the Data Register – PORTA, $1B($3B), Data Direction Register – DDRA, $1A($3A) and the Port A Input Pins – PINA, $19($39). The Port A 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 A output buffers can sink 20 mA and thus 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.

Port A has an alternate function as analog inputs for the ADC. If some Port A pins are configured as outputs, it is essential that these do not switch when a conversion is in progress. This might corrupt the result of the conversion.

During Power-down mode, the Schmitt trigger of the digital input is disconnected. This allows analog signals that are close to $V_{CC}/2$ to be present during power-down without causing excessive power consumption.

Port A Data Register – PORTA

Bit	7	6	5	4	3	2	1	0	
$1B ($3B)	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	PORTA
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 A Data Direction Register – DDRA

Bit	7	6	5	4	3	2	1	0	
$1A ($3A)	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	DDRA
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 A Input Pins Address – PINA

Bit	7	6	5	4	3	2	1	0	
$19 ($39)	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0	PINA
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 A Input Pins address (PINA) is not a register; this address enables access to the physical value on each Port A pin. When reading PORTA, the Port A Data Latch is read and when reading PINA, the logical values present on the pins are read.

Port B As General Digital I/O

All eight pins in Port B have equal functionality when used as digital I/O pins.

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

Table 31. DDBn Effects on Port B Pins

DDBn	PORTBn	I/O	Pull-up	Comment
0	0	Input	No	Tri-state (high-Z)
0	1	Input	Yes	PBn will source current if ext. pulled low
1	0	Output	No	Push-pull Zero Output
1	1	Output	No	Push-pull One Output

Note: n: 7,6…0, pin number.

Alternate Functions of Port B

The alternate pin configuration is as follows:

• **SCK – Port B, Bit 7**

SCK: Master clock output, slave clock input pin for SPI channel. When the SPI is enabled as a slave, this pin is configured as an input regardless of the setting of DDB7. When the SPI is enabled as a master, the data direction of this pin is controlled by DDB7. When the pin is forced to be an input, the pull-up can still be controlled by the PORTB7 bit. See the description of the SPI port for further details.

• **MISO – Port B, Bit 6**

MISO: Master data input, slave data output pin for SPI channel. When the SPI is enabled as a master, this pin is configured as an input regardless of the setting of DDB6. When the SPI is enabled as a slave, the data direction of this pin is controlled by DDB6. When the pin is forced to be an input, the pull-up can still be controlled by the PORTB6 bit. See the description of the SPI port for further details.

• **MOSI – Port B, Bit 5**

MOSI: SPI Master data output, slave data input for SPI channel. When the SPI is enabled as a slave, this pin is configured as an input regardless of the setting of DDB5. When the SPI is enabled as a master, the data direction of this pin is controlled by DDB5. When the pin is forced to be an input, the pull-up can still be controlled by the PORTB5 bit. See the description of the SPI port for further details.

• **$\overline{SS}$ – Port B, Bit 4**

$\overline{SS}$: Slave port select input. When the SPI is enabled as a slave, this pin is configured as an input regardless of the setting of DDB4. As a slave, the SPI is activated when this pin is driven low. When the SPI is enabled as a master, the data direction of this pin is controlled by DDB4. When the pin is forced to be an input, the pull-up can still be controlled by the PORTB4 bit. See the description of the SPI port for further details.

