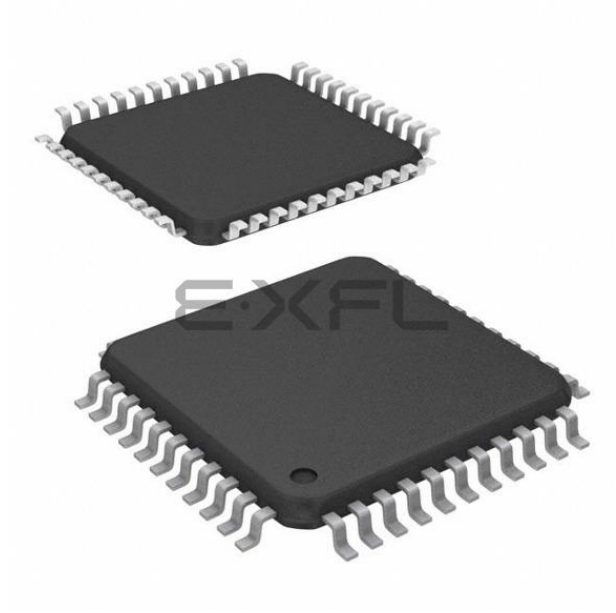
Microchip Technology - AT90S8515A-4AC Datasheet



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	-
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C
Mounting Type	Surface Mount
Package / Case	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at90s8515a-4ac

one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The AT90S8515 provides the following features: 8K bytes of In-System Programmable Flash, 512 bytes EEPROM, 512 bytes SRAM, 32 general-purpose I/O lines, 32 general-purpose working registers, flexible timer/counters with compare modes, internal and external interrupts, a programmable serial UART, programmable Watchdog Timer with internal oscillator, an SPI serial port and two software-selectable power-saving modes. The Idle Mode stops the CPU while allowing the SRAM, timer/counters, SPI port and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the oscillator, disabling all other chip functions until the next external interrupt or hardware reset.

The device is manufactured using Atmel's high-density nonvolatile memory technology. The On-chip In-System Programmable Flash allows the program memory to be reprogrammed In-System through an SPI serial interface or by a conventional nonvolatile memory programmer. By combining an enhanced RISC 8-bit CPU with In-System Programmable Flash on a monolithic chip, the Atmel AT90S8515 is a powerful microcontroller that provides a highly flexible and cost-effective solution to many embedded control applications.

The AT90S8515 AVR is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, in-circuit emulators and evaluation kits.

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 active.

Port A serves as multiplexed address/data input/output when using external SRAM.

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 active.

Port B also serves the functions of various special features of the AT90S8515 as listed on page 66.

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 active.

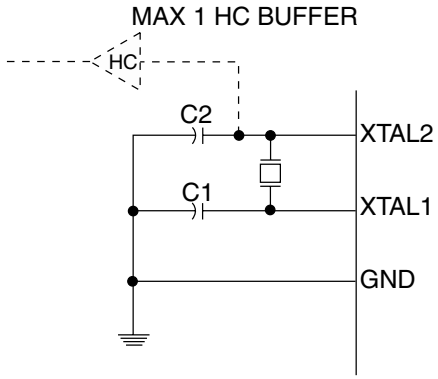
Port C also serves as address output when using external SRAM.

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

Crystal Oscillator

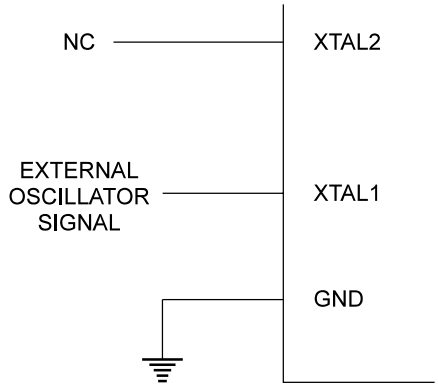
XTAL1 and XTAL2 are input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure 2. Either a quartz crystal or a ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven as shown in Figure 3.

Figure 2. Oscillator Connections



Note: When using the MCU oscillator as a clock for an external device, an HC buffer should be connected as indicated in the figure.

Figure 3. External Clock Drive Configuration



two additional clock cycles is used per byte. This has the following effect: Data transfer instructions take two extra clock cycles, whereas interrupt, subroutine calls and returns will need four clock cycles more than specified in the instruction set manual.

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-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, the 512 bytes of internal data SRAM, and the 64K bytes of optional external data SRAM in the AT90S8515 are all accessible through all these addressing modes.

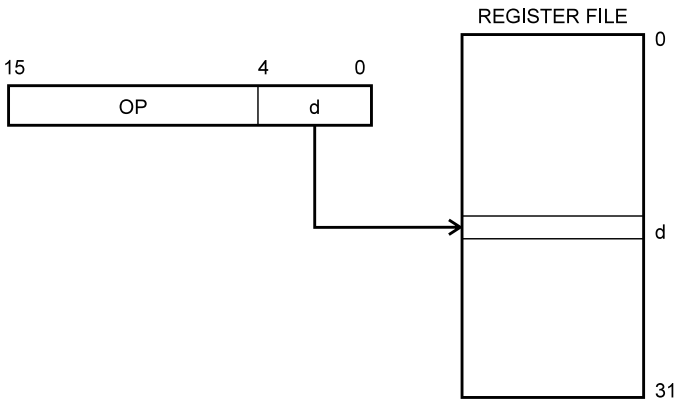
See the next section for a detailed description of the different addressing modes.

