Contents

8-Bit AVR Microcontroller with 2K bytes Downloadable Flash

Features

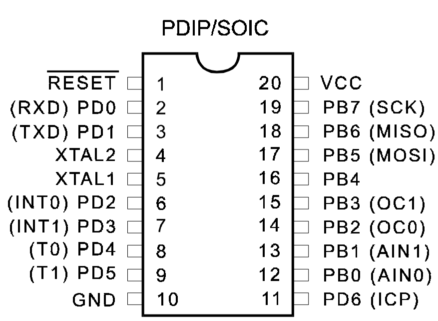
- **Utilizes the AV*R*™ Enhanced RISC Architecture**
- **AV*R*™ - High Performance and Low Power RISC Architecture**
- **118 Powerful Instructions - Most Single Clock Cycle Execution**
- **Efficient Context Switching Concept**
- **2 Kbytes of In-System Reprogrammable Downloadable Flash**
  SPI Serial Interface for Program Downloading
  Endurance: 1,000 Write/Erase Cycles
- **128 bytes EEPROM**
  Endurance: 100,000 Write/Erase Cycles
- **64 bytes Internal RAM**
- **32 x 8 General Purpose Working Registers**
- **15 Programmable I/O Lines**
- **VCC Min.: 2.7 V**
- **Fully Static Operation**
- **One 8-Bit Timer/Counter with Separate Prescaler**
- **One 16-Bit Timer/Counter with Separate Prescaler and Compare and Capture Modes**
- **Full Duplex UART**
- **10 bit PWM**
- **External and Internal Interrupt Sources**
- **Programmable Watchdog Timer**
- **On-Chip Analog Comparator**
- **Low Power Idle and Power Down Modes**
- **Programming lock for Software Security**
- **20-Pins Device**

Description

The AT90S2312 is a low-power CMOS 8 bit microcontroller based on the *AVR* enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the AT90S2312 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

(continued)

Pin Configuration



Block Diagram

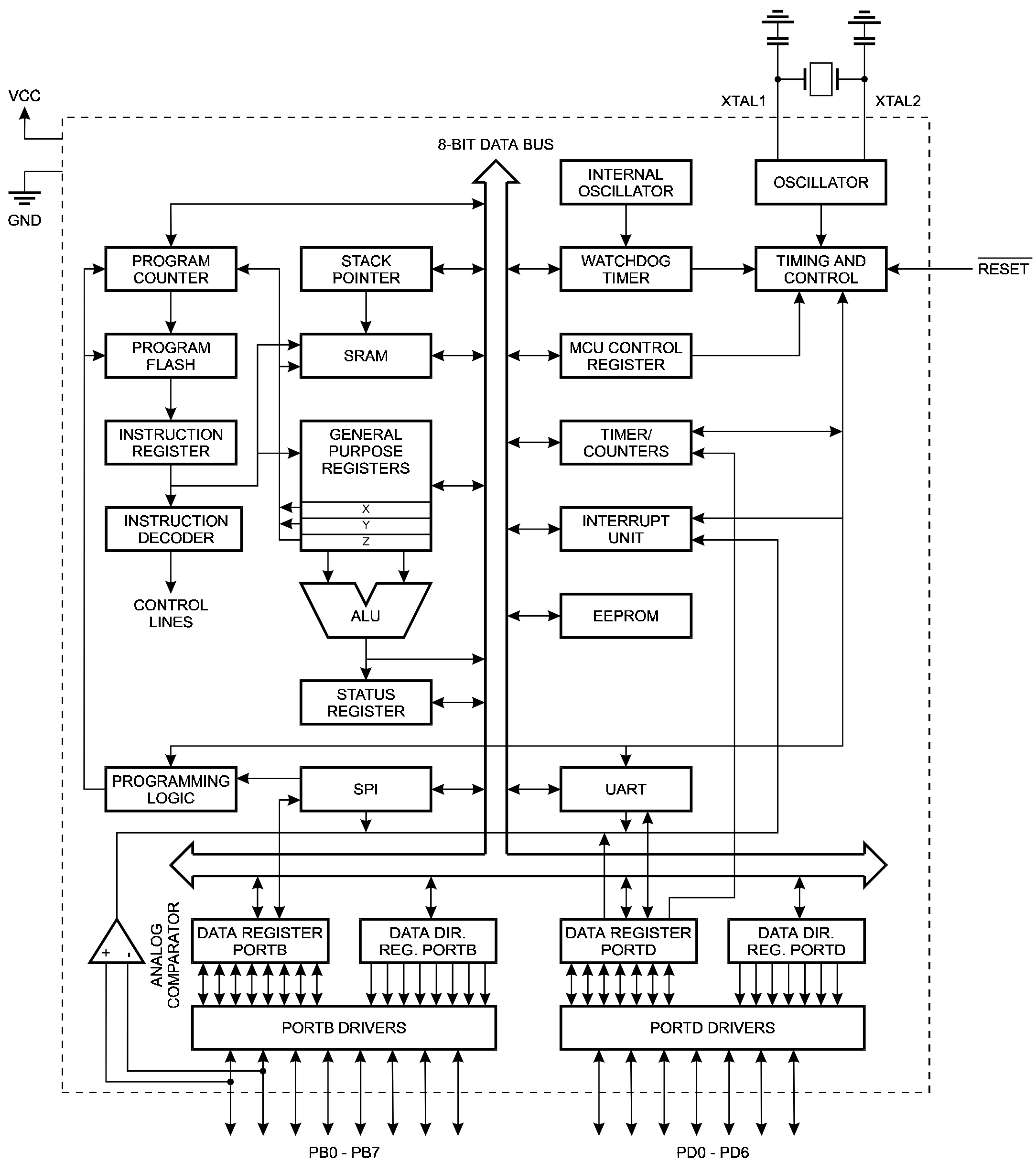


Figure 1: The AT90S2312 Block Diagram

Description (Continued)

The *AVR* core is based on an enhanced RISC architecture that combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers. The architecture supports high level languages efficiently while on-chip context switching hardware allows high performance, low power real timer control system design.

The AT90S2312 provides the following features: 2K bytes of Downloadable Flash, 128 bytes EEPROM, 64-bytes SRAM, 15 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 for Flash Memory downloading 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 interrupt or hardware reset.

The device is manufactured using Atmel's high density non-volatile memory technology. The on-chip Downloadable 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 Downloadable Flash on a monolithic chip, the Atmel AT90S2312 is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications.

The AT90S2312 *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 pin.

GND
Ground pin.

Port B (PB7..PB0)
Port B is an 8-bit bi-directional I/O port. Port pins can provide internal pullups (selected for each bit). PB0 and PB1 also serve as the positive input (AIN0) and the negative input (AIN1), respectively, of the on-chip analog comparator. The Port B output buffers can sink 20mA and can drive LED displays directly. When pins PB0 to PB7 are used as inputs and are externally pulled low, they will source current (IIL) if the internal pullups are activated.

Port B also serves the functions of various special features of the AT90S2312 as listed on Page 3-48.

Port D (PD6..PD0)
Port D has seven bi-directional I/O pins with internal pullups, PD6..PD0. The Port D output buffers can sink 20 mA. As inputs, Port D pins that are externally pulled low will source current (IIL) if the pullups are activated.

Port D also serves the functions of various special features of the AT90S2312 as listed on Page 3-54.

$\overline{\text{RESET}}$
Reset input. A low on this pin for two machine cycles while the oscillator is running resets the device.

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

XTAL2
Output from the inverting oscillator amplifier

Crystal Oscillator

XTAL1 and XTAL2 are input and output, respectively , of an inverting amplifier which 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.

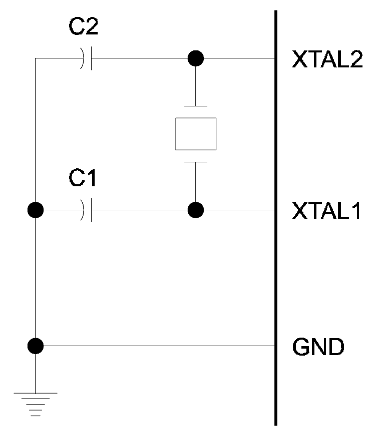


Figure 2: Oscillator Connections

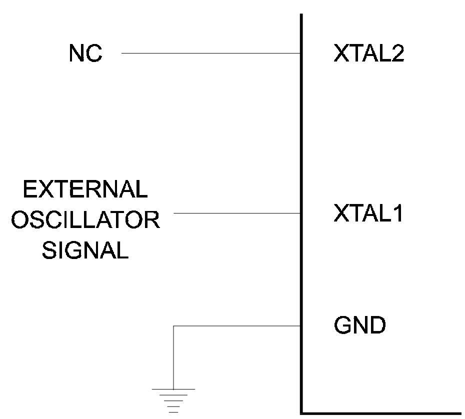


Figure 3: External Clock Drive Configuration

AT90S2312 *AVR* Enhanced RISC Microcontroller CPU

The AT90S2312 *AVR* RISC microcontroller is upward compatible with the *AVR* Enhanced RISC Architecture. The programs written for the AT90S2312 MCU are fully compatible with the range of *AVR* 8-bit MCUs (AT90Sxxxx) with respect to source code and clock cycles for execution.

Architectural Overview

The fast-access register file concept contains 32 x 8-bit general purpose working registers with a single clock cycle access time. This means that during one single clock cycle, one ALU (Arithmetic Logic Unit) operation is executed. Two operands are output from the register file, the operation is executed, and the result is stored back in the register file - in one clock cycle.

Six of the 32 registers can be used as three 16-bits indirect address register pointers for SRAM addressing - enabling efficient address calculations. One of the three address pointers is also used as the address pointer for the constant table look up function. These added function registers are the 16-bits X-register, Y-register and Z-register.

The ALU supports arithmetic and logic functions between registers or between a constant and a register. Single register operations are also executed in the ALU. Figure 4 shows the AT90S2312 *AVR* Enhanced RISC microcontroller architecture.

In addition to the register operation, the conventional memory addressing modes can be used on the register file as well. This is enabled by the fact that the register file is assigned the 32 lowermost SRAM addresses, allowing them to be accessed as though they were ordinary memory locations.

The *AVR* has Harvard architecture - with separate memories and buses for program and data. The program memory is accessed with a single level pipelining. While one instruction is being executed, the next instruction is pre-fetched from the program memory. This concept enables instructions to be executed in every clock cycle. The program memory is in-system downloadable Flash memory.

With the relative jump and call instructions, the whole 1K address space is directly accessed. All *AVR* instructions have a single 16-bit word format, meaning that every program memory address contains a single 16-bit instruction.

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

The 64 bytes data SRAM can be easily accessed through the four different addressing modes supported in the *AVR* architecture.

The I/O memory space contains 64 addresses for CPU peripheral functions as Control Registers, Timer/Counters, A/D-converters, and other I/O functions. The memory spaces in the *AVR* architecture are all linear and regular memory maps.

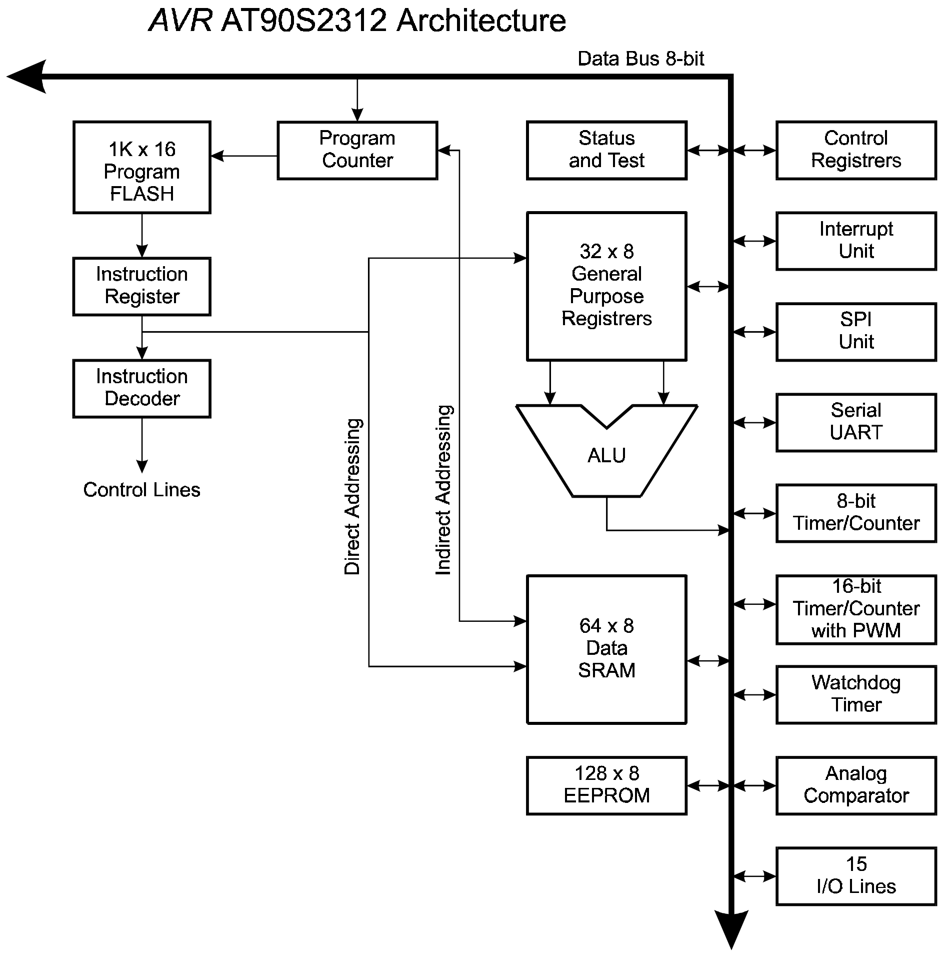


Figure 4: The AT90S2312 *AVR* Enhanced RISC Architecture

A flexible interrupt module has its control registers in the I/O space with an additional global interrupt enable bit in the status register. All the different interrupts have a separate interrupt vector in the interrupt vector table at the beginning of the program memory. The different interrupts have priority in accordance with their interrupt vector position. The lower the interrupt address vector the higher priority.

The General Purpose Register File

Figure 5 shows the structure of the 32 general purpose registers in the CPU.

	7 0	Addr.	
General Purpose Working Registers	R0	$00	
	R1	$01	
	R2	$02	
	…		
	R13	$0D	
	R14	$0E	
	R15	$0F	
	R16	$10	
	R17	$11	
	…		
	R26	$1A	X-register low byte
	R27	$1B	X-register high byte
	R28	$1C	Y-register low byte
	R29	$1D	Y-register high byte
	R30	$1E	Z-register low byte
	R31	$1F	Z-register high byte

Figure 5: ***AVR*** **CPU General Purpose Working Registers**

All the register operating instructions in the instruction set have direct and single cycle access to all registers. The only exception is the five constant arithmetic and logic instructions SBCI, SUBI, CPI, ANDI, ORI between a constant and a register and the LDI instruction for load immediate constant data. These instructions apply to the second half of the registers in the register file - R16..R31. The general SBC, SUB, CP, AND, OR and all other operations between two registers or on a single register apply to the entire register file.

As shown Figure 5, each register is also assigned a data memory address, mapping them directly into the first 32 locations of the user SRAM area. Although not being physically implemented as SRAM locations, this memory organization provides great flexibility in access of the registers, as the X , Y and Z registers can be set to index any register in the file.

The 64 bytes of SRAM available for general data are implemented as addresses $20 to $5F.

THE X-REGISTER, Y-REGISTER AND Z-REGISTER

The registers R26..R31 have some added functions to their general purpose usage. These registers are the address pointers for indirect addressing of the SRAM. The three indirect address registers X, Y and Z are defined as:

	15			0
X - register	7	0	7	0
	R27 ($1B)		R26 ($1A)	
	15			0
Y - register	7	0	7	0
	R29 ($1D)		R28 ($1C)	
	15			0
Z - register	7	0	7	0
	R31 ($1F)		R30 ($1E)	

Figure 6: The X, Y and Z Registers

In the different addressing modes these address registers have functions as fixed displacement, automatic increment and decrement (see the descriptions for the different instructions).

The 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, logic and bit-functions. Some microcontrollers in the *AVR* product family feature a hardware multiplier in the arithmetic part of the ALU.

The Downloadable Flash Program Memory

The AT90S2312 contains 2K bytes on-chip downloadable Flash memory for program storage. Since all instructions are single 16-bit words, the Flash is organized as 1K x 16 words. The Flash memory has an endurance of at least 1000 write/erase cycles.

The AT90S2312 Program Counter PC is 10 bits wide, thus addressing the 1024 program memory addresses.

See Page 3-58 for a detailed description on Flash data downloading.

Constant tables must be allocated within the address 0-2K (see the LPM - Load Program Memory instruction description).

See Page 3-11 for the different addressing modes.

The EEPROM Data Memory

The AT90S2312 contains 128 bytes of EEPROM data memory. It is organized as a separate data space, in which single bytes can be read and written. The EEPROM has an endurance of at least 100,000 write/erase cycles. The access between the EEPROM and the CPU is described on Page 3-38 specifying the EEPROM address register, the EEPROM data register, and the EEPROM control register.

For the SPI data downloading, see Page 3-58 for a detailed description.

The SRAM Data Memory

The following figure shows how the AT90S2312 SRAM Memory is organized:

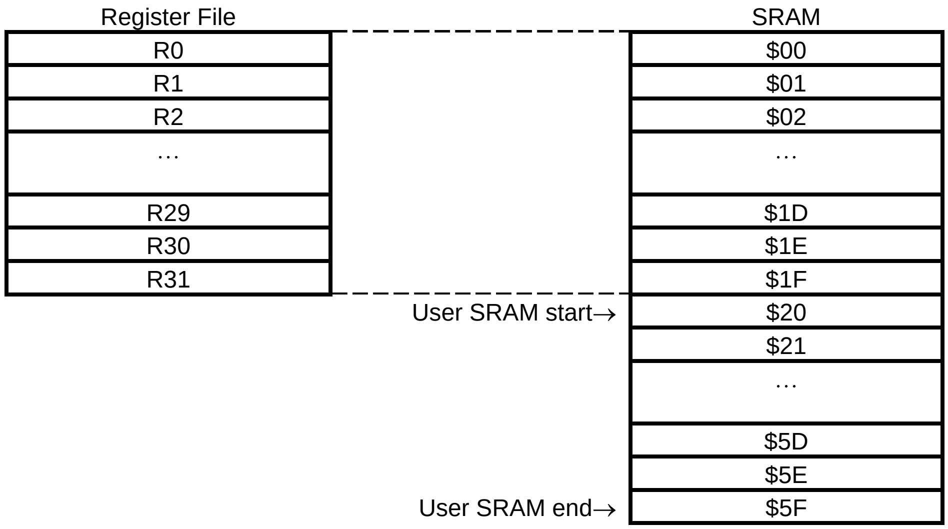


Figure 7: SRAM Organization

The 96 SRAM Memory locations address both the Register file and the data SRAM. The first 32 locations address the register file, and the next 64 locations address the data SRAM.

The four different addressing modes for the SRAM data memory cover: Direct 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 with Displacement mode features 63 address locations reach from the base address given by the Y and Z register.

When using register indirect addressing modes with automatic pre-decrement and post-increment, the address registers X, Y and Z are used and decremented and incremented.