• **AIN1 – Port B, Bit 3**

AIN1, Analog Comparator Negative Input. When configured as an input (DDB3 is cleared [zero]) and with the internal MOS pull-up resistor switched off (PB3 is cleared [zero]), this pin also serves as the negative input of the on-chip Analog Comparator. During Power-down mode, the Schmitt trigger of the digital input is disconnected. This
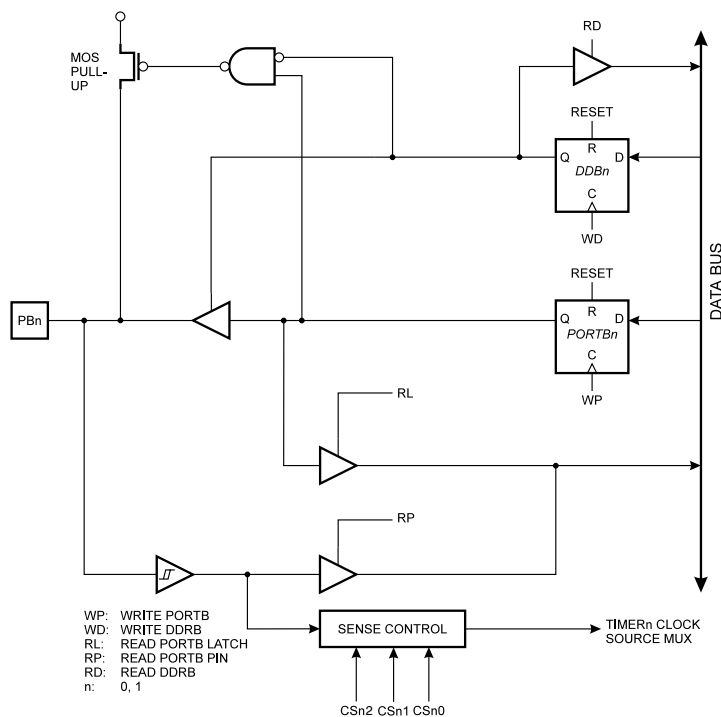
allows analog signals that are close to $V_{CC}/2$ to be present during power-down without causing excessive power consumption.

• **AIN0 – Port B, Bit 2**

AIN0, Analog Comparator Positive Input. When configured as an input (DDB2 is cleared [zero]) and with the internal MOS pull-up resistor switched off (PB2 is cleared [zero]), this pin also serves as the positive input of the on-chip Analog Comparator. During Power-down mode, the Schmitt trigger of the digital input is disconnected. This allows analog signals that are close to $V_{CC}/2$ to be present during power-down without causing excessive power consumption.

• **T1 – Port B, Bit 1**

T1, Timer/Counter1 counter source. See the timer description for further details.

• **T0 – Port B, Bit 0**

T0: Timer/Counter0 counter source. See the timer description for further details.

Port B Schematics   Note that all port pins are synchronized. The synchronization latches are, however, not shown in the figures.

Figure 52. Port B Schematic Diagram (Pins PB0 and PB1)

1041H–11/01

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 20 mA 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.

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	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	

The Port C Input Pins address (PINC) is not a register; 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.

Port C As General Digital I/O

All eight 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 is 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 pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Table 32. DDCn Effects on Port C Pins

DDCn	PORTCn	I/O	Pull-up	Comment
0	0	Input	No	Tri-state (high-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

Note: n: 7…0, pin number

Memory Programming

Program and Data Memory Lock Bits

The AT90S8535 MCU provides two Lock bits that can be left unprogrammed ("1") or can be programmed ("0") to obtain the additional features listed in Table 35. The Lock bits can only be erased with the Chip Erase command.

Table 35. Lock Bit Protection Modes

Memory Lock Bits			
Mode	LB1	LB2	Protection Type
1	1	1	No memory lock features enabled.
2	0	1	Further programming of the Flash and EEPROM is disabled.[1]
3	0	0	Same as mode 2 and verify is also disabled.

Note: 1. In Parallel Mode, further programming of the Fuse bits is also disabled. Program the Fuse bits before programming the Lock bits.

Fuse Bits

The AT90S8535 has two Fuse bits, SPIEN and FSTRT.

- When the SPIEN Fuse is programmed ("0"), Serial Program and Data Downloading is enabled. Default value is programmed ("0"). The SPIEN Fuse is not accessible in Serial Programming Mode.
- When the FSTRT Fuse is programmed ("0"), the short start-up time is selected. Default value is unprogrammed ("1").

The status of the Fuse bits is not affected by Chip Erase.

Signature Bytes

All Atmel microcontrollers have a three-byte signature code that identifies the device. This code can be read in both Serial and Parallel modes. The three bytes reside in a separate address space.

1. $000: $1E (indicates manufactured by Atmel)
2. $001: $93 (indicates 8K bytes Flash memory)
3. $002: $03 (indicates AT90S8535 device when signature byte $001 is $93)

Note: 1. When both Lock bits are programmed (lock mode 3), the signature bytes cannot be read in Serial Mode. Reading the signature bytes will return: $00, $01 and $02.

Programming the Flash and EEPROM

Atmel's AT90S8535 offers 8K bytes of in-system reprogrammable Flash program memory and 512 bytes of EEPROM data memory.

The AT90S8535 is shipped with the On-chip Flash program and EEPROM data memory arrays in the erased state (i.e., contents = $FF) and ready to be programmed. This device supports a high-voltage (12V) Parallel Programming Mode and a low-voltage Serial Programming Mode. The +12V is used for programming enable only and no current of significance is drawn by this pin. The Serial Programming Mode provides a convenient way to download program and data into the AT90S8535 inside the user's system.