Program and Data Addressing Modes

The AT90S8515 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).

Figure 20. The Parallel Instruction Fetches and Instruction Executions

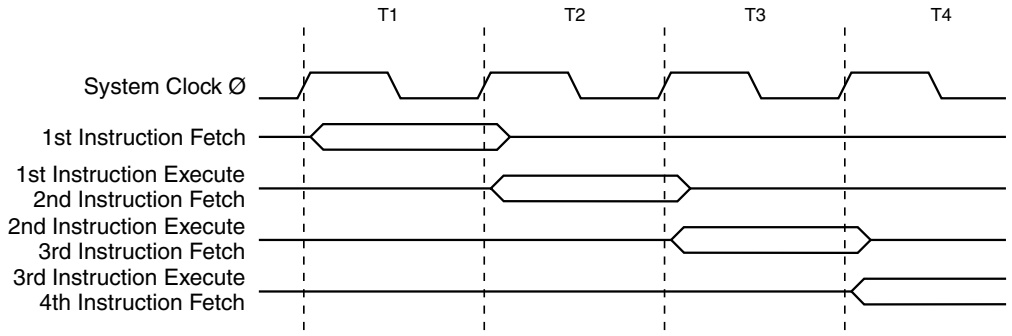
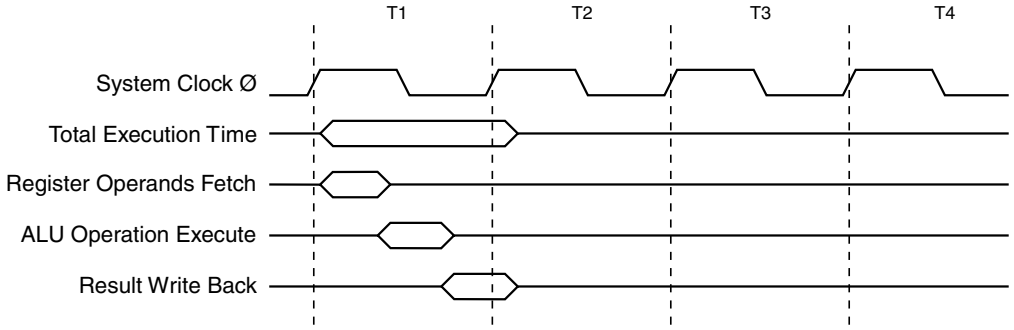


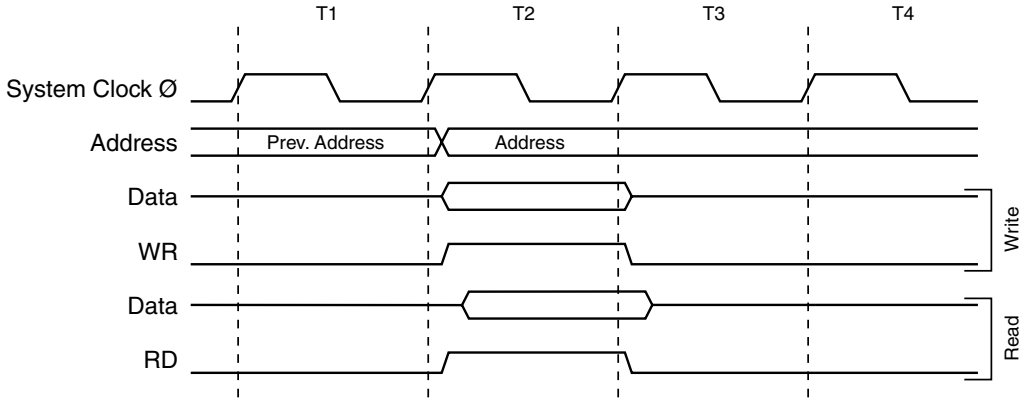
Figure 21 shows the internal timing concept for the register file. In a single clock cycle an ALU operation using two register operands is executed and the result is stored back to the destination register.

Figure 21. Single Cycle ALU Operation



The internal data SRAM access is performed in two System Clock cycles as described in Figure 22.

Figure 22. On-chip Data SRAM Access Cycles



See “Interface to External SRAM” on page 60 for a description of the access to the external SRAM.

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.

- **Bit 5 – H: Half-carry Flag**

The half-carry flag H indicates a half-carry in some arithmetic operations. See the Instruction Set description for detailed information.

- **Bit 4 – S: Sign Bit, S = N ⊕ V**

The S-bit is always an exclusive or between the negative flag N and the two's complement overflow flag V. See the Instruction Set description for detailed information.

- **Bit 3 – V: Two's Complement Overflow Flag**

The two's complement overflow flag V supports two's complement arithmetics. See the Instruction Set description for detailed information.

- **Bit 2 – N: Negative Flag**

The negative flag N indicates a negative result after the different arithmetic and logic operations. See the Instruction Set description for detailed information.

- **Bit 1 – Z: Zero Flag**

The zero flag Z indicates a zero result after the different arithmetic and logic operations. See the Instruction Set description for detailed information.

- **Bit 0 – C: Carry Flag**

The carry flag C indicates a carry in an arithmetic or logic operation. See the Instruction Set description for detailed information.

Note that the Status Register is not automatically stored when entering an interrupt routine and restored when returning from an interrupt routine. This must be handled by software.