The 32 registers plus the 64 bytes of data SRAM in the AT90S2312 are all directly accessible through all these addressing modes.

See Page 3-11 for a detailed description of the different addressing modes.

The Program and Data Addressing Modes

The AT90S2312 *AVR* Enhanced RISC Microcontroller supports powerful and efficient addressing modes for access to the program memory (Flash) and data memory (SRAM). 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

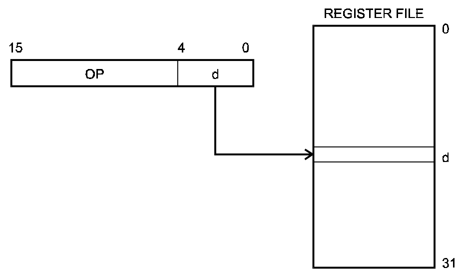


Figure 8: Direct Single Register Addressing

The operand is contained in register d (Rd).

REGISTER DIRECT, TWO REGISTERS Rd AND Rr

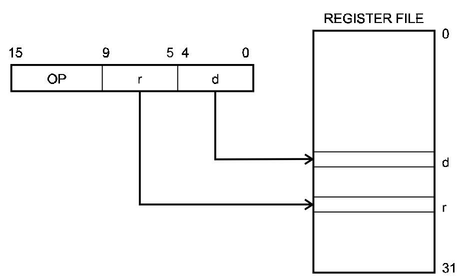


Figure 9: Direct Register Addressing, Two Registers

Operands are contained in register r (Rr) and d (Rd). The result is stored in register d (Rd).

I/O DIRECT

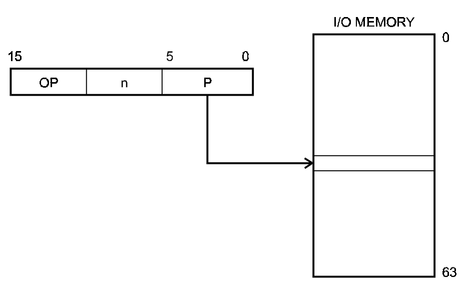


Figure 10: I/O Direct Addressing

Operand address is contained in 6 bits of the instruction word. n is the destination or source register address.

SRAM DIRECT WITH DISPLACEMENT

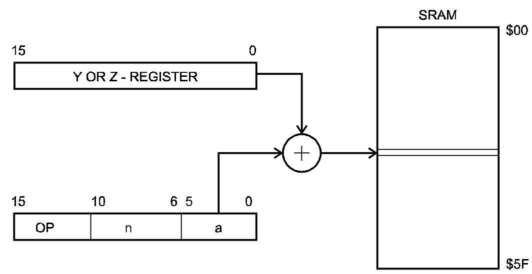


Figure 11: SRAM Direct with Displacement

Operand address is the result of the Y or Z-register contents added to the address contained in 6 bits of the instruction word.

SRAM/REGISTER INDIRECT

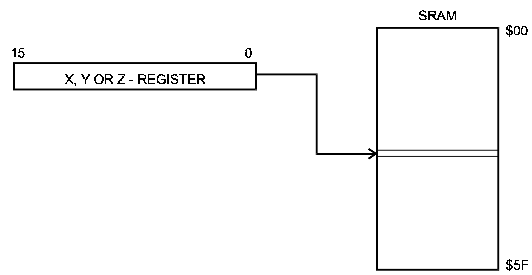


Figure 12: SRAM Indirect Addressing

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

SRAM/REGISTER INDIRECT WITH PRE-DECREMENT

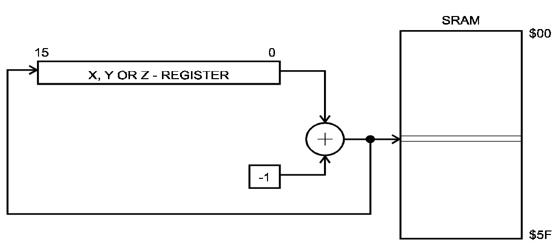


Figure 13: SRAM Indirect Addressing With Pre-Decrement

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

SRAM/REGISTER INDIRECT WITH POST-INCREMENT

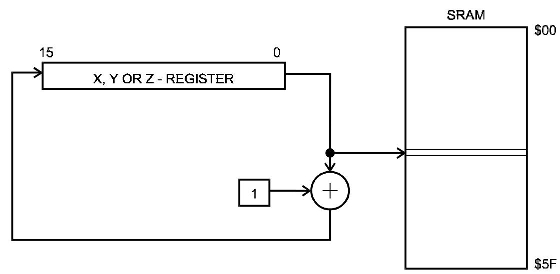


Figure 14: SRAM Indirect Addressing With Post-Increment

The X, Y or the Z-register is incremented after the operation. Operand address is the content of the X, Y or the Z-register prior to incrementing.

CONSTANT ADDRESSING USING THE LPM INSTRUCTION

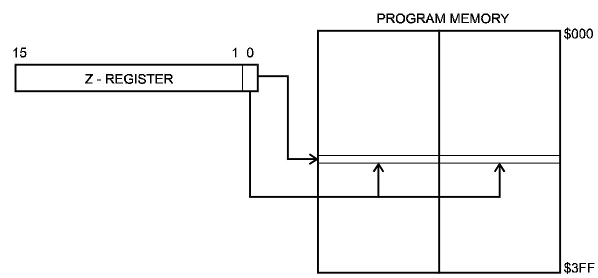


Figure 15: Code Memory Constant Addressing

Constant byte address is specified by the Z-register contents. The 15 MSBs select word address (0 - 1K) and LSB, select low byte if cleared (LSB = 0) or high byte if set (LSB = 1).

INDIRECT PROGRAM ADDRESSING, IJMP AND ICALL

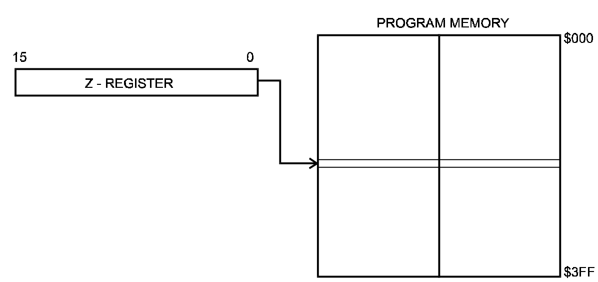


Figure 16: Indirect Program Memory Addressing

Program execution continues at address contained by the Z-register (i.e. the PC is loaded with the content of the Z-register).

RELATIVE PROGRAM ADDRESSING, RJMP AND RCALL

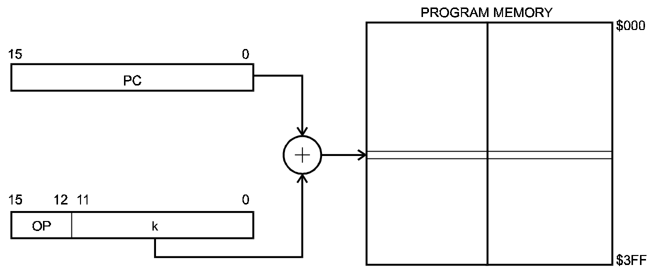


Figure 17: Relative Program Memory Addressing

Program execution continues at address PC + k. The relative address k is in the range from -2K to +(2K - 1).

Memory Access and Instruction Execution Timing

This section describes the general access timing concepts for instruction execution and internal memory access.

The *AVR* CPU is driven by the System Clock Ø, directly generated from the external clock crystal for the chip. No internal clock division is used.

Figure 18 shows the parallel instruction fetches and instruction executions enabled by the Harvard architecture and the fast-access register file concept. This is the basic pipelining concept to obtain up to 1 MIPS per MHz with the corresponding unique results for functions per cost, functions per clocks, and functions per power-unit.

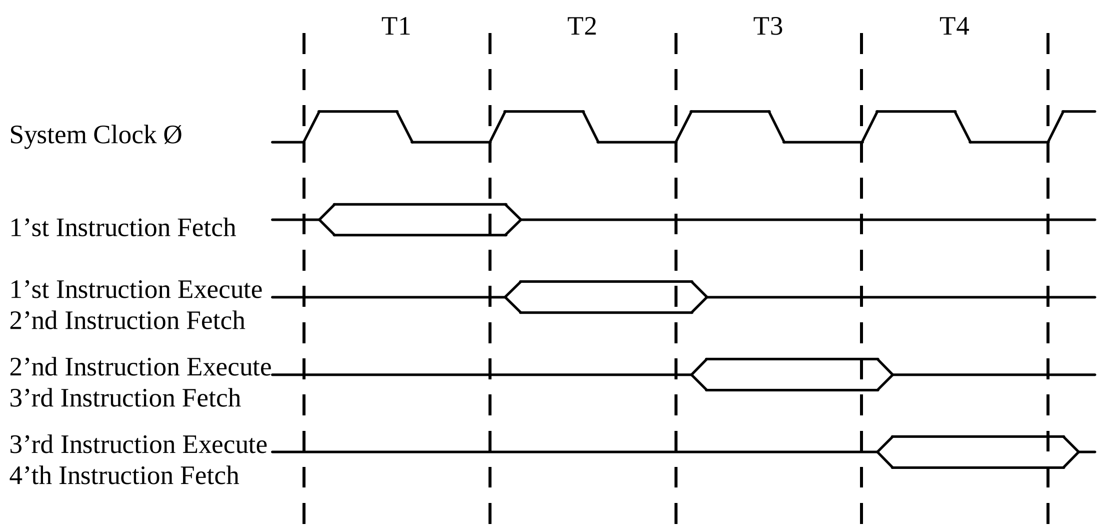


Figure 18: The Parallel Instruction Fetches and Instruction Executions

Figure 19 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.

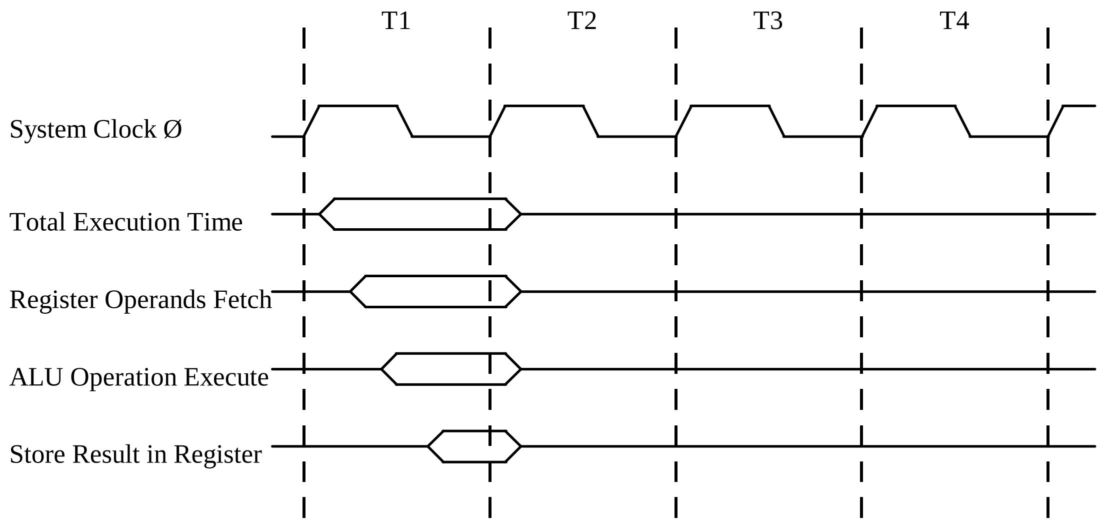


Figure 19: Single Cycle ALU Operation

The internal data SRAM access is performed in two System Clock cycles as described in Figure 20.

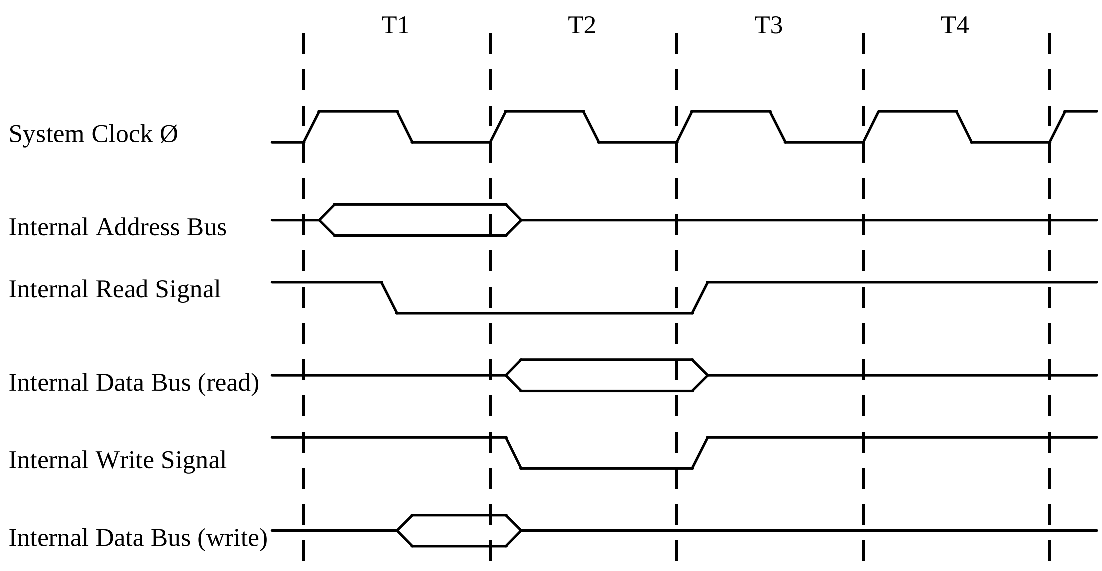


Figure 20: On-Chip Data SRAM Access Cycles

I/O Memory

The I/O space definition of the AT90S2312 is shown in the following table:

Table 1: AT90S2312 I/O Space

Address Hex	Name	Function
$3F	SREG	Status REGister
$3E	SPH	Stack Pointer High
$3D	SPL	Stack Pointer Low
$3B	GIMSK	General Interrupt MaSK register
$39	TIMSK	Timer/Counter Interrupt MaSK register
$38	TIFR	Timer/Counter Interrupt Flag register
$35	MCUCR	MCU general Control Register
$33	TCCR0	Timer/Counter 0 Control Register
$32	TCNT0	Timer/Counter 0 (8-bit)
$31	OCR0	Output Compare Register 0
$2F	TCCR1A	Timer/Counter 1 Control Register A
$2E	TCCR1B	Timer/Counter 1 Control Register B
$2D	TCNT1H	Timer/Counter 1 High Byte
$2C	TCNT1L	Timer/Counter 1 Low Byte
$2B	OCR1H	Output Compare Register 1 High Byte
$2A	OCR1L	Output Compare Register 1 Low Byte
$25	ICR1H	T/C 1 Input Capture Register High Byte
$24	ICR1L	T/C 1 Input Capture Register Low Byte
$21	WDTCR	Watchdog Timer Control Register
$1E	EEAR	EEPROM Address Register
$1D	EEDR	EEPROM Data Register
$1C	EECR	EEPROM Control Register
$18	PORTB	Data Register, Port B
$17	DDRB	Data Direction Register, Port B
$16	PINB	Input Pins, Port B
$12	PORTD	Data Register, Port D
$11	DDRD	Data Direction Register, Port D
$10	PIND	Input Pins, Port D
$0C	UDR	UART I/O Data Register
$0B	USR	UART Status Register
$0A	UCR	UART Control Register
$09	UBRR	UART Baud Rate Register
$08	ACSR	Analog Comparator Control and Status Register

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

All the different AT90S2312 I/O and peripherals are placed in the I/O space. The different 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 - $19 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.

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

THE STATUS REGISTER - SREG

The *AVR* status register - SREG - at I/O space location $3F is defined as:

Bit	7	6	5	4	3	2	1	0	
$3F	**I**	**T**	**H**	**S**	**V**	**N**	**Z**	**C**	**SREG**
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	

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 the interrupt mask registers - GIMSK and TIMSK. If the global interrupt enable register is cleared (zero), none of the interrupts are enabled independent of the GIMSK and TIMSK values. 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.

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, $S = N \oplus V$:

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.

THE 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 and $3D. Since the stack address space is within the data SRAM, a 7 bit stack pointer is used to address the stack in the AT90S2312.

Bit	15	14	13	12	11	10	9	8	
$3E	-	-	-	-	-	-	-	-	**SPH**
$3D	-	**SP6**	**SP5**	**SP4**	**SP3**	**SP2**	**SP1**	**SP0**	**SPL**
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R	R	R	R	
	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	
	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 is decremented by one when data is pushed onto the Stack with the PUSH instruction, and it is decremented by two when data is pushed onto the Stack with subroutine CALL and interrupt. The Stack Pointer is incremented by one when data is popped from the Stack with the POP instruction, and it is incremented by two when data is popped from the Stack with return from subroutine RET or return from interrupt IRET.

Reset and Interrupt Handling

The AT90S2312 provides 11 different interrupt sources. These interrupts and the separate reset vector, each have a separate program vector in the program memory space. All the interrupts are assigned individual enable bits which must be set (one) together with the I-bit in the status register in order to enable the interrupt.

The lowest addresses in the program memory space are automatically defined as the Reset and Interrupt vectors. The complete list of vectors is shown in Table 2. The list also determines the priority levels of the different interrupts. The lower the address the higher is the priority level. RESET has the highest priority, and next is INT0 - the External Interrupt Request 0 etc.

Table 2: Reset and Interrupt Vectors

Vector No.	Program Address	Source	Interrupt Definition
1	$000	RESET	Hardware Pin and Watchdog Reset
2	$001	INT0	External Interrupt Request 0
3	$002	INT1	External Interrupt Request 1
4	$003	TIMER1 CAPT1	Timer/Counter1 Capture Event
5	$004	TIMER1 COMP1	Timer/Counter1 Compare Match
6	$005	TIMER1 OVF1	Timer/Counter1 Overflow
7	$006	TIMER0, COMP0	Timer/Counter0 Compare Match
8	$007	TIMER0, OVF0	Timer/Counter0 Overflow
9	$008	UART, RX	UART, Rx Complete
10	$009	UART, UDRE	UART Data Register Empty
11	$00A	UART, TX	UART, Tx Complete
12	$00B	ANA_COMP	Analog Comparator

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

```
Address       Labels       Code                Comments

$000                       rjmp   RESET        ; Reset Handle
$001                       rjmp   EXT_INT0     ; IRQ0 Handle
$002                       rjmp   EXT_INT1     ; IRQ1 Handle
$003                       rjmp   TIM_CAPT1    ; Timer1 capture Handle
$004                       rjmp   TIM_COMP1    ; Timer1 compare Handle
$005                       rjmp   TIM_OVF1     ; Timer1 overflow Handle
$006                       rjmp   TIM_COMP0    ; Timer0 compare Handle
$007                       rjmp   TIM_OVF0     ; Timer0 overflow Handle
$008                       rjmp   UART_RXC     ; UART RX Complete Handle
$009                       rjmp   UART_DRE     ; UDR Empty Handle
$00a                       rjmp   UART_TXC     ; UART TX Complete Handle
$00b                       rjmp   ANA_COMP     ; Analog Comparator Handle
;
$00c          MAIN:        <instr>  xxx        ; Main program start
  …             …            …       …
```

RESET SOURCES

The AT90S2312 has three sources of reset:

- Power-On Reset. The MCU is reset when a supply voltage is applied to the VCC and GND pins.
- External Reset. The MCU is reset when a low level is present on the $\overline{\text{RESET}}$ pin for more than two XTAL cycles
- Watchdog Reset. The MCU is reset when the Watchdog timer period expires and the Watchdog is enabled.