The program and data memory arrays on the AT90S8535 are programmed byte-by-byte in either programming mode. For the EEPROM, an auto-erase cycle is provided within the self-timed write instruction in the Serial Programming Mode.

During programming, the supply voltage must be in accordance with Table 36.

Table 37. Pin Name Mapping

Signal Name in Programming Mode	Pin Name	I/O	Function
RDY/$\overline{\text{BSY}}$	PD1	O	0: Device is busy programming, 1: Device is ready for new command
$\overline{\text{OE}}$	PD2	I	Output Enable (Active low)
$\overline{\text{WR}}$	PD3	I	Write Pulse (Active low)
BS	PD4	I	Byte Select ("0" selects low byte, "1" selects high byte)
XA0	PD5	I	XTAL Action Bit 0
XA1	PD6	I	XTAL Action Bit 1
DATA	PB7 - 0	I/O	Bi-directional Data Bus (Output when $\overline{\text{OE}}$ is low)

Table 38. XA1 and XA0 Coding

XA1	XA0	Action when XTAL1 is Pulsed
0	0	Load Flash or EEPROM Address (high or low address byte determined by BS)
0	1	Load Data (High or low data byte for Flash determined by BS)
1	0	Load Command
1	1	No Action, Idle

Table 39. Command Byte Bit Coding

Command Byte	Command Executed
1000 0000	Chip Erase
0100 0000	Write Fuse Bits
0010 0000	Write Lock Bits
0001 0000	Write Flash
0001 0001	Write EEPROM
0000 1000	Read Signature Bytes
0000 0100	Read Lock and Fuse Bits
0000 0010	Read Flash
0000 0011	Read EEPROM

Enter Programming Mode

The following algorithm puts the device in Parallel Programming Mode:

1. Apply supply voltage according to Table 36, between V_{CC} and GND.
2. Set the $\overline{\text{RESET}}$ and BS pin to "0" and wait at least 100 ns.
3. Apply 11.5 - 12.5V to $\overline{\text{RESET}}$. Any activity on BS within 100 ns after +12V has been applied to $\overline{\text{RESET}}$, will cause the device to fail entering programming mode.

Figure 90. I/O Pin Source Current vs. Output Voltage

I/O PIN SOURCE CURRENT vs. OUTPUT VOLTAGE
$V_{cc} = 5V$
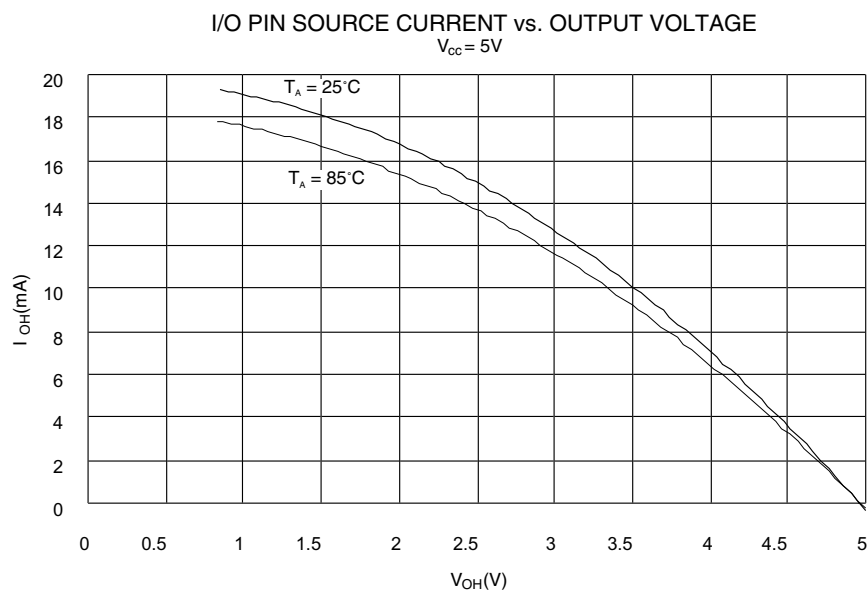
Figure 91. I/O Pin Sink Current vs. Output Voltage

I/O PIN SOURCE CURRENT vs. OUTPUT VOLTAGE
$V_{cc} = 2.7V$

115