Stack Pointer – SP

The general AVR 16-bit Stack Pointer is effectively built up of two 8-bit registers in the I/O space locations $3E ($5E) and $3D ($5D). As the AT90S8515 supports up to 64 Kb external SRAM, all 16 bits are used.

Bit	15	14	13	12	11	10	9	8	
$3E ($5E)	**SP15**	**SP14**	**SP13**	**SP12**	**SP11**	**SP10**	**SP9**	**SP8**	**SPH**
$3D ($5D)	**SP7**	**SP6**	**SP5**	**SP4**	**SP3**	**SP2**	**SP1**	**SP0**	**SPL**
	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 Stack Pointer points to the data SRAM stack area where the Subroutine and Interrupt stacks are located. This stack space in the data SRAM must be defined by the program before any subroutine calls are executed or interrupts are enabled. The Stack Pointer must be set to point above $60. The Stack Pointer is decremented by 1 when data is pushed onto the stack with the PUSH instruction and it is decremented by 2 when an address is pushed onto the stack with subroutine calls and interrupts. The Stack Pointer is incremented by 1 when data is popped from the stack with the POP instruction and it is incremented by 2 when an address is popped from the stack with return from subroutine RET or return from interrupt RETI.

- **Bit 6 – INTF0: External Interrupt Flag0**

When an edge on the INT0 pin triggers an interrupt request, the corresponding interrupt flag, INTF0, becomes set (one). If the I-bit in SREG and the corresponding interrupt enable bit, INT0 in GIMSK are set (one), the MCU will jump to the interrupt vector. The flag is cleared when the interrupt routine is executed. Alternatively, the flag is cleared by writing a logical "1" to it. This flag is always cleared when INT0 is configured as level interrupt.

- **Bits 5..0 – Res: Reserved Bits**

These bits are reserved bits in the AT90S8515 and always read as zero.

Timer/Counter Interrupt Mask Register – TIMSK

Bit	7	6	5	4	3	2	1	0	
$39 ($59)	TOIE1	OCIE1A	OCIE1B	–	TICIE1	–	TOIE0	–	TIMSK
Read/Write	R/W	R/W	R/W	R	R/W	R	R/W	R	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – TOIE1: Timer/Counter1 Overflow Interrupt Enable**

When the TOIE1 bit is set (one) and the I-bit in the Status Register is set (one), the Timer/Counter1 Overflow interrupt is enabled. The corresponding interrupt (at vector $006) is executed if an overflow in Timer/Counter1 occurs, i.e., when the TOV1 bit is set in the Timer/Counter Interrupt Flag Register (TIFR).

- **Bit 6 – OCE1A: Timer/Counter1 Output CompareA Match Interrupt Enable**

When the OCIE1A bit is set (one) and the I-bit in the Status Register is set (one), the Timer/Counter1 CompareA Match interrupt is enabled. The corresponding interrupt (at vector $004) is executed if a CompareA match in Timer/Counter1 occurs, i.e., when the OCF1A bit is set in the Timer/Counter Interrupt Flag Register (TIFR).

- **Bit 5 – OCIE1B: Timer/Counter1 Output CompareB Match Interrupt Enable**

When the OCIE1B bit is set (one) and the I-bit in the Status Register is set (one), the Timer/Counter1 CompareB Match interrupt is enabled. The corresponding interrupt (at vector $005) is executed if a CompareB match in Timer/Counter1 occurs, i.e., when the OCF1B bit is set in the Timer/Counter Interrupt Flag Register (TIFR).

- **Bit 4 – Res: Reserved Bit**

This bit is a reserved bit in the AT90S8515 and always reads zero.

- **Bit 3 – TICIE1: Timer/Counter1 Input Capture Interrupt Enable**

When the TICIE1 bit is set (one) and the I-bit in the Status Register is set (one), the Timer/Counter1 Input Capture Event interrupt is enabled. The corresponding interrupt (at vector $003) is executed if a capture-triggering event occurs on pin 31, ICP, i.e., when the ICF1 bit is set in the Timer/Counter Interrupt Flag Register (TIFR).

- **Bit 2 – Res: Reserved Bit**

This bit is a reserved bit in the AT90S8515 and always reads zero.

- **Bit 1 – TOIE0: Timer/Counter0 Overflow Interrupt Enable**

When the TOIE0 bit is set (one) and the I-bit in the Status Register is set (one), the Timer/Counter0 Overflow interrupt is enabled. The corresponding interrupt (at vector $007) is executed if an overflow in Timer/Counter0 occurs, i.e., when the TOV0 bit is set in the Timer/Counter Interrupt Flag Register (TIFR).

- **Bit 0 – Res: Reserved Bit**

This bit is a reserved bit in the AT90S8515 and always reads zero.

(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 clock period. The external clock signal is sampled on the rising edge of the internal CPU clock.

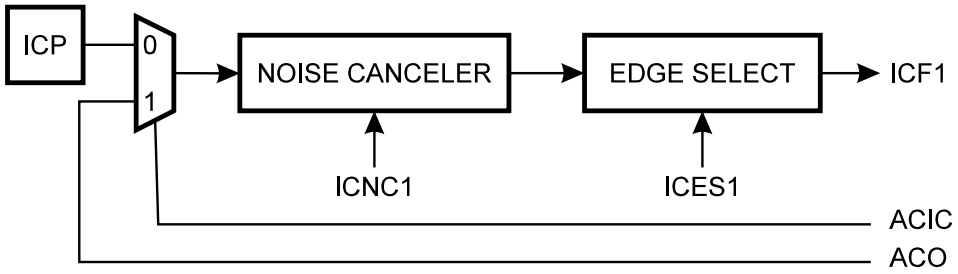
The 16-bit Timer/Counter1 features both a high-resolution and a high-accuracy usage with the lower prescaling opportunities. Similarly, the high prescaling opportunities make the Timer/Counter1 useful for lower speed functions or exact timing functions with infrequent actions.