During reset, all I/O registers are then set to their initial values, and the program starts execution from address $000. The instruction placed in address $000 must be an RJMP - relative jump - instruction to the reset handling routine. If the program never enables an interrupt source, the interrupt vectors are not used, and regular program code can be placed at these locations. The circuit diagram in Figure 21 shows the reset logic. Table 3 defines the timing and electrical parameters of the reset circuitry.

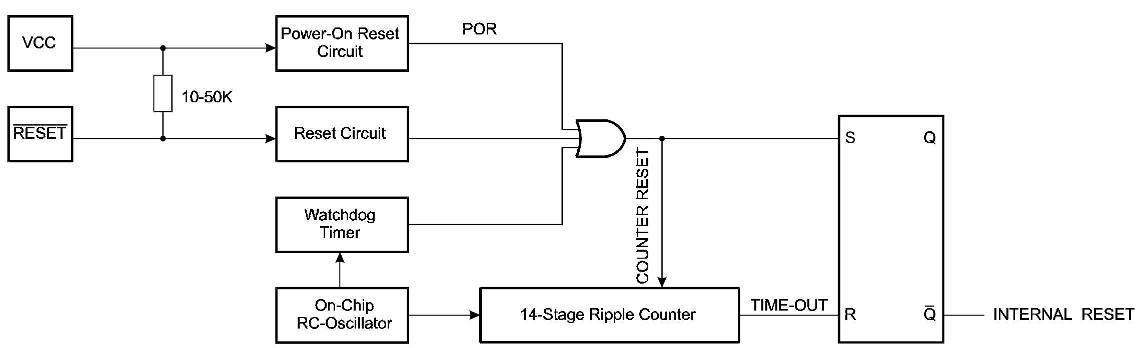


Figure 21: Reset Logic

Table 3: Reset Characteristics

Symbol	Parameter	Min	Typ	Max	Units
V_{POT}	Power-On Reset Threshold Voltage	1.8	2	2.2	V
V_{RST}	$\overline{\text{RESET}}$ Pin Threshold Voltage		VCC/2		V
T_{POR}	Power-On Reset Period	2	3	4	ms
T_{TOUT}	Reset Delay Time-Out Period	11	16	21	ms

POWER-ON RESET

A Power-On Reset (POR) circuit ensures that the device is not started until the XTAL oscillator has stabilized. As shown in Figure 21, an internal timer clocked from the Watchdog timer oscillator prevents the MCU from starting until after a certain period after VCC has reached the Power-On Threshold voltage - V_{POT}, regardless of the VCC rise time (see Figure 22 and Figure 23). The total reset period is the Power-On Reset period - t_{POR} + the Delay Time-out period - t_{TOUT}.

As the $\overline{\text{RESET}}$ pin is pulled high by an on-chip resistor, the pin can be left unconnected if no external reset is required. Connecting $\overline{\text{RESET}}$ to VCC will have the same effect. By holding the $\overline{\text{RESET}}$ pin low for a period after VCC has been applied, the Power-On Reset period can be extended. Refer to Figure 24 for a timing example on this.

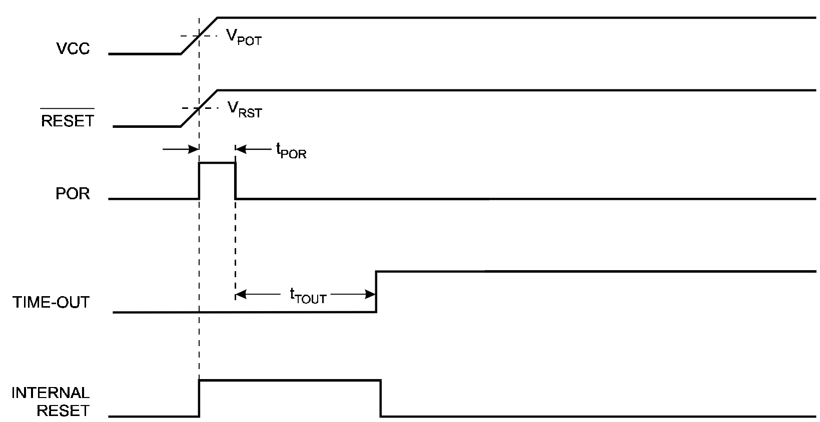


Figure 22: MCU Start-Up, $\overline{\text{RESET}}$ Tied to VCC or Unconnected. Rapidly Rising VCC

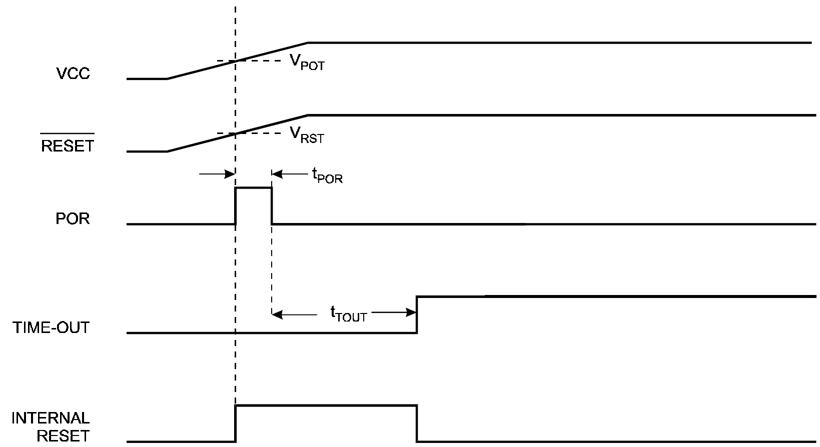


Figure 23: MCU Start-Up, $\overline{\text{RESET}}$ Tied to VCC or Unconnected. Slowly Rising VCC

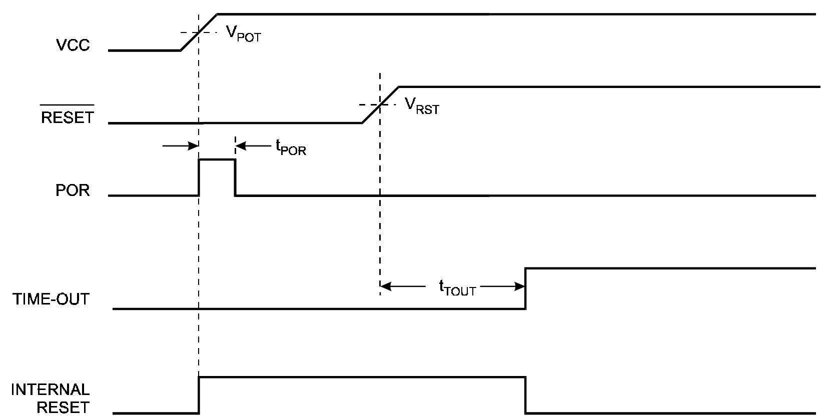


Figure 24: MCU Start-Up, $\overline{\text{RESET}}$ Controlled Externally

EXTERNAL RESET

An external reset is generated by a low level on the $\overline{\text{RESET}}$ pin. The pin must be held low for at least two crystal clock cycles. When $\overline{\text{RESET}}$ reaches the Reset Threshold Voltage - V_{RST} on its positive edge, the delay timer starts the MCU after the Time-out period t_{TOUT} has expired.

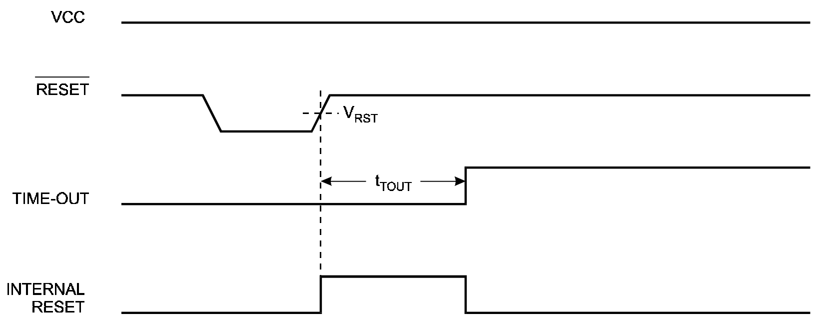


Figure 25: External Reset During Operation

WATCHDOG RESET

When the Watchdog times out, it will generate a short reset pulse of 1 XTAL cycle duration. On the falling edge of this pulse, the delay timer starts counting the Time-out period t_{TOUT} . Refer to Page 3-36 for details on operation of the Watchdog.

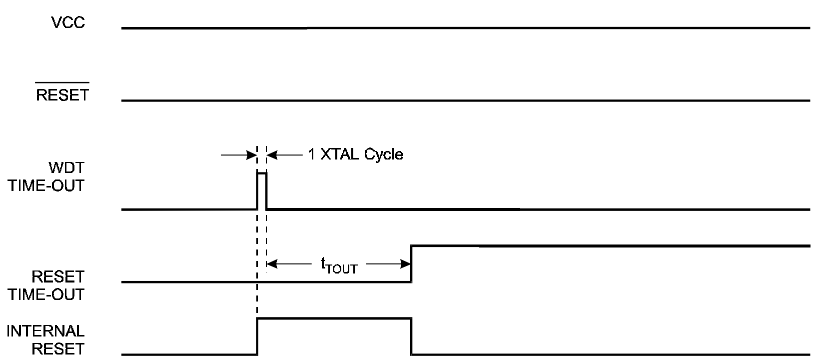


Figure 26: Watchdog Reset During Operation

INTERRUPT HANDLING

The AT90S2312 has two 8-bit Interrupt Mask control registers; GIMSK - General Interrupt Mask register and TIMSK - Timer/Counter Interrupt Mask register.

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

For Interrupts triggered by events that can remain static (E.g. the Output Compare register0 matching the value of Timer/Counter0) the interrupt flag is set when the event occurs. If the interrupt flag is cleared and the interrupt condition persists, the flag will not be set until the event occurs the next time.

When the Program Counter is vectored to the actual interrupt vector in order to execute the interrupt handling routine, hardware clears the corresponding flag that generated the interrupt. Some of the interrupt flags can also be cleared by writing a logic one to the flag bit position(s) to be cleared.

THE GENERAL INTERRUPT MASK REGISTER - GIMSK

Bit	7	6	5	4	3	2	1	0	
$3B	**INT1**	**INT0**	-	-	-	-	-	-	**GIMSK**
Read/Write	R/W	R/W	R	R	R	R	R	R	
Initial value	0	0	0	0	0	0	0	0	

Bit 7 - INT1 : External Interrupt Request 1 Enable:

When the INT1 bit is set (one) and the I-bit in the Status Register (SREG) is set (one), the external pin interrupt is activated. The Interrupt Sense Control1 bits 1/0 (ISC11 and ISC10) in the MCU general Control Register (MCUCR) defines whether the external interrupt is activated on rising or falling edge of the INT1 pin or level sensed. If the INT1 pin for external interrupts shall be activated, the DDD3 bit in the Data Direction Register PORTD (DDRD) must be cleared (zero) to force an input pin. The corresponding interrupt of External Interrupt Request 1 is executed from program memory address $002. See also "External Interrupts".

Bit 6 - INT0 : External Interrupt Request 0 Enable:

When the INT0 bit is set (one) and the I-bit in the Status Register (SREG) is set (one), the external pin interrupt is activated. The Interrupt Sense Control0 bits 1/0 (ISC01 and ISC00) in the MCU general Control Register (MCUCR) defines whether the external interrupt is activated on rising or falling edge of the INT0 pin or level sensed. If the INT0 pin for external interrupts shall be activated, the DDD2 bit in the Data Direction Register PORTD (DDRD) must be cleared (zero) to force an input pin. The corresponding interrupt of External Interrupt Request 0 is executed from program memory address $001. See also "External Interrupts".

Bits 5..0 - Res : Reserved bits:

These bits are reserved bits in the AT90S2312 and always read as zero.

THE TIMER/COUNTER INTERRUPT MASK REGISTER - TIMSK

Bit	7	6	5	4	3	2	1	0	
$39	**TOIE1**	**OCIE1**	-	-	**TICIE1**	-	**TOIE0**	**OCIE0**	**TIMSK**
Read/Write	R/W	R/W	R	R	R/W	R	R/W	R/W	
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 $005) is executed if an overflow in Timer/Counter1 occurs. The Overflow Flag (Timer/Counter1) is set (one) in the Timer/Counter Interrupt Flag Register - TIFR. When Timer/Counter1 is in PWM mode, the Timer Overflow flag is set when the counter changes counting direction at $0000.

Bit 6 - OCIE1 :Timer/Counter1 Output Compare Match Interrupt Enable:

When the OCIE1 bit is set (one) and the I-bit in the Status Register is set (one), the Timer/Counter1 Compare Match interrupt is enabled. The corresponding interrupt (at vector $004) is executed if a Compare match in Timer/Counter1 occurs. The Compare Flag in Timer/Counter1 is set (one) in the Timer/Counter Interrupt Flag Register - TIFR.

Bit 5,4 - Res :Reserved bits:

These bits are reserved bits in the AT90S2312 and always read 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 11, PD6(ICP). The Input Capture Flag in Timer/Counter1 is set (one) in the Timer/Counter Interrupt Flag Register - TIFR.

Bit 2 - Res : Reserved bit:
This bit is a reserved bit in the AT90S2312 and always reads as 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. The Overflow Flag (Timer0) is set (one) in the Timer/Counter Interrupt Flag Register - TIFR.

Bit 0 - OCIE0 :Timer/Counter0 Output Compare Match Interrupt Enable:
When the OCIE0 bit is set (one) and the I-bit in the Status Register is set (one), the Timer/Counter0 Compare Match interrupt is enabled. The corresponding interrupt (at vector $006) is executed if a compare match in Timer/Counter0 occurs. The Compare Flag (Timer0) is set (one) in the Timer/Counter Interrupt Flag Register - TIFR.

THE TIMER/COUNTER INTERRUPT FLAG REGISTER - TIFR

Bit	7	6	5	4	3	2	1	0	
$38	TOV1	OCF1	-	-	ICF1	-	TOV0	OCF0	TIFR
Read/Write	R/W	R/W	R	R	R/W	R	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

Bit 7 - 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 logic one 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 PWM mode, this bit is set when Timer/Counter1 changes counting direction at $0000.

Bit 6 - OCF1 : Output Compare Flag 1:
The OCF1 bit is set (one) when compare match occurs between the Timer/Counter1 and the data in OCR1 - Output Compare Register 1. OCF1 is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, OCF1 is cleared by writing a logic one to the flag. When the I-bit in SREG, and OCIE1 (Timer/Counter1 Compare match Interrupt Enable), and the OCF1 is set (one), the Timer/Counter1 Compare match Interrupt is executed.

Bit s 5, 4 - Res : Reserved bits:
These bits are reserved bits in the AT90S2312 and always read zero.

Bit 3 - 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 logic one to the flag.

Bit 2 - Res : Reserved bit:
This bit is a reserved bit in the AT90S2312 and always reads zero.

Bit 1 - 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 logic one 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.

Bit 0 - OCF0 : Output Compare Flag 0:
The OCF0 bit is set (one) when a compare match occurs between the Timer/Counter0 and the compared data in OCR0 - Output Compare Register0. OCF0 is cleared by hardware when executing the corresponding interrupt handling vector.

Alternatively, OCF0 is cleared by writing a logic one to the flag.. When the SREG I-bit, and OCIE0 (Timer/Counter0 Compare match Interrupt Enable), and OCF0 are set (one), the Timer/Counter0 Compare match Interrupt is executed.

EXTERNAL INTERRUPTS

The external interrupts are triggered by the INT1 and INT0 pins. Since these pins are alternate function pins in the general I/O ports, the corresponding pins must be set as input pins in the data direction register - DDRX.

The external interrupts are set up as indicated in the specification for the general interrupt mask register - GIMSK.

INTERRUPT RESPONSE TIME

The interrupt response time for all the enabled *AVR* interrupts are 4 clock cycles. After the 4 clock cycles the program vector address for the actual interrupt handling routine is executed. During this 4 clock cycle period, the Program Counter (2 bytes) is pushed onto the Stack, and the Stack Pointer is decremented by 2.

A return from an interrupt handling routine takes 4 clock cycles. During these 4 clock cycles, the Program Counter (2 bytes) is popped back from the Stack, and the Stack Pointer is incremented by 2.

Note that the Status Register - SREG - is not handled by the *AVR* hardware, neither for interrupts nor for subroutines. For the routines requiring a storage of the SREG, this must be performed by user software.

THE MCU CONTROL REGISTER - MCUCR

The MCU Control Register contains control bits for general MCU functions.

Bit	7	6	5	4	3	2	1	0	
$35	**-**	**-**	**SE**	**SM**	**ISC11**	**ISC10**	**ISC01**	**ISC00**	**MCUCR**
Read/Write	R	R	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

Bits 7, 6 - Res : Reserved bits:
These bits are reserved bits in the AT90S2312 and always read as zero.

Bit 5 - SE : Sleep Enable:
The SE bit must be set (one) to make the MCU enter the sleep mode when the SLEEP instruction is executed. To avoid the MCU entering the sleep mode unless it is the programmers purpose, it is recommended to set the Sleep Enable SE bit just before the execution of the SLEEP instruction.

Bit 4 - SM : Sleep Mode:
This bit selects between the two available sleep modes. When SM is cleared (zero), Idle Mode is selected as Sleep Mode. When SM is set (one), Power Down mode is selected as sleep mode. For details, refer to the paragraph “Sleep Modes” below.

Bits 3, 2 - ISC11, ISC10 : Interrupt Sense Control 1 bit 1 and bit 0:
The External Interrupt 1 is activated by the external pin INT1 if the SREG I-flag and the corresponding interrupt mask in the GIMSK register is set. The level and edges on the external INT1 pin that activate the interrupt are defined in the following table:

Table 4: 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.

Note: When changing the ISC11/ISC10 bits, INT1 must be disabled by clearing its Interrupt Enable bit in the GIMSK Register. Otherwise an interrupt can occur when the bits are changed.

Bits 1, 0 - ISC01, ISC00 : Interrupt Sense Control 0 bit 1 and bit 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 the following table:

Table 5: 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.

Note: When changing the ISC10/ISC00 bits, INT0 must be disabled by clearing its Interrupt Enable bit in the GIMSK Register. Otherwise an interrupt can occur when the bits are changed.

Sleep Modes

To enter the sleep modes, the SE bit in MCUCR must be set (one) and a SLEEP instruction must be executed. If an enabled interrupt occurs while the MCU is in a sleep mode, the MCU awakes, executes the instruction following SLEEP, enters the interrupt routine and resumes execution from the address following the last executed instruction. To avoid entering the Interrupt Routine, let a CLI - Disable Global Interrupts follow the SLEEP instruction. If reset occurs while the MCU is in Sleep Mode, the MCU awakes and executes from the reset vector. The contents of the register file and the I/O memory are unaltered in both modes.