The Timer/Counter1 supports two Output Compare functions using the Output Compare Register 1 A and B (OCR1A and OCR1B) as the data sources to be compared to the Timer/Counter1 contents. The Output Compare functions include optional clearing of the counter on compareA match and actions on the Output Compare pins on both compare matches.

Timer/Counter1 can also be used as an 8-, 9- or 10-bit Pulse Width Modulator. In this mode, the counter and the OCR1A/OCR1B registers serve as a dual, glitch-free, standalone PWM with centered pulses. Refer to page 47 for a detailed description of this function.

The Input Capture function of Timer/Counter1 provides a capture of the Timer/Counter1 contents to the Input Capture Register (ICR1), triggered by an external event on the input capture pin (ICP). The actual capture event settings are defined by the Timer/Counter1 Control Register (TCCR1B). In addition, the Analog Comparator can be set to trigger the Input Capture. Refer to “Analog Comparator” on page 59 for details on this. The ICP pin logic is shown in Figure 31.

Figure 31. ICP Pin Schematic Diagram



ACIC: COMPARATOR IC ENABLE
ACO: COMPARATOR OUTPUT

If the Noise Canceler function is enabled, the actual trigger condition for the capture event is monitored over four samples and all four must be equal to activate the capture flag.

Timer/Counter1 Control Register A – TCCR1A

Bit	7	6	5	4	3	2	1	0	
$2F ($4F)	COM1A1	COM1A0	COM1B1	COM1B0	–	–	PWM11	PWM10	TCCR1A
Read/Write	R/W	R/W	R/W	R/W	R	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bits 7, 6 – COM1A1, COM1A0: Compare Output Mode1A, Bits 1 and 0**

The COM1A1 and COM1A0 control bits determine any output pin action following a compare match in Timer/Counter1. Any output pin actions affect pin OC1A (Output CompareA pin 1). This is an alternative function to an I/O port and the corresponding direction control bit must be set (one) to control the output pin. The control configuration is shown in Table 8.

- **Bits 5, 4 – COM1B1, COM1B0: Compare Output Mode1B, Bits 1 and 0**

The COM1B1 and COM1B0 control bits determine any output pin action following a compare match in Timer/Counter1. Any output pin actions affect pin OC1B (Output CompareB). The control configuration is given in Table 8.

Table 8. 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 12 on page 40 for a detailed description.

- **Bits 3..2 – Res: Reserved Bits**

These bits are reserved bits in the AT90S8515 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 9. This mode is described on page 40.

Table 9. 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

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 for Timer/Counter1, transitions on PB1/(T1) will clock the counter even if the pin is configured as an output. This feature can give the user software control of the counting.

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 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	

1. In the same operation, write a logical “1” to WDTOE and WDE. A logical “1” must be written to WDE even though it is set to one before the disable operation starts.
2. Within the next four clock cycles, write a logical “0” to WDE. This disables the Watchdog.

- **Bits 2..0 – WDP2, WDP1, WDP0: Watchdog Timer Prescaler 2, 1 and 0**

The WDP2, WDP1 and WDP0 bits determine the Watchdog Timer prescaling when the Watchdog Timer is enabled. The different prescaling values and their corresponding Time-out periods are shown in Table 14.

Table 14. Watchdog Timer Prescale Select

WDP2	WDP1	WDP0	Number of WDT Oscillator Cycles	Typical Time-out at V_{CC} = 3.0V	Typical Time-out at V_{CC} = 5.0V
0	0	0	16K cycles	47.0 ms	15.0 ms
0	0	1	32K cycles	94.0 ms	30.0 ms
0	1	0	64K cycles	0.19 s	60.0 ms
0	1	1	128K cycles	0.38 s	0.12 s
1	0	0	256K cycles	0.75 s	0.24 s
1	0	1	512K cycles	1.5 s	0.49 s
1	1	0	1,024K cycles	3.0 s	0.97 s
1	1	1	2,048K cycles	6.0 s	1.9 s

Note: The frequency of the Watchdog oscillator is voltage-dependent as shown in the Electrical Characteristics section.
The WDR (Watchdog Reset) instruction should always be executed before the Watchdog Timer is enabled. This ensures that the reset period will be in accordance with the Watchdog Timer prescale settings. If the Watchdog Timer is enabled without reset, the Watchdog Timer may not start to count from zero.
To avoid unintentional MCU reset, the Watchdog Timer should be disabled or reset before changing the Watchdog Timer Prescale Select.

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	R/W	R/W	R/W	R/W	R/W	R/W	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.

UART Status Register – USR

Bit	7	6	5	4	3	2	1	0	
$0B ($2B)	RXC	TXC	UDRE	FE	OR	–	–	–	USR
Read/Write	R	R/W	R	R	R	R	R	R	
Initial Value	0	0	1	0	0	0	0	0	

The USR register is a read-only register providing information on the UART status.

- **Bit 7 – RXC: UART Receive Complete**

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

- **Bit 6 – TXC: 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 UDR. 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 TXCIE bit in UCR is set, setting of TXC causes the UART Transmit Complete interrupt to be executed. TXC is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, the TXC bit is cleared (zero) by writing a logical “1” to the bit.