Note that if a *level* triggered interrupt is used for wake-up, the low level must be held for a time longer than the oscillator start-up time of 16 ms. Otherwise, the interrupt flag may return to zero before the MCU starts executing.

IDLE MODE

When the SM bit is cleared (zero), the SLEEP instruction forces the MCU into the Idle Mode stopping the CPU but allowing 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 Timer Overflow interrupt and watchdog reset. If wakeup 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 during Idle Mode.

POWER DOWN MODE

When the SM bit is set (one), the SLEEP instruction forces the MCU into the Power Down Mode. In this mode, the oscillator is stopped. The user can select whether the watchdog shall be enabled during power-down mode. If the watchdog is enabled, it will wake up the MCU when the Watchdog Time-out period expires. If the watchdog is disabled, only an external reset or an external interrupt can wake up the MCU.

Timer / Counters

The AT90S2312 provides two general purpose Timer/Counters - one 8-bit T/C and one 16-bit T/C. The Timer/Counters have separate prescaling selection from the same 10-bit prescaling timer. Both Timer/Counters can either be used as a timer with an internal clock timebase or as a counter with an external pin connection which triggers the counting.

The Timer/Counter Prescaler

Figure 27 shows the general Timer/Counter prescaler.

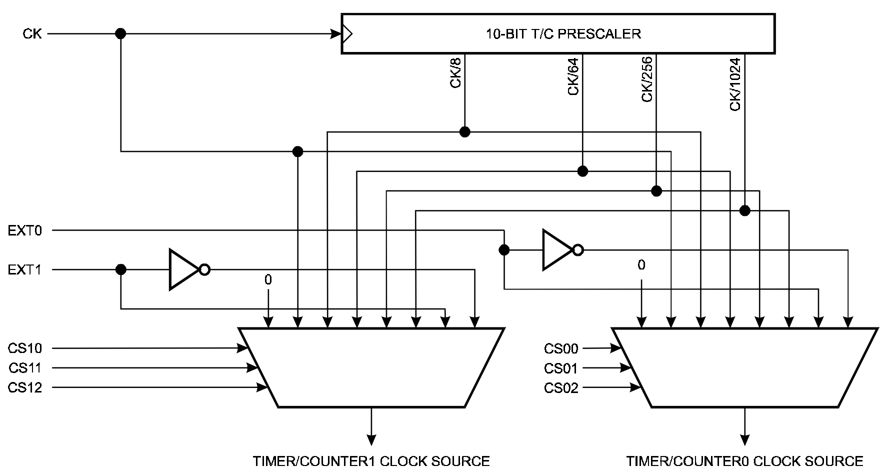


Figure 27: Timer/Counter Prescaler

The four different prescaled selections are: CK/8, CK/64, CK/256 and CK/1024 where CK is the oscillator clock. For the two Timer/Counters, added selections as CK, external source and stop, can be selected as clock sources.

The 8-Bit Timer/Counter0

Figure 28 shows the block diagram for Timer/Counter0.

The 8-bit Timer/Counter0 can select clock source from CK, prescaled CK, or an external pin. In addition it can be stopped as described in the specification for the Timer/Counter0 Control Register - TCCR0. The different status flags (overflow and compare match) and control signals are found in the Timer/Counter0 Control Register - TCCR0. The interrupt enable/disable settings for Timer/Counter0 are found in the Timer/Counter Interrupt Mask Register - TIMSK.

When Timer/Counter0 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 for the external clock being low and high 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 8-bit Timer/Counter0 features both a high resolution and a high accuracy usage with the lower prescaling opportunities. Similarly, the high prescaling opportunities make the Timer/Counter0 useful for lower speed functions or exact timing functions with infrequent actions.

The Timer/Counter0 supports an Output Compare function using the Output Compare Register 0 - OCR0 as the data source to be compared to the Timer/Counter0 contents. The Output Compare functions include optional clearing of the counter on compare matches, and actions on the Output Compare pin 0 on compare matches. The Output Compare pin 0 function makes the Timer/Counter0 useful for PWM (Pulse Width Modulation) functions.

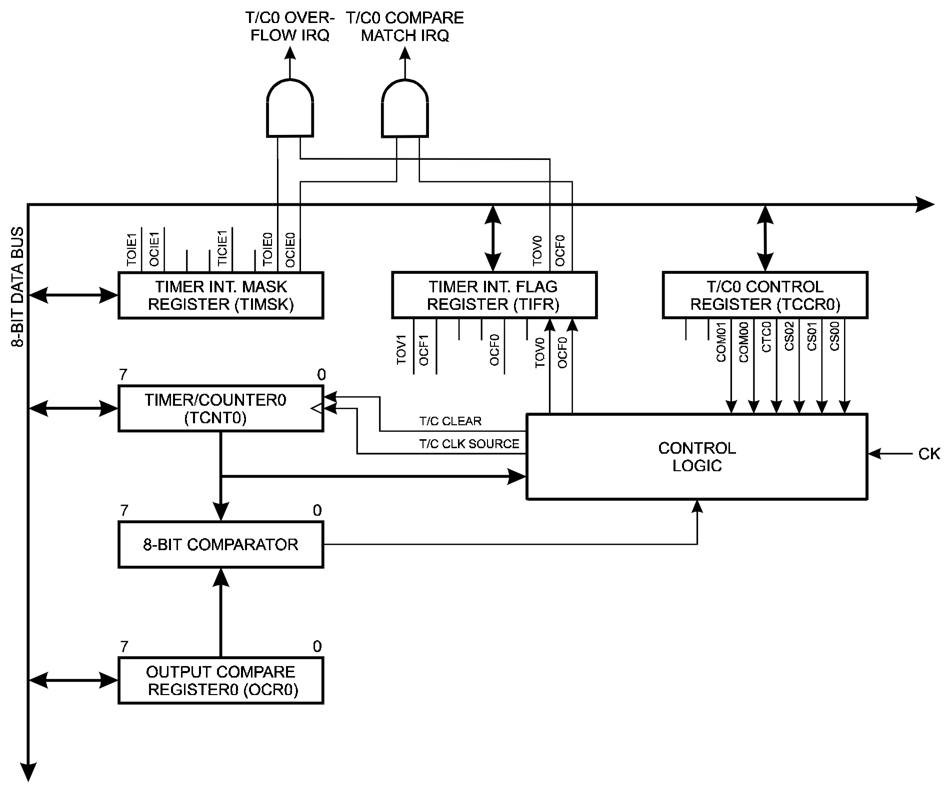


Figure 28: Timer/Counter 0 Block Diagram

THE TIMER/COUNTER0 CONTROL REGISTER - TCCR0

Bit	7	6	5	4	3	2	1	0	
$33	-	-	COM01	COM00	CTC0	CS02	CS01	CS00	TCCR0
Read/Write	R	R	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

Bits 7,6 - Res : Reserved bits:
These bits are reserved bits in the AT90S2312 and always read zero.

Bits 5,4 - COM01, COM00 : Compare Output Mode0, bit 1 and 0:
The COM01 and COM00 control bits determine any output pin action following a compare match in Timer/Counter0. Any output pin actions are effective on pin OC0 - Output Compare pin 0. Since this is an alternative function to an I/O port, the corresponding data direction control bit must be set (one) to control an output pin. The control configuration is defined in the following table:

Table 6: Compare 0 Mode Select

COM01	COM00	Description
0	0	Timer/Counter0 disconnected from output pin OC0.
0	1	Toggle the OC0 output line.
1	0	Clear the OC0 output line (to zero).
1	1	Set the OC0 output line (to one).

Note: When changing the COM01/COM00 bits, Output Compare Interrupt 0 must be disabled by clearing its Interrupt Enable bit in the TIMSK Register. Otherwise an interrupt can occur when the bits are changed.

Bit 3 - CTC0 : Clear Timer/Counter0 on Compare match:

When the CTC0 control bit is set (one), the Timer/Counter0 is reset to $00 in the clock cycle after the compare match. If the CTC0 control bit is cleared, the Timer/Counter0 continues running freely until it is stopped, set, cleared or wraps around (overflow).

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 Timer0.

Table 7: 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, rising edge
1	1	1	External Pin T0, falling 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 data direction control register (cleared to zero gives an input pin).

Bits 5..3 - Res : Reserved bits:

These bits are reserved bits in the AT90S2312 and always read zero.

THE TIMER COUNTER 0 - TCNT0

Bit	7	6	5	4	3	2	1	0	
$32	MSB							LSB	TCNT0
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 Timer/Counter0 is realized as an up-counter with read and write access. If the Timer/Counter0 is written and a clock source is present, the Timer/Counter0 continues counting in the clock cycle following the write operation.

THE OUTPUT COMPARE REGISTER 0 - OCR0

Bit	7	6	5	4	3	2	1	0	
$31	MSB							LSB	OCR0
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 Output Compare Register 0 is the source register for the Timer/Counter0 compare match functions.

The 16-Bit Timer/Counter1

Figure 29 shows the block diagram for Timer/Counter1.

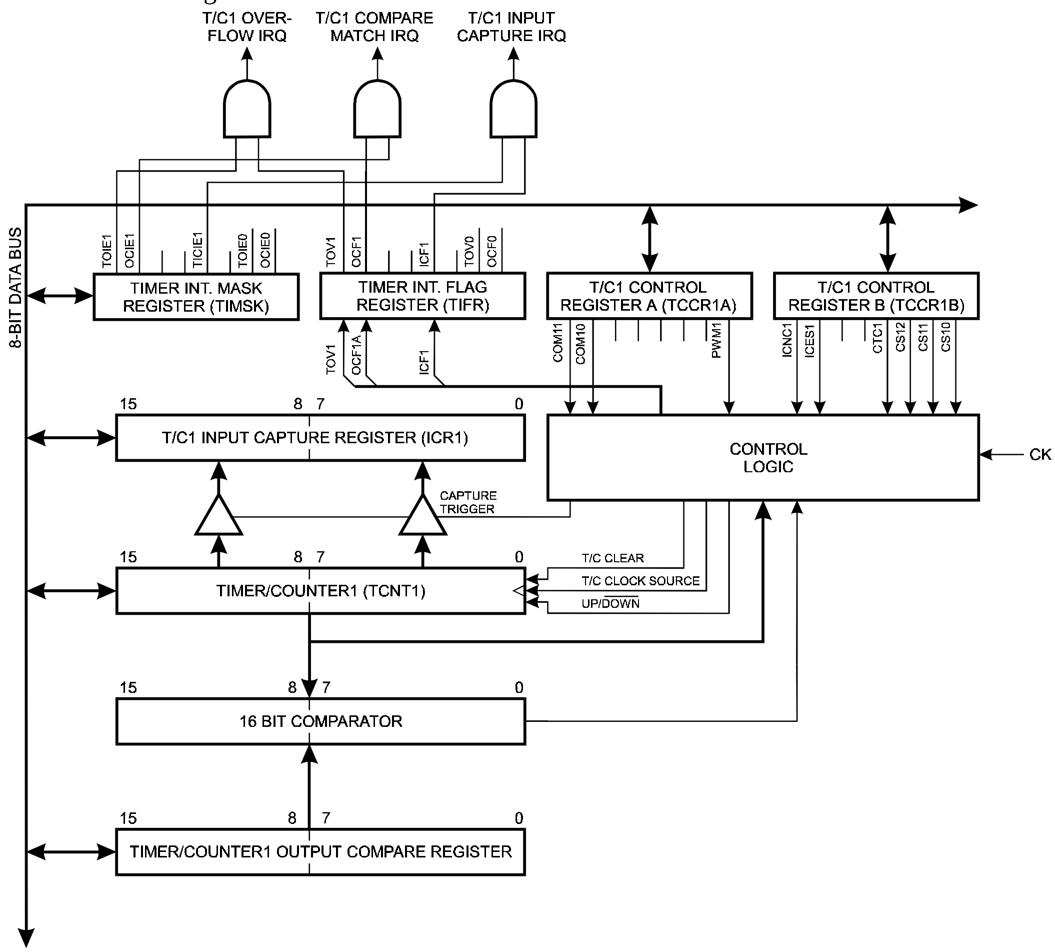


Figure 29: Timer/Counter1 Block Diagram

The 16-bit Timer/Counter1 can select clock source from CK, prescaled CK, or an external pin. In addition it can be stopped as described in the specification for the Timer/Counter1 Control Registers - TCCR1A and TCCR1B. The different status flags (overflow, compare match and capture event) and control signals are found in the Timer/Counter Interrupt Flag Register - TIFR. 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 for the external clock being low and high 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 makes the Timer/Counter1 useful for lower speed functions or exact timing functions with infrequent actions.

The Timer/Counter1 supports an Output Compare function using the Output Compare Register 1 - OCR1 as the data source to be compared to the Timer/Counter1 contents. The Output Compare functions include optional clearing of the counter on compare matches, and actions on the Output Compare pin 1 on compare matches.

Timer/Counter1 can also be used as a 10 bit Pulse With Modulator. In this mode the counter and the OCR1 register serve as a glitch-free stand-alone PWM with centered pulses. Refer to Page 3-35 for a detailed description on 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 - TCCR1.

The Timer/Counter1 input capture noise canceler block diagram is shown in Figure 30.

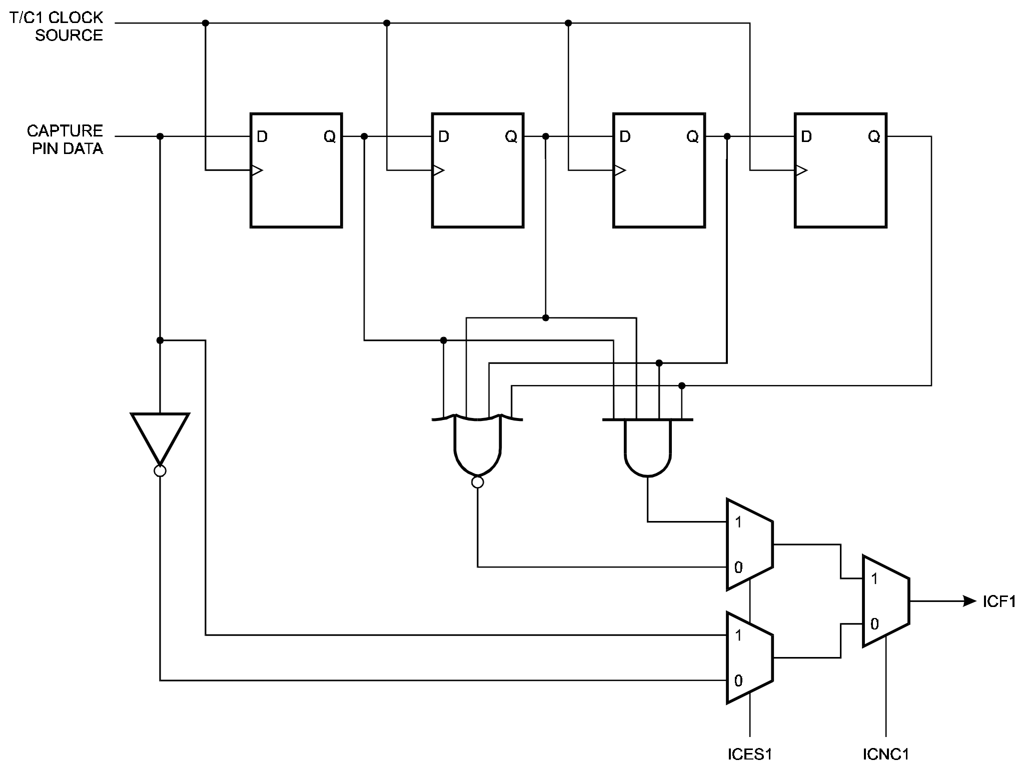


Figure 30: The Input Capture Noise Canceler

If the noise canceler function is enabled, the actual trigger condition for the capture event is monitored over 4 samples before the capture is activated. The sampling clock is the same clock as the clock source selected for the Timer/Counter1.

THE TIMER/COUNTER1 CONTROL REGISTER A - TCCR1A

Bit	7	6	5	4	3	2	1	0	
$2F	**COM11**	**COM10**	-	-	-	-	-	**PWM1**	**TCCR1A**
Read/Write	R/W	R/W	R	R	R	R	R	R/W	
Initial value	0	0	0	0	0	0	0	0	

Bits 7,6 - COM11, COM10 : Compare Output Mode1, bits 1 and 0:

The COM11 and COM10 control bits determine any output pin action following a compare match in Timer/Counter1. Any output pin actions affect pin OC1 - Output Compare pin 1. Since this is an alternative function to an I/O port, the corresponding direction control bit must be set (one) to control an output pin. The control configuration is shown 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).

Notes: X = A or B

In PWM mode, these bits have a different function. Refer to Table 10 for a detailed description.

When changing the COM1X1/COM1X0 bits, Output Compare Interrupts 1 must be disabled by clearing their Interrupt Enable bits in the TIMSK Register. Otherwise an interrupt can occur when the bits are changed.

Bits 5..1 - Res : Reserved bits:

These bits are reserved bits in the AT90S2312 and always read zero.

Bit 0 - PWM1 : Pulse Width Modulator enable:

This bit enables the PWM mode for Timer/Counter1. This mode is described on Page 4-35.

THE TIMER/COUNTER1 CONTROL REGISTER B - TCCR1B

Bit	7	6	5	4	3	2	1	0	
$2E	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 the same as the clock source frequency selected for the Timer/Counter1.

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 AT90S2312 and always read zero.

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 compare match. If the CTC1 control bit is cleared, the Timer/Counter1 continues counting until it is stopped, cleared, wraps around (overflow) or changes direction. In PWM mode, this bit has no effect.

Bits 2,1,0 - CS12, CS11, CS10 : Clock Select1, bit 2,1 and 0:
The Clock Select1 bits 2,1 and 0 define the prescaling source of Timer/Counter1.

Table 9: 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, rising edge
1	1	1	External Pin T1, falling 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).

THE TIMER/COUNTER1 - TCNT1H AND TCNT1L

Bit	15	14	13	12	11	10	9	8	
$2D	MSB								TCNT1H
$2C								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).

- **TCNT1 Timer/Counter1 Write:**
  When the CPU writes to the low byte TCNT1L, the written data is placed in the TEMP register. Next, when the CPU writes the high byte TCNT1H, this byte of data is combined with the byte data in the TEMP register, and all 16 bits are written in the TCNT1 Timer/Counter1 register simultaneously. Consequently, the low byte TCNT1L 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 clock cycle after it is preset with the written value.

TIMER/COUNTER1 OUTPUT COMPARE REGISTER - OCR1H AND OCR1L

Bit	15	14	13	12	11	10	9	8	
$2B	**MSB**								**OCR1H**
$2A								**LSB**	**OCR1L**
	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 register is a 16-bit read/write register.

The Timer/Counter1 Output Compare Register contains the data to be continuously compared with Timer/Counter1. Actions on compare matches are specified in the Timer/Counter1 Control and Status register.

Since the Output Compare Register - OCR1 - is a 16-bit register, a temporary register TEMP is used when OCR1 is written to ensure that both bytes are updated simultaneously. When the CPU writes the low byte, OCR1L, the data is temporarily stored in the TEMP register. When the CPU writes the high byte, OCR1H, the TEMP register is simultaneously written to OCR1L. Consequently, the low byte OCR1L must be written first for a full 16-bit register write operation.