- **Bit 5 – UDRE: UART Data Register Empty**

This bit is set (one) when a character written to UDR 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 UDRIE bit in UCR is set, the UART Transmit Complete interrupt to be executed as long as UDRE is set. UDRE is cleared by writing UDR. When interrupt-driven data transmittal is used, the UART Data Register Empty Interrupt routine must write UDR in order to clear UDRE, otherwise a new interrupt will occur once the interrupt routine terminates.

UDRE is set (one) during reset to indicate that the transmitter is ready.

- **Bit 4 – FE: 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.

Table 17. UBRR Settings at Various Crystal Frequencies

Baud Rate	1 MHz		%Error	1.8432 MHz		%Error	2 MHz		%Error	2.4576 MHz		%Error
2400	UBRR=	**25**	**0.2**	UBRR=	**47**	**0.0**	UBRR=	**51**	**0.2**	UBRR=	**63**	**0.0**
4800	UBRR=	**12**	**0.2**	UBRR=	**23**	**0.0**	UBRR=	**25**	**0.2**	UBRR=	**31**	**0.0**
9600	UBRR=	6	7.5	UBRR=	**11**	**0.0**	UBRR=	**12**	**0.2**	UBRR=	**15**	**0.0**
14400	UBRR=	3	7.8	UBRR=	**7**	**0.0**	UBRR=	8	3.7	UBRR=	10	3.1
19200	UBRR=	2	7.8	UBRR=	**5**	**0.0**	UBRR=	6	7.5	UBRR=	**7**	**0.0**
28800	UBRR=	1	7.8	UBRR=	**3**	**0.0**	UBRR=	3	7.8	UBRR=	4	6.3
38400	UBRR=	1	22.9	UBRR=	**2**	**0.0**	UBRR=	2	7.8	UBRR=	**3**	**0.0**
57600	UBRR=	0	7.8	UBRR=	**1**	**0.0**	UBRR=	1	7.8	UBRR=	2	12.5
76800	UBRR=	0	22.9	UBRR=	1	33.3	UBRR=	1	22.9	UBRR=	**1**	**0.0**
115200	UBRR=	0	84.3	UBRR=	**0**	**0.0**	UBRR=	0	7.8	UBRR=	0	25.0

Baud Rate	3.2768 MHz		%Error	3.6864 MHz		%Error	4 MHz		%Error	4.608 MHz		%Error
2400	UBRR=	**84**	**0.4**	UBRR=	**95**	**0.0**	UBRR=	**103**	**0.2**	UBRR=	**119**	**0.0**
4800	UBRR=	**42**	**0.8**	UBRR=	**47**	**0.0**	UBRR=	**51**	**0.2**	UBRR=	**59**	**0.0**
9600	UBRR=	**20**	**1.6**	UBRR=	**23**	**0.0**	UBRR=	**25**	**0.2**	UBRR=	**29**	**0.0**
14400	UBRR=	**13**	**1.6**	UBRR=	**15**	**0.0**	UBRR=	16	2.1	UBRR=	**19**	**0.0**
19200	UBRR=	10	3.1	UBRR=	**11**	**0.0**	UBRR=	**12**	**0.2**	UBRR=	**14**	**0.0**
28800	UBRR=	**6**	**1.6**	UBRR=	**7**	**0.0**	UBRR=	8	3.7	UBRR=	**9**	**0.0**
38400	UBRR=	4	6.3	UBRR=	**5**	**0.0**	UBRR=	6	7.5	UBRR=	7	6.7
57600	UBRR=	3	12.5	UBRR=	**3**	**0.0**	UBRR=	3	7.8	UBRR=	**4**	**0.0**
76800	UBRR=	2	12.5	UBRR=	**2**	**0.0**	UBRR=	2	7.8	UBRR=	3	6.7
115200	UBRR=	1	12.5	UBRR=	**1**	**0.0**	UBRR=	1	7.8	UBRR=	2	20.0

Baud Rate	7.3728 MHz		%Error	8 MHz		%Error	9.216 MHz		%Error	11.059 MHz		%Error
2400	UBRR=	**191**	**0.0**	UBRR=	**207**	**0.2**	UBRR=	**239**	**0.0**	UBRR=	287	-
4800	UBRR=	**95**	**0.0**	UBRR=	**103**	**0.2**	UBRR=	**119**	**0.0**	UBRR=	**143**	**0.0**
9600	UBRR=	**47**	**0.0**	UBRR=	**51**	**0.2**	UBRR=	**59**	**0.0**	UBRR=	**71**	**0.0**
14400	UBRR=	**31**	**0.0**	UBRR=	**34**	**0.8**	UBRR=	**39**	**0.0**	UBRR=	**47**	**0.0**
19200	UBRR=	**23**	**0.0**	UBRR=	**25**	**0.2**	UBRR=	**29**	**0.0**	UBRR=	**35**	**0.0**
28800	UBRR=	**15**	**0.0**	UBRR=	16	2.1	UBRR=	**19**	**0.0**	UBRR=	**23**	**0.0**
38400	UBRR=	**11**	**0.0**	UBRR=	**12**	**0.2**	UBRR=	**14**	**0.0**	UBRR=	**17**	**0.0**
57600	UBRR=	**7**	**0.0**	UBRR=	8	3.7	UBRR=	**9**	**0.0**	UBRR=	**11**	**0.0**
76800	UBRR=	**5**	**0.0**	UBRR=	6	7.5	UBRR=	7	6.7	UBRR=	**8**	**0.0**
115200	UBRR=	**3**	**0.0**	UBRR=	3	7.8	UBRR=	**4**	**0.0**	UBRR=	**5**	**0.0**

UART BAUD Rate Register – UBRR

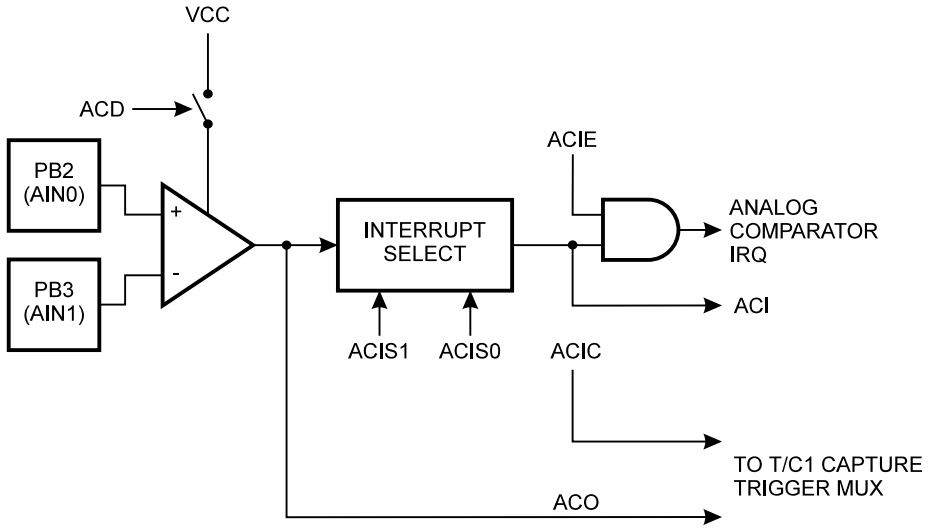
Bit	7	6	5	4	3	2	1	0	
$09 ($29)	**MSB**							**LSB**	**UBRR**
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

The UBRR register is an 8-bit read/write register that specifies the UART Baud Rate according to the equation on the previous page.

Analog Comparator

The Analog Comparator compares the input values on the positive input PB2 (AIN0) and negative input PB3 (AIN1). When the voltage on the positive input PB2 (AIN0) is higher than the voltage on the negative input PB3 (AIN1), the Analog Comparator Output (ACO) is set (one). The comparator's output can be set to trigger the Timer/Counter1 Input Capture function. In addition, the comparator can trigger a separate interrupt, exclusive to the Analog Comparator. The user can select interrupt triggering on comparator output rise, fall or toggle. A block diagram of the comparator and its surrounding logic is shown in Figure 41.

Figure 41. Analog Comparator Block Diagram



Analog Comparator Control and Status Register – ACSR

Bit	7	6	5	4	3	2	1	0	
$08 ($28)	ACD	–	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	ACSR
Read/Write	R/W	R	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	N/A	0	0	0	0	0	

- **Bit 7 – ACD: Analog Comparator Disable**

When this bit is set (one), the power to the Analog Comparator is switched off. This bit can be set at any time to turn off the Analog Comparator. This will reduce power consumption in active and idle mode. When changing the ACD bit, the Analog Comparator interrupt must be disabled by clearing the ACIE bit in ACSR. Otherwise an interrupt can occur when the bit is changed.

- **Bit 6 – Res: Reserved Bit**

This bit is a reserved bit in the AT90S8515 and will always read as zero.

- **Bit 5 – ACO: Analog Comparator Output**

ACO is directly connected to the comparator output.

- **Bit 4 – ACI: Analog Comparator Interrupt Flag**

This bit is set (one) when a comparator output event triggers the interrupt mode defined by ACI1 and ACI0. The Analog Comparator Interrupt routine is executed if the ACIE bit is set (one) and the I-bit in SREG is set (one). ACI is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, ACI is cleared by writing a logical "1" to the flag. Observe however, that if another bit in this register is modified

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

Table 21. 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.

- **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.

Table 27. 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)
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 $\overline{OE}$ is low)

Table 28. 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 29. 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 26, between V_{CC} and GND.
2. Set the $\overline{RESET}$ and BS pin to “0” and wait at least 100 ns.
3. Apply 11.5 - 12.5V to $\overline{RESET}$. Any activity on BS within 100 ns after +12V has been applied to $\overline{RESET}$ will cause the device to fail entering programming mode.

Chip Erase

The Chip Erase command will erase the Flash and EEPROM memories and the Lock bits. The Lock bits are not reset until the Flash and EEPROM have been completely erased. The Fuse bits are not changed. Chip Erase must be performed before the Flash or EEPROM is reprogrammed.

Load Command “Chip Erase”

1. Set XA1, XA0 to “10”. This enables command loading.
2. Set BS to “0”.
3. Set DATA to “1000 0000”. This is the command for Chip Erase.
4. Give XTAL1 a positive pulse. This loads the command.
5. Give $\overline{WR}$ a t_{WLWH_CE}-wide negative pulse to execute Chip Erase. See Table 30 on page 85 for t_{WLWH_CE} value. Chip Erase does not generate any activity on the RDY/$\overline{BSY}$ pin.

Programming the Flash

A: Load Command “Write Flash”

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

B: Load Address High Byte

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

C: Load Address Low Byte

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

D: Load Data Low Byte

1. Set XA1, XA0 to “01”. This enables data loading.
2. Set DATA = Data low byte ($00 - $FF).
3. Give XTAL1 a positive pulse. This loads the data low byte.