THE TIMER/COUNTER1 INPUT CAPTURE REGISTER - ICR1H AND ICR1L

Bit	15	14	13	12	11	10	9	8	
$25	**MSB**								**ICR1H**
$24								**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.

TIMER/COUNTER1 IN PWM MODE

When the PWM mode is selected, Timer/Counter1 and the Output Compare Register1 - OCR1, form a 10-bit, free-running, glitch-free and phase correct PWM with output on the PD1(OC1) pin. Timer/Counter1 acts as an up/down counter, counting up from $0000 to $03FF (i.e. from 0 to 1023), when 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 OCR1, the PD1(OC1) pin is set or cleared according to the settings of the COM11 and COM10 bits in the Timer/Counter1 Control Register TCCR1. Refer to Table 10 for details.

Table 10: Compare1 Mode Select in PWM Mode

COM11	COM10	Effect on OC1
0	0	Not connected
0	1	Not connected
1	0	Cleared on compare match, upcounting. Set on compare match, downcounting (non-inverted PWM).
1	1	Cleared on compare match, downcounting. Set on compare match, upcounting (inverted PWM).

Note that in the PWM mode, the 10 least significant OCR1 bits, when written, are transferred to a temporary location. They are latched when Timer/Counter1 reaches the top - $03FF. This prevents the occurrence of odd-length PWM pulses (glitches) in the event of an unsynchronized OCR1 write. See Figure 31 for an example.

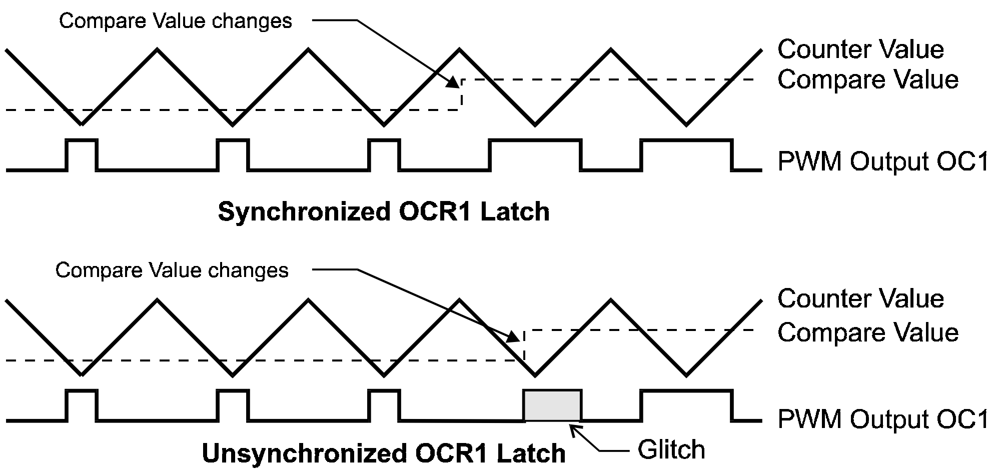


Figure 31: Effects on Unsynchronized OCR1 Latching

When OCR1 contains $0000 or $03FF, the output OC1 is held low or high according to the settings of COM11 and COM10. This is shown in Table 11.

Table 11: PWM Outputs OCR = $0000 or $03FF

COM11	COM10	OCR1	Output OC1
1	0	$0000	L
1	0	$03FF	H
1	1	$0000	H
1	1	$03FF	L

In PWM mode, the Timer Overflow Flag1, TOV1, is set when the counter changes direction at $0000. Timer Overflow Interrupt1 operates exactly as in normal Timer/Counter mode, i.e. it is executed when TOV1 is set provided that Timer Overflow Interrupt1 and global interrupts are enabled. This does also apply to the Timer Output Compare1 flag and interrupt.

The PWM output frequency is given by: $f_{PWM} = \frac{f_{TC1}}{2046}$, where f_{TC1} is the Timer/Counter1 Clock Source frequency.

The Watchdog Timer

The Watchdog Timer is clocked from a separate on-chip oscillator which runs at 1MHz. By controlling the Watchdog Timer prescaler, the Watchdog reset interval can be adjusted from 16 to 2048 ms. The WDR - Watchdog Reset - instruction resets the Watchdog Timer. From the Watchdog is reset, eight different clock cycle periods can be selected to determine the reset period. If the reset period expires without another Watchdog reset, the AT90S2312 resets and executes from the reset vector. For timing details on the Watchdog reset, refer to Page 2-23.

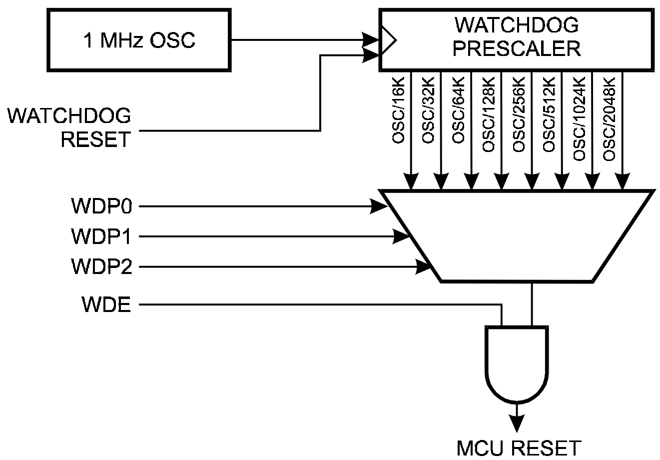


Figure 32: Watchdog Timer

THE WATCHDOG TIMER CONTROL REGISTER - WDTCR

Bit	7	6	5	4	3	2	1	0	
$21	-	-	-	-	WDE	WDP2	WDP1	WDP0	WDTCR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

Bits 7..4 - Res : Reserved bits:

These bits are reserved bits in the AT90S2312 and will always read as zero.

Bit 3 - WDE : Watch Dog Enable:

When the WDE is set (one) the Watchdog Timer is enabled, and if the WDE is cleared (zero) the Watchdog Timer function is disabled.

Bits 2..0 - WDP2, WDP1, WDP0 : Watchdog Timer Prescaler 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 timeout periods are shown in Table 12.

Table 12: Watch Dog Timer Prescale Select

WDP2	WDP1	WDP0	Timeout Period
0	0	0	16 ms
0	0	1	32 ms
0	1	0	64 ms
0	1	1	128 ms
1	0	0	256 ms
1	0	1	512 ms
1	1	0	1024 ms
1	1	1	2048 ms

EEPROM Read/Write Access

The EEPROM access registers are accessible in the I/O space using the IN and OUT instructions.

The write access time is in the range of 2.5 - 4ms, depending on the Vcc voltages. A self-timing function, however, lets the user software detect when the next byte can be written.

The read access time is the same as for the Flash memory and is of no concern to the user software.

THE EEPROM ADDRESS REGISTER - EEAR

Bit	7	6	5	4	3	2	1	0	
$1E	-	MSB						LSB	EEAR
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 AT90S2312 and will always read as zero.

Bit 6..0 - EEAR6..0 : EEPROM Address:

The EEPROM Address Register - EEAR6..0 - specifies the EEPROM address in the 128 bytes EEPROM space. The EEPROM data bytes are addressed linearly between 0 and 127.

THE EEPROM DATA REGISTER - EEDR

Bit	7	6	5	4	3	2	1	0	
$1D	MSB							LSB	EEDR
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	

Bit 7..0 - EEDR7..0 : EEPROM Data:
For the EEPROM write operation, the EEDR register contains the data to be written to the EEPROM in the address given by the EEAR register. For the EEPROM read operation, the EEDR contains the data read out from the EEPROM at the address given by EEAR.

THE EEPROM CONTROL REGISTER - EECR

Bit	7	6	5	4	3	2	1	0	
$1C	-	-	-	-	-	-	EEWE	EERE	EECR
Read/Write	R	R	R	R	R	R	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

Bit 7..2 - Res : Reserved bits:
These bits are reserved bits in the AT90S2312 and will always read as zero.

Bit 1 - EEWE : EEPROM Write Enable:
The EEPROM Write Enable Signal EEWE is the write strobe to the EEPROM. When address and data are correctly set up, the EEWE bit must be set to write the value into the EEPROM. When the write access time (typically 2.5ms at Vcc=5V or 4ms at Vcc=2.7V) has elapsed, the EEWE bit is cleared (zero) by hardware. The user software can poll this bit and wait for a zero before writing the next byte.

Bit 0 - EERE : EEPROM Read Enable:
The EEPROM Read Enable Signal EERE is the read strobe to the EEPROM. When the correct address is set up in the EEAR register, the EERE bit must be set. When the EERE bit is cleared (zero) by hardware, requested data is found in the EEDR register. The EEPROM read access time is within a single clock cycle and there is no need to poll the EERE bit.

The UART

The AT90S2312 features a full duplex Universal Asynchronous Receiver and Transmitter (UART). The main features are:

- Baud rate generator generates any baud rate
- High baud rates at low XTAL frequencies
- 8 or 9 bits data
- Noise filtering
- Overrun detection
- Framing Error detection
- False Start Bit detection
- Three separate interrupts on TX Complete, TX Data Register Empty and RX Complete

Data Transmission

A block schematic of the UART transmitter is shown in Figure 33.

Data transmission is initiated by writing the data to be transmitted to the UART I/O Data Register, UDR. Data is transferred from UDR to the Transmit shift register when:

- A new character has been written to UDR after the stop bit from the previous character has been shifted out. The shift register is loaded immediately.
- A new character has been written to UDR before the stop bit from the previous character has been shifted out. The shift register is loaded when the stop bit of the character currently being transmitted has been shifted out.

At this time the UDRE (UART Data Register Empty) bit in the UART Status Register, USR, is set. When this bit is set (one), the UART is ready to receive the next character. At the same time as the data is transferred from UDR to the 10(11)-bit shift register, bit 0 of the shift register is cleared (start bit) and bit 9 or 10 is set (stop bit). If 9 bit data word is selected (the CHR9 bit in the UART Control Register, UCR is set), the TXB8 bit in UCR is transferred to bit 9 in the Transmit shift register.

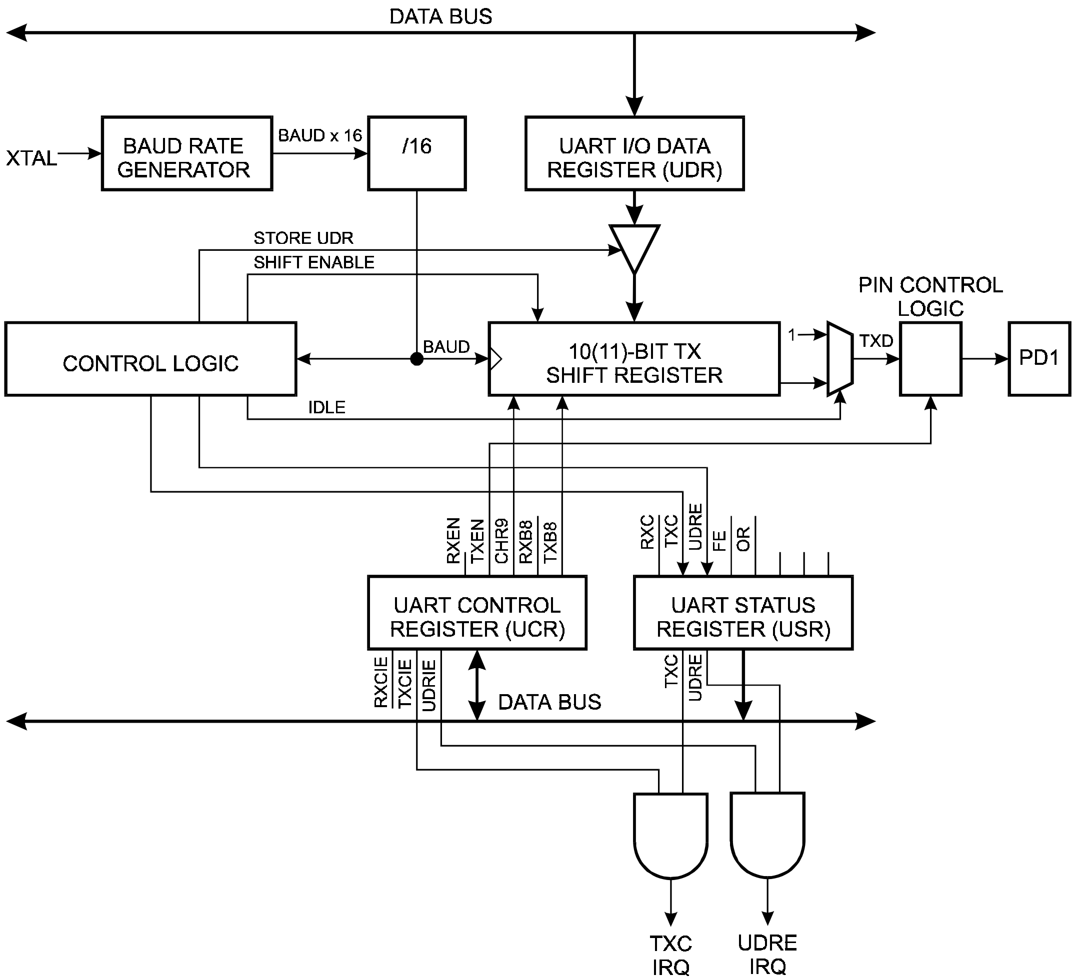


Figure 33: UART Transmitter

On the Baud Rate clock following the transfer operation to the shift register, the start bit is shifted out on the TXD pin. Then follows the data, LSB first. When the stop bit has been shifted out, the shift register is loaded if any new data has been written to the UDR during the transmission. During loading, UDRE is set. If there is no new data in the UDR register to send when the stop bit is shifted out, the UDRE flag will remain set. In this case, after the stop bit has been present on TXD for one bit length, the TX Complete Flag, TXC, in USR is set.

The TXEN bit in UCR enables the UART transmitter when set (one). By clearing this bit (zero), the PD1 pin can be used for general I/O. When TXEN is set, the UART Transmitter will be connected to the PD1 pin regardless of the setting of the DDD1 bit in DDRD.

Data Reception

Figure 34 shows a block diagram of the UART Receiver

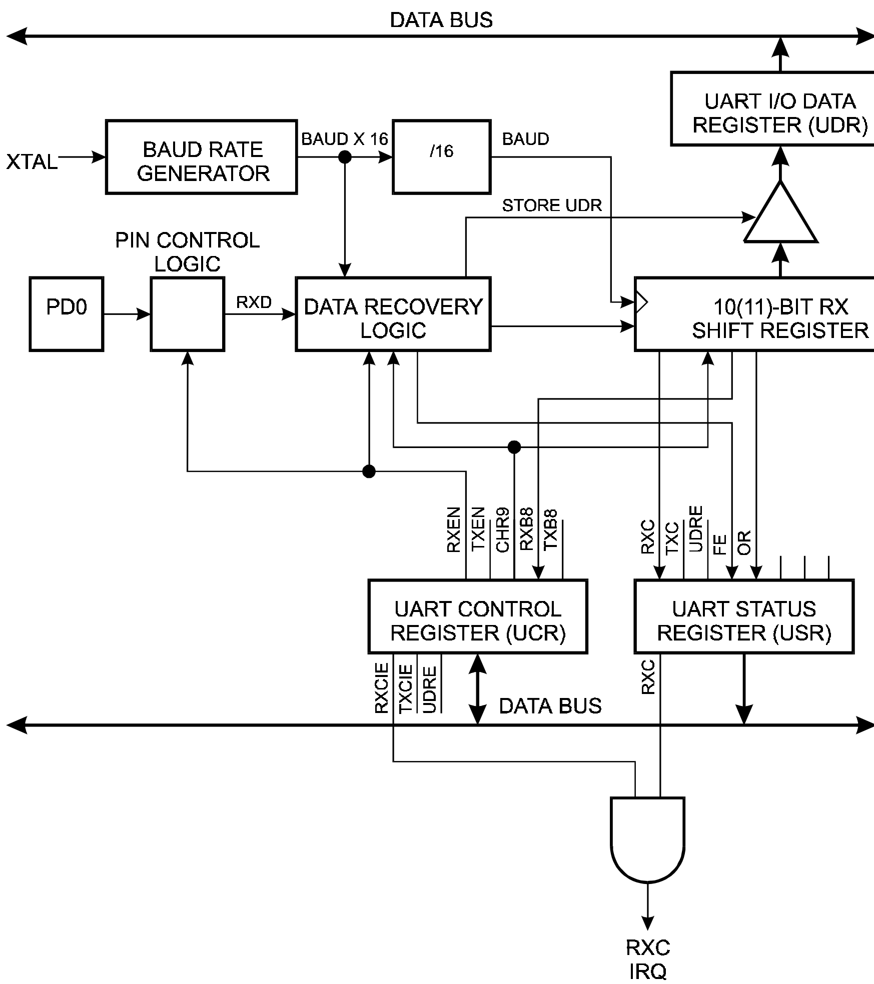


Figure 34: UART Receiver

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 sample 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 35.

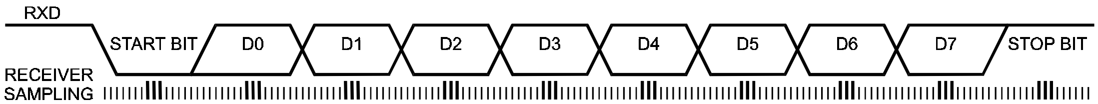


Figure 35: Sampling Received Data

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 accessed since the last receive, the OverRun (OR) flag in UCR is set. This means that the new data transferred to the shift register has overwritten the old data not yet read, and the old data is lost. The user should always check the OR bit before reading from the UDR register in order to detect any overruns.

By clearing the RXEN bit in the UCR register, 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 the PD0 pin regardless of the setting of the DDD0 bit in DDRD.

UART Control

THE UART I/O DATA REGISTER - UDR

Bit	7	6	5	4	3	2	1	0	
$0C	**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.

THE UART STATUS REGISTER - USR

Bit	7	6	5	4	3	2	1	0	
$0B	**RXC**	**TXC**	**UDRE**	**FE**	**OR**	**-**	**-**	**-**	**USR**
Read/Write	R	R	R	R	R	R	R	R	
Initial value	0	1	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, setting of RXC causes the UART Receive Complete interrupt to be executed. RXC is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, the bit is cleared (zero) by first reading USR while RXC is set (one) and then reading UDR.
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 the 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 first reading USR while TXC is set and then writing UDR.

This bit is set (one) during reset to indicate that the transmitter is not busy transmitting anything.

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, setting of UDRE causes the UART Transmit Complete interrupt to be executed. UDRE is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, the UDRE bit is cleared (zero) by first reading USR while UDRE is set and then writing UDR.

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.

The FE bit is cleared (zero) by first reading USR while FE is set and then reading UDR.

Bit 3 - OR : OverRun:
This bit is set if an Overrun condition is detected, i.e. when a character already present in the UDR register is not read before the next character is transferred from the Receiver Shift register.

The OR bit is cleared (zero) by first reading USR while OR is set and then reading UDR.

Bits 2..0 - Res : Reserved bits:
These bits are reserved bits in the AT90S2312 and will always read as zero.