E: Write Data Low Byte

1. Set BS to “0”. This selects low data.
2. Give $\overline{WR}$ a negative pulse. This starts programming of the data byte. RDY/$\overline{BSY}$ goes low.
3. Wait until RDY/$\overline{BSY}$ goes high to program the next byte.

(See Figure 61 for signal waveforms.)

F: Load Data High Byte

1. Set XA1, XA0 to “01”. This enables data loading.
2. Set DATA = Data high byte ($00 - $FF).
3. Give XTAL1 a positive pulse. This loads the data high byte.

G: Write Data High Byte

Bit 5 = SPIEN Fuse bit

Bit 0 = FSTRT Fuse bit

Bit 7 - 6, 4 - 1 = “1”. These bits are reserved and should be left unprogrammed (“1”).

3. Give $\overline{WR}$ a t_{WLWH_PFB}-wide negative pulse to execute the programming, t_{WLWH_PFB} is found in Table 30. Programming the Fuse bits does not generate any activity on the RDY/$\overline{BSY}$ pin.

Programming the Lock Bits

The algorithm for programming the Lock bits is as follows (refer to “Programming the Flash” on page 81 for details on command and data loading):

1. A: Load Command “0010 0000”.
2. D: Load Data Low Byte. Bit n = “0” programs the Lock bit.

   Bit 2 = Lock Bit2

   Bit 1 = Lock Bit1

   Bit 7 - 3, 0 = “1”. These bits are reserved and should be left unprogrammed (“1”).
3. E: Write Data Low Byte.

The Lock bits can only be cleared by executing Chip Erase.

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 81 for details on Command loading):

1. A: Load Command “0000 0100”.
2. Set $\overline{OE}$ to “0”, and BS to “1”. The status of the Fuse and Lock bits can now be read at DATA (“0” means programmed).

   Bit 7 = Lock Bit1

   Bit 6 = Lock Bit2

   Bit 5 = SPIEN Fuse bit

   Bit 0 = FSTRT Fuse bit
3. Set $\overline{OE}$ to “1”.

Observe that BS needs to be set to “1”.

Reading the Signature Bytes

The algorithm for reading the signature bytes is as follows (refer to “Programming the Flash” on page 81 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”.

Table 39. External Data Memory Characteristics, 2.7V - 4.0V, No Wait State

			4 MHz Oscillator		Variable Oscillator		
	Symbol	Parameter	Min	Max	Min	Max	Unit
0	$1/t_{CLCL}$	Oscillator Frequency			0.0	4.0	MHz
1	t_{LHLL}	ALE Pulse Width	70.0		0.5 t_{CLCL} - 55.0(1)		ns
2	t_{AVLL}	Address Valid A to ALE Low	60.0		0.5 t_{CLCL} - 65.0(1)		ns
3a	t_{LLAX_ST}	Address Hold after ALE Low, ST/STD/STS Instructions	130.0		0.5 t_{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	60.0		0.5 t_{CLCL} - 65.0(1)		ns
5	t_{AVRL}	Address Valid to RD Low	200.0		1.0 t_{CLCL} - 50.0		ns
6	t_{AVWL}	Address Valid to WR Low	325.0		1.5 t_{CLCL} - 50.0(1)		ns
7	t_{LLWL}	ALE Low to WR Low	230.0	270.0	1.0 t_{CLCL} - 20.0	1.0 t_{CLCL} + 20.0	ns
8	t_{LLRL}	ALE Low to RD Low	105.0	145.0	0.5 t_{CLCL} - 20.0(2)	0.5 t_{CLCL} + 20.0(2)	ns
9	t_{DVRH}	Data Setup to RD High	95.0		95.0		ns
10	t_{RLDV}	Read Low to Data Valid		170.0		1.0 t_{CLCL} - 80.0	ns
11	t_{RHDX}	Data Hold after RD High	0.0		0.0		ns
12	t_{RLRH}	RD Pulse Width	230.0		1.0 t_{CLCL} - 20.0		ns
13	t_{DVWL}	Data Setup to WR Low	70.0		0.5 t_{CLCL} - 55.0(1)		ns
14	t_{WHDX}	Data Hold after WR High	0.0		0.0		ns
15	t_{DVWH}	Data Valid to WR High	210.0		1.0 t_{CLCL} - 40.0		ns
16	t_{WLWH}	WR Pulse Width	105.0		0.5 t_{CLCL} - 20.0(2)		ns

Table 40. External Data Memory Characteristics, 2.7V - 4.0V, One Cycle Wait State

			4 MHz Oscillator		Variable Oscillator		
	Symbol	Parameter	Min	Max	Min	Max	Unit
0	$1/t_{CLCL}$	Oscillator Frequency			0.0	4.0	MHz
10	t_{RLDV}	Read Low to Data Valid		420.00		2.0 t_{CLCL} - 80.0	ns
12	t_{RLRH}	RD Pulse Width	480.0		2.0 t_{CLCL} - 20.0		ns
15	t_{DVWH}	Data Valid to WR High	460.0		2.0 t_{CLCL} - 40.0		ns
16	t_{WLWH}	WR Pulse Width	355.0		1.5 t_{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.