THE UART CONTROL REGISTER - UCR

Bit	7	6	5	4	3	2	1	0	
$0A	RXCIE	TXCIE	UDRIE	RXEN	TXEN	CHR9	RXB8	TXB8	UCR
Read/Write	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 - RXCIE : RX Complete Interrupt Enable:
When this bit is set (one), a setting of the RXC bit in USR will cause the Receive Complete interrupt routine to be executed provided that global interrupts are enabled.

Bit 6 - TXCIE : TX Complete Interrupt Enable:
When this bit is set (one), a setting of the TXC bit in USR will cause the Transmit Complete interrupt routine to be executed provided that global interrupts are enabled.

Bit 5 - UDRIE : UART Data Register Empty Interrupt Enable:

When this bit is set (one), a setting of the UDRE bit in USR will cause the UART Data Register Empty interrupt routine to be executed provided that global interrupts are enabled.

Bit 4 - RXEN : Receiver Enable:
This bit enables the UART receiver when set (one). When the receiver is disabled, the TXC, OR and FE status flags cannot become set. If these flags are set, turning off RXEN does not cause them to be cleared.

Bit 3 - TXEN : Transmitter Enable:
This bit enables the UART transmitter when set (one). When disabling the transmitter while transmitting a character, the transmitter is not disabled before the character in the shift register plus any following character in UDR has been completely transmitted.

Bit 2 - CHR9 : 9 Bit Characters: When this bit is set (one) transmitted and received characters are 9 bit long plus start and stop bits. The 9^{th} bit is read and written by using the RXB8 and TXB8 bits in UCR, respectively. The 9^{th} data bit can be used as an extra stop bit or a parity bit.

Bit 1 - RXB8 : Receive Data Bit 8
When CHR9 is set (one), RXB8 is the 9^{th} data bit of the received character.

Bit 0 - TXB8 : Transmit Data Bit 8
When CHR9 is set (one), TXB8 is the 9^{th} data bit in the character to be transmitted.

THE BAUD RATE GENERATOR

The baud rate generator is a frequency divider which generates baud-rates according to the following equation:

$$BAUD = \frac{f_{ck}}{16(UBRR + 1)}$$

- BAUD = Baud-Rate
- f_{ck} = Crystal Clock frequency
- UBRR = Contents of the UART Baud Rate register, UBRR (0-255)

For standard crystal frequencies, the most commonly used baud rates can be generated by using the UBRR settings in Table 13. UBRR values which yield an actual baud rate differing less than 2% from the target baud rate, are bolded in the table.

Table 13: 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
57600	UBRR=	0	7.8	UBRR=	1	0.0	UBRR=	1	7.8	UBRR=	2	12.5
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
57600	UBRR=	3	12.5	UBRR=	3	0.0	UBRR=	3	7.8	UBRR=	4	0.0
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
57600	UBRR=	7	0.0	UBRR=	8	3.7	UBRR=	9	0.0	UBRR=	11	0.0
115200	UBRR=	3	0.0	UBRR=	3	7.8	UBRR=	4	0.0	UBRR=	5	0.0

Baud Rate	14.746 MHz		%Error	16 MHz		%Error	18.432 MHz		%Error	20 MHz		%Error
2400	UBRR=	383	-	UBRR=	416	-	UBRR=	479	-	UBRR=	520	-
4800	UBRR=	191	0.0	UBRR=	207	0.2	UBRR=	239	0.0	UBRR=	259	-
9600	UBRR=	95	0.0	UBRR=	103	0.2	UBRR=	119	0.0	UBRR=	129	0.2
14400	UBRR=	63	0.0	UBRR=	68	0.6	UBRR=	79	0.0	UBRR=	86	0.2
19200	UBRR=	47	0.0	UBRR=	51	0.2	UBRR=	59	0.0	UBRR=	64	0.2
28800	UBRR=	31	0.0	UBRR=	34	0.8	UBRR=	39	0.0	UBRR=	42	0.9
57600	UBRR=	15	0.0	UBRR=	16	2.1	UBRR=	19	0.0	UBRR=	21	1.4
115200	UBRR=	7	0.0	UBRR=	8	3.7	UBRR=	9	0.0	UBRR=	10	1.4

THE UART BAUD RATE REGISTER - UBRR

Bit	7	6	5	4	3	2	1	0	
$09	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 which specifies the UART Baud Rate according to the description on Page 3-42.

The Analog Comparator

The analog comparator compares the input values on the positive pin AIN0 (PB0) and the negative pin PB1(AIN1). When the voltage on the positive pin PB0 (AIN0) is higher than the voltage on the negative PB1 (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 36.

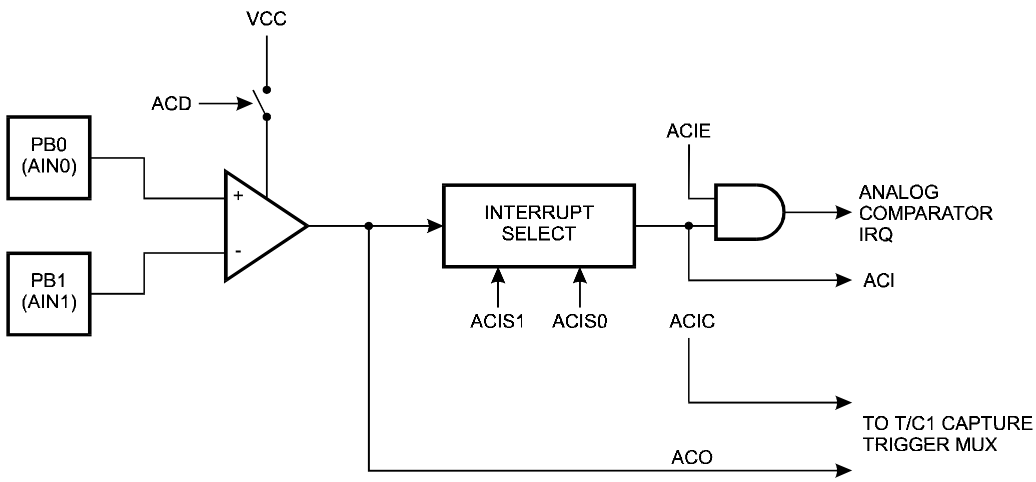


Figure 36: Analog Comparator Block Diagram

THE ANALOG COMPARATOR CONTROL AND STATUS REGISTER - ACSR

Bit	7	6	5	4	3	2	1	0	
$08	**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	0	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. It is most commonly used if power consumption during Idle Mode is critical, and wake-up from the analog comparator is not required. 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 AT90S2312 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 ACIS1 and ACIS0. 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 logic one to the flag.

Bit 3 - ACIE : Analog Comparator Interrupt Enable:

When the ACIE bit is set (one) and the I-bit in the Status Register is set (one), the analog comparator interrupt is activated. When cleared (zero), the interrupt is disabled. For details on the comparator, refer to Page 4-46.

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 14.

Table 14: 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.

I/O-Ports

Port B

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

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

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

The Port B pins with alternate functions are shown in the following table:

Table 15: Port B Pins Alternate Functions

Port Pin	Alternate Functions
PB0	AIN0 (Analog comparator positive input)
PB1	AIN1 (Analog comparator negative input)
PB2	OC0 (Timer/Counter0 Output compare match output)
PB3	OC1 (Timer/Counter1 Output compare match output)
PB5	MOSI (Data input line for memory downloading)
PB6	MISO (Data output line for memory uploading)
PB7	SCK (Master clock input)

When the pins are used for the alternate function the DDRB and PORTB register has to be set according to the alternate function description.

THE PORT B DATA REGISTER - PORTB

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

THE PORT B DATA DIRECTION REGISTER - DDRB

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

THE PORT B INPUT PINS ADDRESS - PINB

Bit	7	6	5	4	3	2	1	0	
$16	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	PINB
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 B Input Pins address - PINB - is not a register, and this address enables access to the physical value on each Port B pin. When reading PORTB, the PORTB Data Latch is read, and when reading PINB, the logical values present on the pins are read.

PORTB AS GENERAL DIGITAL I/O

All 8 bits in port B are equal 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 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.

Table 16: DDBn Effects on Port B Pins

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

n: 7,6…0, pin number.

ALTERNATE FUNCTIONS FOR PORTB

The alternate pin functions of Port B are:

SCK - PORTB, Bit 7:

SCK , Clock input pin for Memory downloading.

MISO - PORTB, Bit 6:

MISO, Data output pin for Memory downloading.

MOSI - PORTB, Bit 5:

MOSI , Data input pin for Memory downloading.

OC1 - PORTB, Bit 3:

OC1, Output compare match output: The PB3 pin can serve as an external output when timer 1 compare match. The PB3 pin has to be configured as an output (DDB3 is set (one)) to serve this function. See the timer description for further details, and how to enable the output.

OC0 - PORTB, Bit 2:

OC0, Output compare match output: The PB2 pin can serve as an external output when timer 0 compare match. The PB2 pin has to be configured as an output (DDB2 is set (one)) to serve this function. See the timer description for further details, and how to enable the output.

AIN1 - PORTB, Bit 1:

AIN1, Analog Comparator Negative Input. When configured as an input (DDB1 is cleared (zero)) and with the internal MOS pull up resistor switched off (PB1 is cleared (zero)), this pin also serves as the negative input of the on-chip analog comparator.

AIN0 - PORTB, Bit 0:

AIN0, Analog Comparator Positive Input. When configured as an input (DDB0 is cleared (zero)) and with the internal MOS pull up resistor switched off (PB0 is cleared (zero)), this pin also serves as the positive input of the on-chip analog comparator.

PORTB SCHEMATICS

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

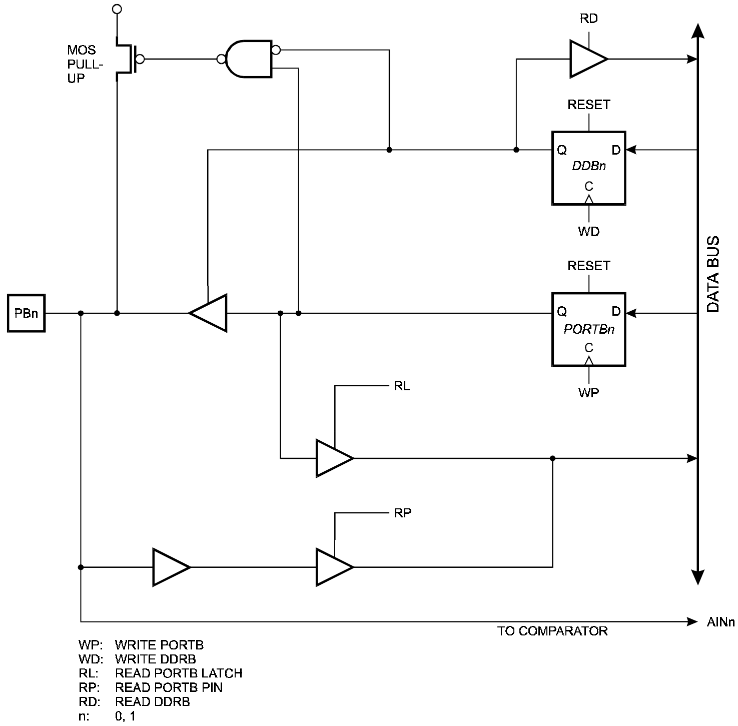


Figure 37: PORTB Schematic Diagram (pins PB0 and PB1)

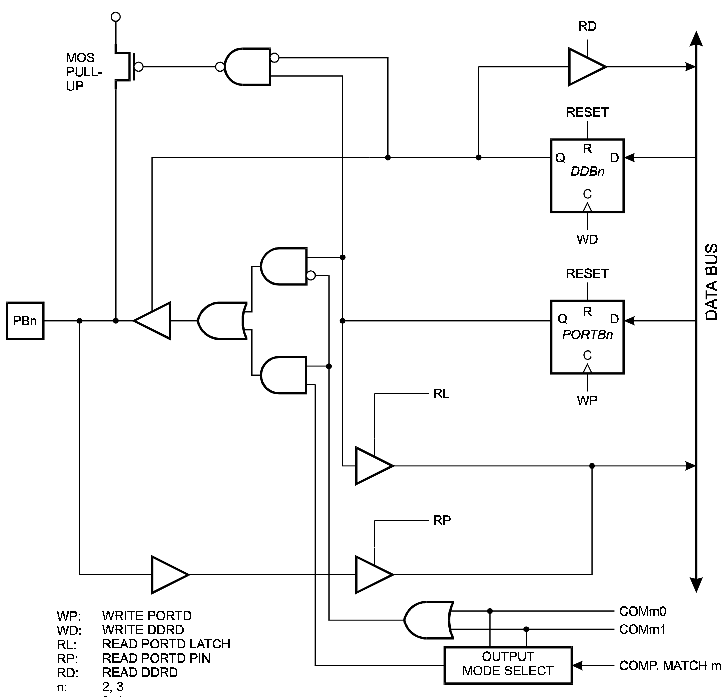


Figure 38: PORTB Schematic Diagram (Pins PB2 and PB3)

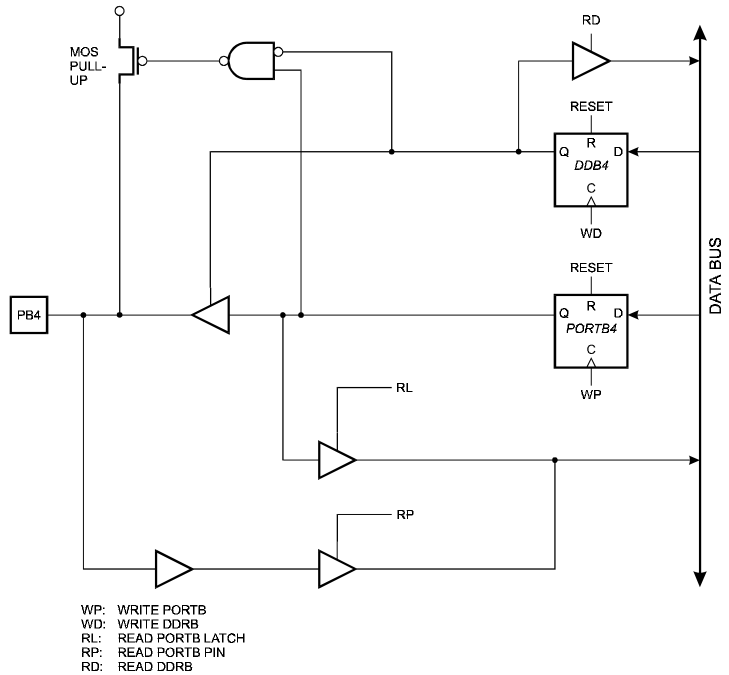


Figure 39: PORTB Schematic Diagram, Pin PB4

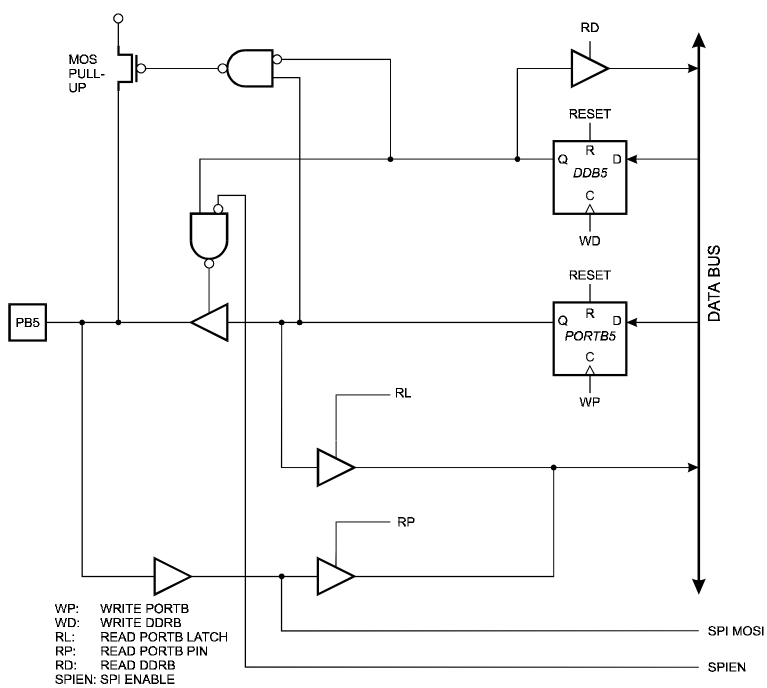


Figure 40: PORTB Schematic Diagram Pin PB5

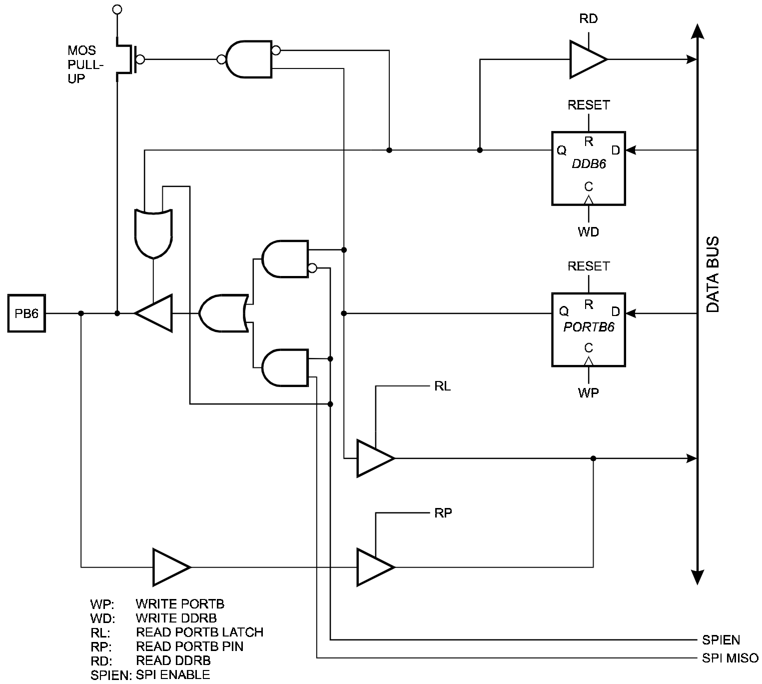


Figure 41: PORTB Schematic Diagram, Pin PB6

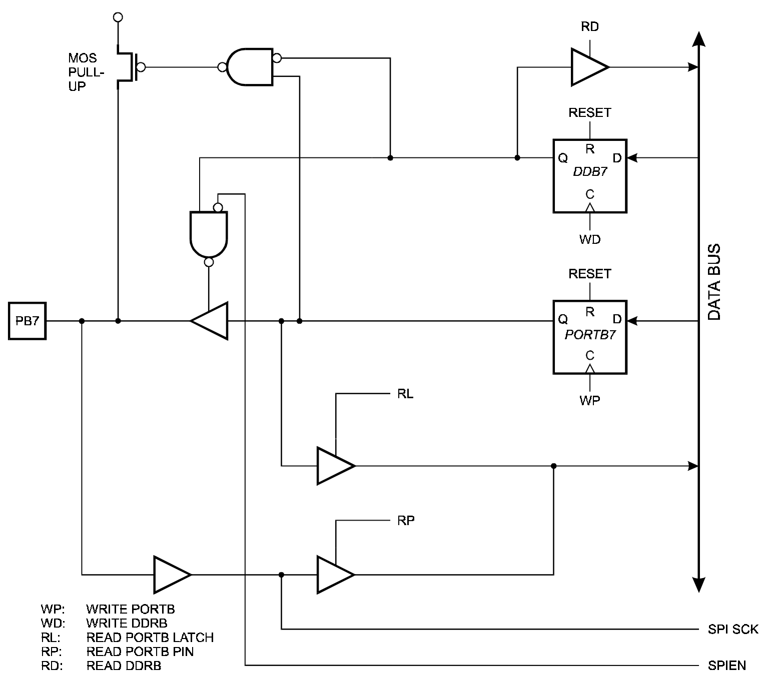


Figure 42: PORTB Schematic Diagram, Pin PB7

Port D

Three data memory address locations are allocated for the Port D, one each for the Data Register - PORTD ($12), Data Direction Register - DDRD ($11) and the Port D Input Pins - PIND ($10). The Port D Input Pins address is read only, while the Data Register and the Data Direction Register are read/write.