Figure 78. Analog Comparator Input Leakage Current

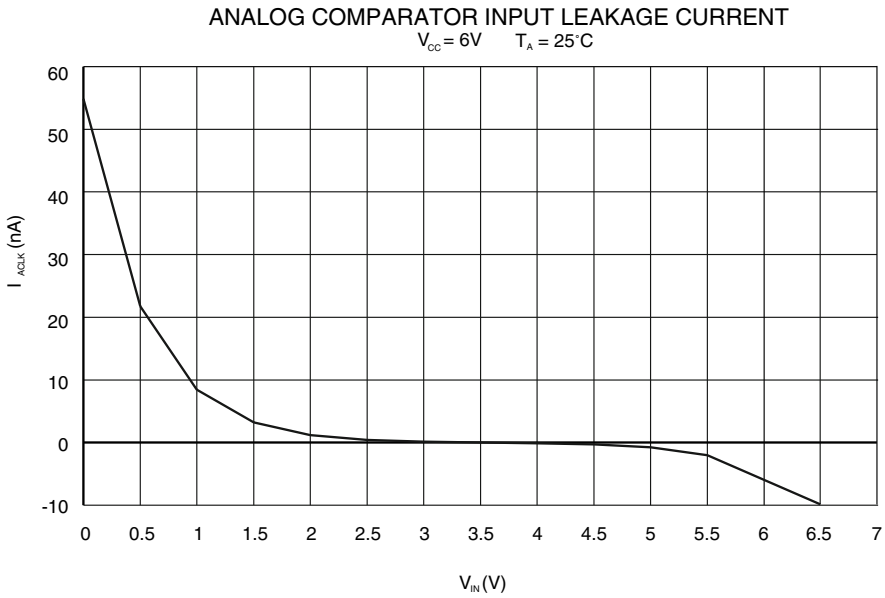


Figure 79. Watchdog Oscillator Frequency vs. V_{CC}

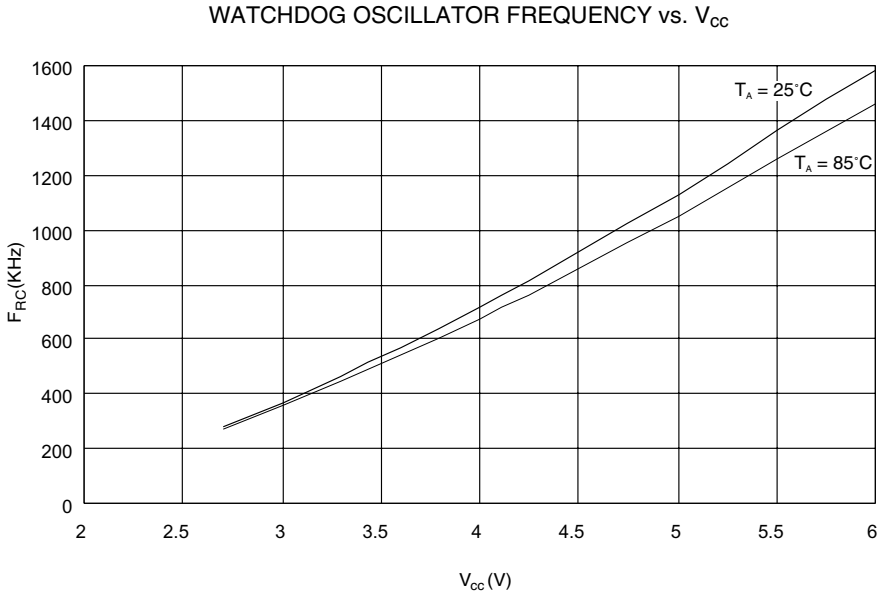


Figure 86. I/O Pin Input Hysteresis vs. V_{CC}

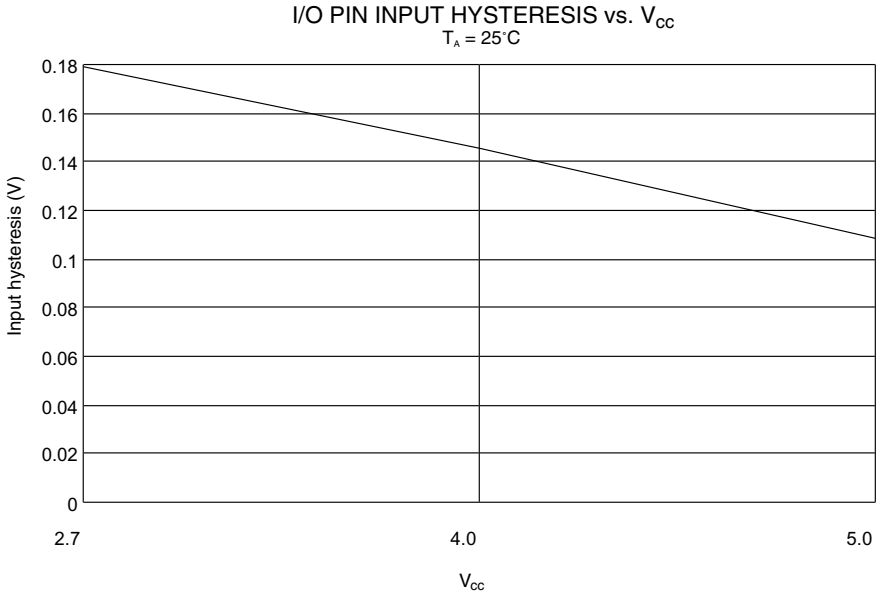


Figure 87. I/O Pin Sink Current vs. Output Voltage