Port D has seven bi-directional I/O pins with internal pullups, PD6..PD0. The Port D output buffers can sink 20 mA. As inputs, Port D pins that are externally pulled low will source current (IIL) if the pullups are activated.

Some Port D pins have alternate functions as shown in the following table:

Table 17: Port D Pins Alternate Functions

Port Pin	Alternate Function
PD0	RXD (Receive data input for the UART)
PD1	TXD (Transmit data output for the UART)
PD2	INT0 (External interrupt 0 input)
PD3	INT1 (External interrupt 1 input)
PD4	TO (Timer/Counter0 external input)
PD5	T1 (Timer/Counter1 external input)
PD6	ICP (Timer/Counter1Input Capture pin)

When the pins are used for the alternate function the DDRD and PORTD register has to be set according to the alternate function description.

THE PORT D DATA REGISTER - PORTD

Bit	7	6	5	4	3	2	1	0	
$12	-	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	PORTD
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	

THE PORT D DATA DIRECTION REGISTER - DDRB

Bit	7	6	5	4	3	2	1	0	
$11	-	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	DDRD
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	

THE PORT D INPUT PINS ADDRESS

Bit	7	6	5	4	3	2	1	0	
$10	-	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	PIND
Read/Write	R	R	R	R	R	R	R	R	
Initial value	0	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	

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

PORTD AS GENERAL DIGITAL I/O

PDn, General I/O pin: The DDDn bit in the DDRD register selects the direction of this pin. If DDDn is set (one), PDn is configured as an output pin. If DDDn is cleared (zero), PDn is configured as an input pin. If PORTDn is set (one) when configured as an input pin the MOS pull up resistor is activated. To switch the pull up resistor off the PORTDn has to be cleared (zero) or the pin has to be configured as an output pin.

Table 18: DDDn Bits on Port D Pins

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

n: 6…0, pin number.

ALTERNATE FUNCTIONS FOR PORTD

The alternate functions of Port D are:

ICP - PORTD, Bit 6:

Timer/Counter1 Input Capture pin. The PD6 pin has to be configured as an input (DDD6 cleared (zero)) to serve this function. See the Timer/Counter1 description for further details. The internal pull up MOS resistor can be activated as described above.

T1 - PORTD, Bit 5:

T1, :Timer 1 clock source. The PD5 pin has to be configured as an input (DDD5 is cleared (zero)) to serve this function. See the timer description for further details. The internal pull up MOS resistor can be activated as described above.

T0 - PORTD, Bit 4:

T0, Timer/Counter0 clock source: The PD4 pin has to be configured as an input (DDD4 is cleared (zero)) to serve this function. See the Timer description for further details. The internal pull up MOS resistor can be activated as described above.

INT1 - PORTD, Bit 3:

INT1, External Interrupt source 1: The PD3 pin can serve as an external active low interrupt source to the MCU. The PD3 pin has to be configured as an input (DDD3 is cleared (zero)) to serve this function. The internal pull up MOS resistor can be activated as described above. See the interrupt description for further details, and how to enable the source.

INT0 - PORTD, Bit 2:

INT0, External Interrupt source 0: The PD2 pin can serve as an external active low interrupt source to the MCU. The PD2 pin has to be configured as an input (DDD2 is cleared (zero)) to serve this function. The internal pull up MOS resistor can be activated as described above. See the interrupt description for further details, and how to enable the source.

TXD - PORTD, Bit 1:

Transmit Data (Data output pin for the UART)

RXD - PORTD, Bit 0:

Receive Data (Data input pin for the UART)

PORTD SCHEMATICS

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

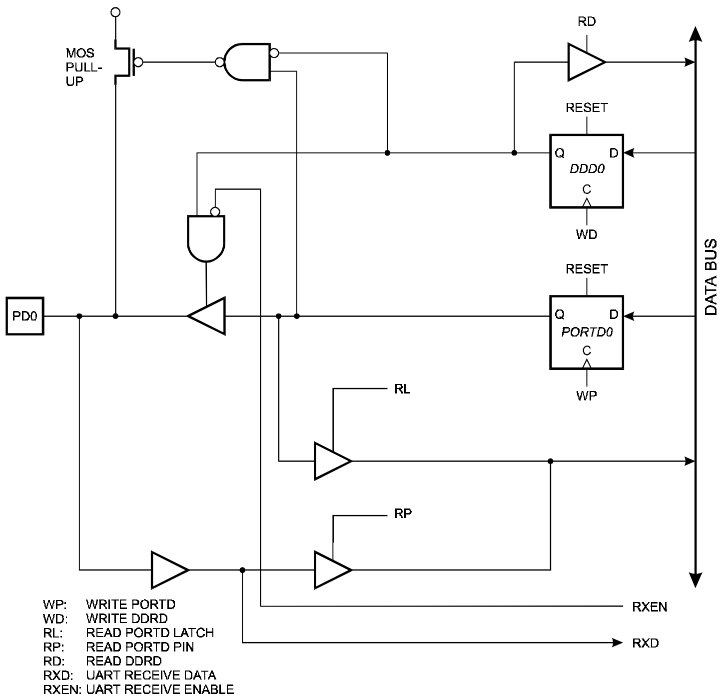


Figure 43: PORTD Schematic Diagram (Pin PD0)

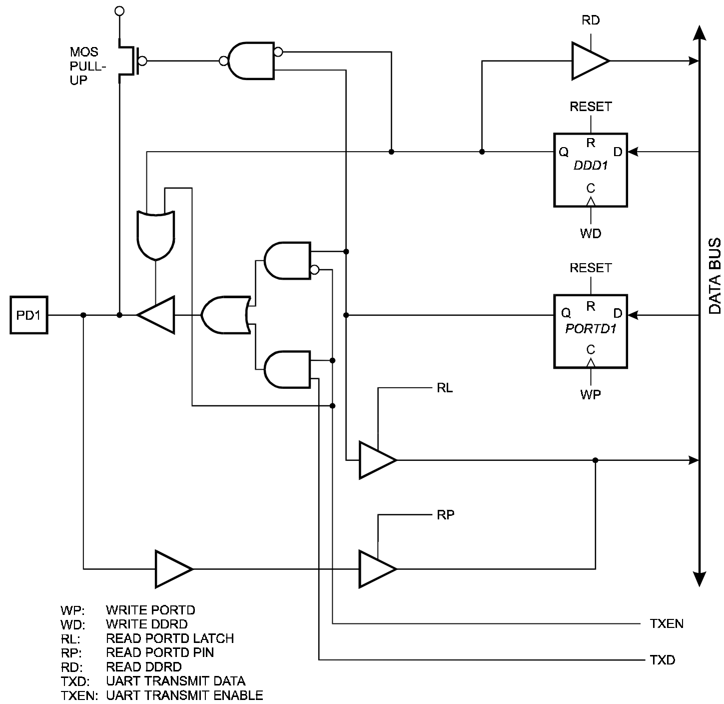


Figure 44: PORTD Schematic Diagram, Pin PD1

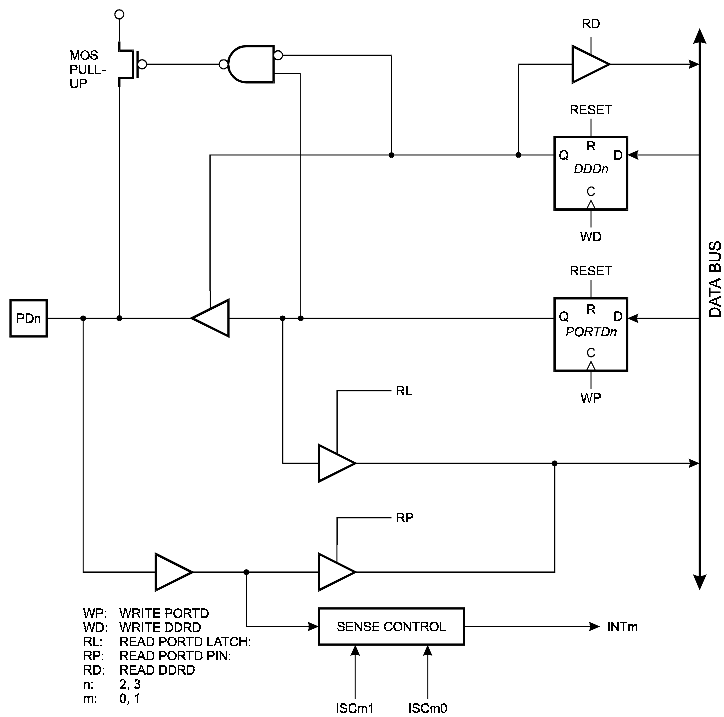


Figure 45: PORTD Schematic Diagram (Pins PD2 and PD3)

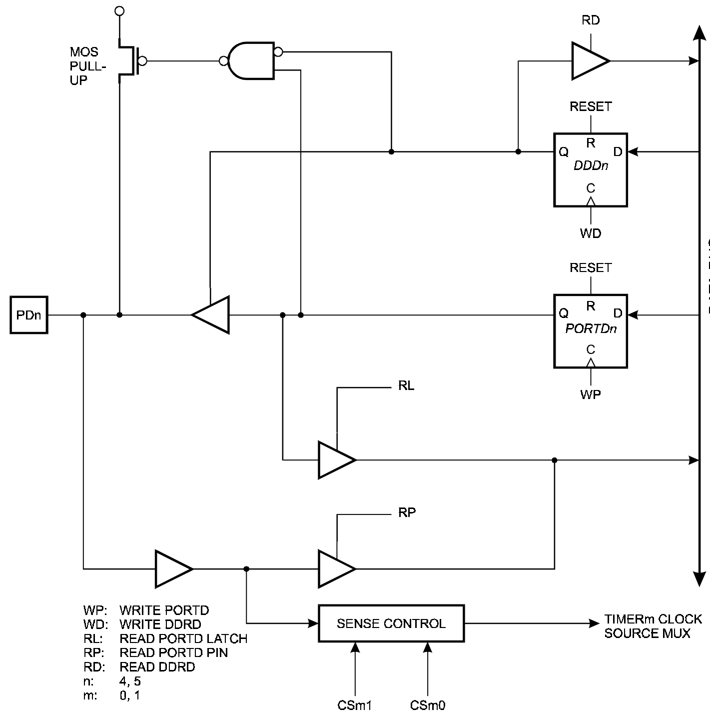


Figure 46: PORTD Schematic Diagram (Pins PD4 and PD5)

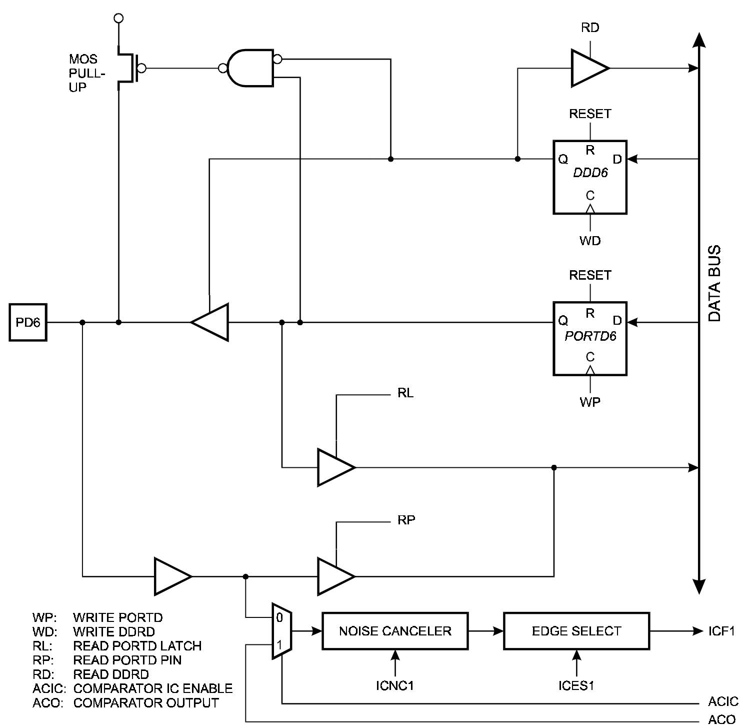


Figure 47: PORTD Schematic Diagram (Pin PD6)

Memory Programming

Program Memory Lock Bits

The AT90S2312 MCU provides two lock bits which can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in Table 19.

Table 19: Lock Bit Protection Modes

Program Lock Bits			Protection Type
Mode	LB1	LB2	
1	U	U	No program lock features
2	P	U	Further programming of the Flash is disabled
3	P	P	Same as mode 2, but verify is also disabled.

Note: The Lock Bits can only be erased with the Chip Erase operation.

Programming the Flash and EEPROM

Atmel's AT90S2312 offers 2K bytes of in-system reprogrammable Flash PEROM Program memory and 128 bytes of EEPROM Data memory.

The AT90S2312 is normally 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 serial programming mode provides a convenient way to download the Program and Data into the AT90S2312 inside the user's system.

The Program and EEPROM memory arrays on the AT90S2312 are programmed byte-by-byte in either programming modes. An auto-erase cycle is provided with the self-timed programming operation in the serial programming mode.

Parallel Programming

To be determined.

Serial Downloading

Both the Program and Data memory arrays can be programmed using the serial SPI bus while $\overline{\text{RESET}}$ is pulled to GND. The serial interface consists of pins SCK, MOSI (input) and MISO (output). After $\overline{\text{RESET}}$ is set low, the Programming Enable instruction needs to be executed first before program/erase operations can be executed.

An auto-erase cycle is built into the self-timed programming operation (in the serial mode ONLY) and there is no need to first execute the Chip Erase instruction. The Chip Erase operation turns the content of every memory location in both the Program and EEPROM arrays into $FF.

The Program and EEPROM memory arrays have separate address spaces, $000 to $7FF for Program Flash memory and $000 to $07F for EEPROM Data memory.

Either an external system clock is supplied at pin XTAL1 or a crystal needs to be connected across pins XTAL1 and XTAL2. The maximum serial clock (SCK) frequency should be less than 1/4 of the crystal frequency. With a 10 MHz oscillator clock, the maximum SCK frequency is 2.5 MHz.

SERIAL PROGRAMMING ALGORITHM

To program and verify the AT90S1200/1300 in the serial programming mode, the following sequence is recommended (See four byte instruction formats in Table 20):

1. *Power-up sequence:*
   Apply power between VCC and GND.
   Set $\overline{\text{RESET}}$ pin to 'L'.
   If a crystal is not connected across pins XTAL1 and XTAL2, apply a 1 MHz to 24 MHz clock to the XTAL1 pin and wait for at least 10 milliseconds.
2. Enable serial programming by sending the Programming Enable serial instruction to pin MOSI/PB5. The frequency of the shift clock supplied at pin SCK/PB7 needs to be less than the CPU clock at XTAL1 divided by 4.
3. The Program or EEPROM array is programmed one byte at a time by supplying the address and data together with the appropriate Write instruction. The selected memory location is first automatically erased before new data is written.
4. Any memory location can be verified by using the Read instruction which returns the content at the selected address at serial output MISO/PB6.
   $\overline{\text{DATA}}$ polling is used to indicate the end of a write cycle, which typically takes less than 2.5 ms.
5. At the end of the programming session, $\overline{\text{RESET}}$ can be set high to commence normal operation.
6. *Power-off sequence (if needed):*
   Set XTAL1 to 'L' (if a crystal is not used).
   Set $\overline{\text{RESET}}$ to 'H'
   Float all other I/O pins.
   Turn Vcc power off.

Table 20: Serial Programming Instruction Set

Instruction	Instruction Format				Operation
	Byte 1	Byte 2	Byte 3	Byte4	
Programming Enable	1010 1100	0101 0011	xxxx xxxx	xxxx xxxx	Enable Serial Programming after $\overline{\text{RESET}}$ goes low.
Chip Erase	1010 1100	100x xxxx	xxxx xxxx	xxxx xxxx	Chip erase both 2K & 128 byte memory arrays
Read Program Memory	0010 **H**000	xxxx xx**aa**	**bbbb bbbb**	**oooo oooo**	Read **H**(high or low) data **o** from Program memory at word address **a:b**
Write Program Memory	0110 **H**000	xxxx xx**aa**	**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 xxxx	x**bbb bbbb**	**oooo oooo**	Read data **o** from EEPROM memory at address **b**
Write EEPROM Memory	1110 0000	xxxx xxxx	x**bbb bbbb**	**iiii iiii**	Write data **i** to EEPROM memory at address **b**
Write Lock Bits	1010 1100	111x x0***12***	xxxx xxxx	xxxx xxxx	Write lock bits. Set bits ***1,2***=’0’ to program lock bits.
Read Device Code	0011 0000	xxxx xxxx	xxxx xxxx	**oooo oooo**	Read Device Code **o**

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

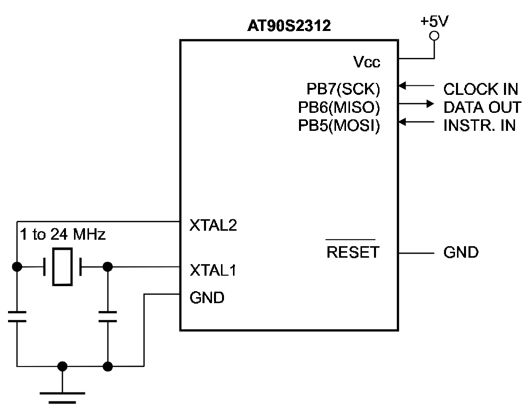


Figure 48: Serial Programming and Verify

When *writing* serial data to the AT90S2312, data is clocked on the *rising* edge of CLK.
When *reading* data from the AT90S2312, data is clocked on the *falling* edge of CLK. See Figure 49 for an explanation.

Programming Characteristics

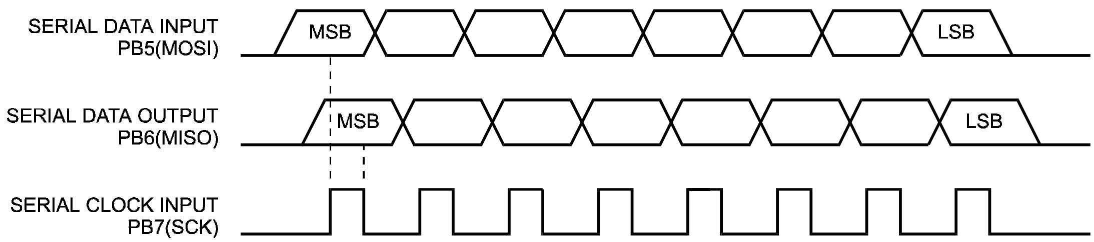


Figure 49: Serial Downloading Waveforms

Absolute Maximum Ratings

Operating Temperature.............	-55°C to +125°C
Storage Temperature................	-65°C to +150°C
Voltage on any Pin with respect to Ground................	-1.0 V to +7.0 V
Maximum Operating Voltage.....................	6.6 V
DC Output Current............................	25.0 mA

*NOTICE: Stresses beyond those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

D.C. Characteristics

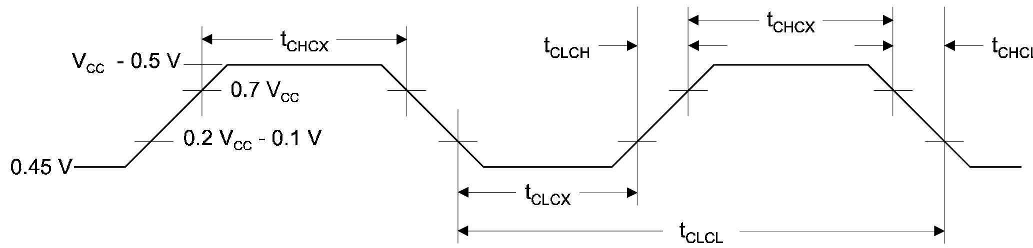
T_A=-40°C to 85°C Vcc=2.7V to 6.0V (unless otherwise noted)

Symbol	Parameter	Condition	Min	Typ	Max	Units
V_{IL}	Input Low Voltage		-0.5		0.2 Vcc - 0.1	V
V_{IH}	Input High Voltage	(Except XTAL1, RESET)	0.2 VCC + 0.9		VCC + 0.5	V
V_{IH1}	Input High Voltage	(XTAL1, RESET)	0,7 VCC		VCC + 0.5	V
V_{OL}	Output Low Voltage[1] (Ports B, D)	I_{OL} = 20 mA, VCC = 5 V I_{OL} = 10 mA, VCC = 2.7 V			0.5	V
V_{OH}	Output High Voltage (Ports B,D)	I_{HI} = 10 mA, VCC = 5 V I_{HI} = 5 mA, VCC = 2.7 V	4.5			V
I_{OH}	Output Source Current (Ports B,D)	VCC = 5 V VCC = 2.7 V			10 5	mA
I_{OL}	Output Sink Current (Port B,D)	VCC = 5 V VCC = 2.7 V			20 10	mA
RRST	Reset Pull-Up Resistor		10		50	KΩ
I_{CC}	Power Supply Current	Active Mode, 3V, 4MHz		2.5		mA
		Idle Mode 3V, 4MHz		800		uA
I_{CC}	Power Down Mode[2]	WDT enabled, 3V		50		µA
		WDT disabled, 3V		<1		µA
V_{ACIO}	Analog Comparator Input Offset Voltage	VCC = 5V			20	mV
I_{ACLK}	Analog Comparator Input Leakage Current		1	5	10	nA
t_{ACPD}	Analog Comparator Propagation Delay	VCC = 2.7 V VCC = 4.0 V		750 500		ns

Notes:

1. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
   Maximum I_{OL} per port pin: 20mA
   Maximum total I_{OL} for all output pins: 80mA
   If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
2. Minimum VCC for Power Down is 2 V.

External Clock Drive Waveforms



External Clock Drive

Symbol	Parameter	VCC = 2.7 V to 6.0 V		VCC = 4.0 V to 6.0 V		Units
		Min	Max	Min	Max	
$1/t_{CLCL}$	Oscillator Frequency	0	10	0	24	MHz
t_{CLCL}	Clock Period	100		41.7		ns
t_{CHCX}	High Time	40		16.7		ns
t_{CLCX}	Low Time	40		16.7		ns
t_{CLCH}	Rise Time		10		4.15	ns
t_{CHCL}	Fall Time		10		4.15	ns

Ordering Information

Ordering Code	Package	Operation Range
AT90S2312-PC AT90S2312-SC	20P3 20S	Commercial (0°C to 70°C)
AT90S2312-PI AT90S2312-SI	20P3 20S	Industrial (-40°C to 85°C)

Package Type	
20P3	20 Lead, 0.300” Wide Plastic Dual In-Line Package (PDIP)
20S	20 Lead, 0.300” Wide, Plastic Gull Wing Small Outline (SOIC)

AT90S2312 Register Summary

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
$3F	SREG	I	T	H	S	V	N	Z	C	3-18
$3E	Reserved									
$3D	SPL	-	SP6	SP5	SP4	SP3	SP2	SP1	SP0	3-19
$3C	Reserved									
$3B	GIMSK	INT1	INT0	-	-	-	-	-	-	3-24
$3A	Reserved									
$39	TIMSK	TOIE1	OCIE1	-	-	TICIE1	-	TOIE0	OCIE0	3-24
$38	TIFR	TOV1	OCF1	-	-	ICF1	-	TOV0	OCF0	3-25
$37	Reserved									
$36	Reserved									
$35	MCUCR	-	-	SE	SM	ISC11	ISC10	ISC01	ISC00	3-26
$34	Reserved									
$33	TCCR0	-	-	COM01	COM00	CTC0	CS02	CS01	CS00	3-29
$32	TCNT0	Timer/Counter0 (8 Bit)								3-30
$31	OCR0	Timer/Counter0 Output Compare Register								3-30
$30	Reserved									
$2F	TCCR1A	COM11	COM10	-	-	-	-	-	PWM1	3-32
$2E	TCCR1B	ICNC1	ICES1	.	-	CTC1	CS12	CS11	CS10	3-33
$2D	TCNT1H	Timer/Counter1 - Counter Register High Byte								3-32
$2C	TCNT1L	Timer/Counter1 - Counter Register Low Byte								3-32
$2B	OCR1H	Timer/Counter1 - Compare Register High Byte								3-34
$2A	OCR1L	Timer/Counter1 - Compare Register Low Byte								3-34
$29	Reserved									
$28	Reserved									
$27	Reserved									
$26	Reserved									
$25	ICR1H	Timer/Counter1 - Input Capture Register High Byte								3-35
$24	ICR1L	Timer/Counter1 - Input Capture Register Low Byte								3-35
$23	Reserved									
$22	Reserved									
$21	WDTCR	-	-	-	-	WDE	WDP2	WDP1	WDP0	3-37
$20	Reserved									
$1F	Reserved									
$1E	EEAR	-	EEPROM Address Register							3-38
$1D	EEDR	EEPROM Data register								3-38
$1C	EECR	-	-	-	-	-	-	EEWE	EERE	3-38
$1B	Reserved									
$1A	Reserved									
$19	Reserved									
$18	PORTB	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0	3-48
$17	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	3-48
$16	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	3-48
$15	Reserved									
$14	Reserved									
$13	Reserved									
$12	PORTD	-	PD6	PD5	PD4	PD3	PD2	PD1	PD0	3-53
$11	DDRD	-	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	3-53
$10	PIND	-	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	3-53
$0F	Reserved									
$0E	Reserved									
$0D	Reserved									
$0C	UDR	UART I/O Data Register								3-42
$0B	USR	RXC	TXC	UDRE	FE	OR	-	-	-	3-42
$0A	UCR	RXCIE	TXCIE	UDRIE	RXEN	TXEN	CHR9	RXB8	TXB8	3-43
$09	UBRR	UART Baud Rate Register								3-45
$08	ACSR	ACD	-	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	3-45
…	Reserved									
$00	Reserved									

AT90S2312 Instruction Set Summary

Mnemonics	Operands	Description	Operation	Flags	#Clocks
ARITHMETIC AND LOGIC INSTRUCTIONS					
ADD	Rd, Rr	Add two Registers	Rd ← Rd + Rr	Z,C,N,V,H	1
ADC	Rd, Rr	Add with Carry two Registers	Rd ← Rd + Rr + C	Z,C,N,V,H	1
ADIW	Rdl,K	Add Immediate to Word	Rdh:Rdl ← Rdh:Rdl + K	Z,C,N,V,S	2
SUB	Rd, Rr	Subtract two Registers	Rd ← Rd - Rr	Z,C,N,V,H	1
SUBI	Rd, K	Subtract Constant from Register	Rd ← Rd - K	Z,C,N,V,H	1
SBIW	Rdl,K	Subtract Immediate from Word	Rdh:Rdl ← Rdh:Rdl - K	Z,C,N,V,S	2
SBC	Rd, Rr	Subtract with Carry two Registers	Rd ← Rd - Rr - C	Z,C,N,V,H	1
SBCI	Rd, K	Subtract with Carry Constant from Reg.	Rd ← Rd - K - C	Z,C,N,V,H	1
AND	Rd, Rr	Logical AND Registers	Rd ← Rd • Rr	Z,N,V	1
ANDI	Rd, K	Logical AND Register and Constant	Rd ← Rd • K	Z,N,V	1
OR	Rd, Rr	Logical OR Registers	Rd ← Rd v Rr	Z,N,V	1
ORI	Rd, K	Logical OR Register and Constant	Rd ← Rd v K	Z,N,V	1
EOR	Rd, Rr	Exclusive OR Registers	Rd ← Rd ⊕ Rr	Z,N,V	1
COM	Rd	One's Complement	Rd ← $FF - Rd	Z,C,N,V	1
NEG	Rd	Two's Complement	Rd ← $00 - Rd	Z,C,N,V,H	1
SBR	Rd,K	Set Bit(s) in Register	Rd ← Rd v K	Z,N,V	1
CBR	Rd,K	Clear Bit(s) in Register	Rd ← Rd • ($FF - K)	Z,N,V	1
INC	Rd	Increment	Rd ← Rd + 1	Z,N,V	1
DEC	Rd	Decrement	Rd ← Rd - 1	Z,N,V	1
TST	Rd	Test for Zero or Minus	Rd ← Rd • Rd	Z,N,V	1
CLR	Rd	Clear Register	Rd ← Rd ⊕ Rd	Z,N,V	1
SER	Rd	Set Register	Rd ← $FF	None	1
BRANCH INSTRUCTIONS					
RJMP	k	Relative Jump	PC ← PC + k + 1	None	2
IJMP		Indirect Jump to (Z)	PC ← Z	None	2
RCALL	k	Relative Subroutine Call	PC ← PC + k + 1	None	3
ICALL		Indirect Call to (Z)	PC ← Z	None	3
RET		Subroutine Return	PC ← STACK	None	4
RETI		Interrupt Return	PC ← STACK	I	4
CPSE	Rd,Rr	Compare, Skip if Equal	if (Rd = Rr) PC ← PC + 2 or 3	None	1 / 2
CP	Rd,Rr	Compare	Rd - Rr	Z, N,V,C,H	1
CPC	Rd,Rr	Compare with Carry	Rd - Rr - C	Z, N,V,C,H	1
CPI	Rd,K	Compare Register with Immediate	Rd - K	Z, N,V,C,H	1
SBRC	Rr, b	Skip if Bit in Register Cleared	if (Rr(b)=0) PC ← PC + 2 or 3	None	1 / 2
SBRS	Rr, b	Skip if Bit in Register is Set	if (Rr(b)=1) PC ← PC + 2 or 3	None	1 / 2
SBIC	Rr, b	Skip if Bit in I/O Register Cleared	if (Rr(b)=0) PC ← PC + 2 or 3	None	2 / 3
SBIS	Rr, b	Skip if Bit in I/O Register is Set	if (Rr(b)=1) PC ← PC + 2 or 3	None	2 / 3
BRBS	s, k	Branch if Status Flag Set	if (SREG(s) = 1) then PC←PC + k + 1	None	1 / 2
BRBC	s, k	Branch if Status Flag Cleared	if (SREG(s) = 0) then PC←PC + k + 1	None	1 / 2
BREQ	k	Branch if Equal	if (Z = 1) then PC ← PC + k + 1	None	1 / 2
BRNE	k	Branch if Not Equal	if (Z = 0) then PC ← PC + k + 1	None	1 / 2
BRCS	k	Branch if Carry Set	if (C = 1) then PC ← PC + k + 1	None	1 / 2
BRCC	k	Branch if Carry Cleared	if (C = 0) then PC ← PC + k + 1	None	1 / 2
BRSH	k	Branch if Same or Higher	if (C = 0) then PC ← PC + k + 1	None	1 / 2
BRLO	k	Branch if Lower	if (C = 1) then PC ← PC + k + 1	None	1 / 2
BRMI	k	Branch if Minus	if (N = 1) then PC ← PC + k + 1	None	1 / 2
BRPL	k	Branch if Plus	if (N = 0) then PC ← PC + k + 1	None	1 / 2
BRGE	k	Branch if Greater or Equal, Signed	if (N ⊕ V= 0) then PC ← PC + k + 1	None	1 / 2
BRLT	k	Branch if Less Than Zero, Signed	if (N ⊕ V= 1) then PC ← PC + k + 1	None	1 / 2
BRHS	k	Branch if Half Carry Flag Set	if (H = 1) then PC ← PC + k + 1	None	1 / 2
BRHC	k	Branch if Half Carry Flag Cleared	if (H = 0) then PC ← PC + k + 1	None	1 / 2
BRTS	k	Branch if T Flag Set	if (T = 1) then PC ← PC + k + 1	None	1 / 2
BRTC	k	Branch if T Flag Cleared	if (T = 0) then PC ← PC + k + 1	None	1 / 2
BRVS	k	Branch if Overflow Flag is Set	if (V = 1) then PC ← PC + k + 1	None	1 / 2
BRVC	k	Branch if Overflow Flag is Cleared	if (V = 0) then PC ← PC + k + 1	None	1 / 2
BRIE	k	Branch if Interrupt Enabled	if (I = 1) then PC ← PC + k + 1	None	1 / 2
BRID	k	Branch if Interrupt Disabled	if (I = 0) then PC ← PC + k + 1	None	1 / 2

(continued)

AT90S2312 Instruction Set Summary (Continued)

Mnemonic	Operands	Description	Operation	Flags	#Clocks
DATA TRANSFER INSTRUCTIONS					
MOV	Rd, Rr	Move Between Registers	Rd ← Rr	None	1
LDI	Rd, K	Load Immediate	Rd ← K	None	1
LD	Rd, X	Load Indirect	Rd ← (X)	None	2
LD	Rd, X+	Load Indirect and Post-Inc.	Rd ← (X), X ← X + 1	None	2
LD	Rd, - X	Load Indirect and Pre-Dec.	X ← X - 1, Rd ← (X)	None	2
LD	Rd, Y	Load Indirect	Rd ← (Y)	None	2
LD	Rd, Y+	Load Indirect and Post-Inc.	Rd ← (Y), Y ← Y + 1	None	2
LD	Rd, - Y	Load Indirect and Pre-Dec.	Y ← Y - 1, Rd ← (Y)	None	2
LDD	Rd,Y+q	Load Indirect with Displacement	Rd ← (Y + q)	None	2
LD	Rd, Z	Load Indirect	Rd ← (Z)	None	2
LD	Rd, Z+	Load Indirect and Post-Inc.	Rd ← (Z), Z ← Z+1	None	2
LD	Rd, -Z	Load Indirect and Pre-Dec.	Z ← Z - 1, Rd ← (Z)	None	2
LDD	Rd, Z+q	Load Indirect with Displacement	Rd ← (Z + q)	None	2
ST	X, Rr	Store Indirect	(X) ← Rr	None	2
ST	X+, Rr	Store Indirect and Post-Inc.	(X) ← Rr, X ← X + 1	None	2
ST	- X, Rr	Store Indirect and Pre-Dec.	X ← X - 1, (X) ← Rr	None	2
ST	Y, Rr	Store Indirect	(Y) ← Rr	None	2
ST	Y+, Rr	Store Indirect and Post-Inc.	(Y) ← Rr, Y ← Y + 1	None	2
ST	- Y, Rr	Store Indirect and Pre-Dec.	Y ← Y - 1, (Y) ← Rr	None	2
STD	Y+q,Rr	Store Indirect with Displacement	(Y + q) ← Rr	None	2
ST	Z, Rr	Store Indirect	(Z) ← Rr	None	2
ST	Z+, Rr	Store Indirect and Post-Inc.	(Z) ← Rr, Z ← Z + 1	None	2
ST	-Z, Rr	Store Indirect and Pre-Dec.	Z ← Z - 1, (Z) ← Rr	None	2
STD	Z+q,Rr	Store Indirect with Displacement	(Z + q) ← Rr	None	2
LPM		Load Program Memory	R0 ← (Z)	None	3
IN	Rd, P	In Port	Rd ← P	None	1
OUT	P, Rr	Out Port	P ← Rr	None	1
PUSH	Rr	Push Register on Stack	STACK ← Rr	None	2
POP	Rd	Pop Register from Stack	Rd ← STACK	None	2
BIT AND BIT-TEST INSTRUCTIONS					
SBI	P,b	Set Bit in I/O Register	I/O(P,b) ← 1	N	2
CBI	P,b	Clear Bit in I/O Register	I/O(P,b) ← 0	N	2
LSL	Rd	Logical Shift Left	Rd(n+1) ← Rd(n), Rd(0) ← 0	Z,C,N,V	1
LSR	Rd	Logical Shift Right	Rd(n) ← Rd(n+1), Rd(7) ← 0	Z,C,N,V	1
ROL	Rd	Rotate Left Through Carry	Rd(0)←C,Rd(n+1)← Rd(n),C←Rd(7)	Z,C,N,V	1
ROR	Rd	Rotate Right Through Carry	Rd(7)←C,Rd(n)← Rd(n+1),C←Rd(0)	Z,C,N,V	1
ASR	Rd	Arithmetic Shift Right	Rd(n) ← Rd(n+1), n=0..6	Z,C,N,V	1
SWAP	Rd	Swap Nibbles	Rd(3..0)←Rd(7..4),Rd(7..4)←Rd(3..0)	None	1
BSET	s	Flag Set	SREG(s) ← 1	SREG(s)	1
BCLR	s	Flag Clear	SREG(s) ← 0	SREG(s)	1
BST	Rr, b	Bit Store from Register to T	T ← Rr(b)	T	1
BLD	Rd, b	Bit load from T to Register	Rd(b) ← T	None	1
SEC		Set Carry	C ← 1	C	1
CLC		Clear Carry	C ← 0	C	1
SEN		Set Negative Flag	N ← 1	N	1
CLN		Clear Negative Flag	N ← 0	N	1
SEZ		Set Zero Flag	Z ← 1	Z	1
CLZ		Clear Zero Flag	Z ← 0	Z	1
SEI		Global Interrupt Enable	I ← 1	I	1
CLI		Global Interrupt Disable	I ← 0	I	1
SES		Set Signed Test Flag	S ← 1	S	1
CLS		Clear Signed Test Flag	S ← 0	S	1
SEV		Set Twos Complement Overflow	V ← 1	V	1
CLV		Clear Twos Complement Overflow	V ← 0	V	1
SET		Set T in SREG	T ← 1	T	1
CLT		Clear T in SREG	T ← 0	T	1
SEH		Set Half Carry Flag in SREG	H ← 1	H	1
CLH		Clear Half Carry Flag in SREG	H ← 0	H	1
NOP		No Operation		None	1
SLEEP		Sleep	(see specific descr. for Sleep function)	None	1
WDR		Watchdog Reset	(see specific descr. for WDR/timer)	None	1