Features

- **Utilizes the *AVR*® Enhanced RISC Architecture**
- **AV*R* - High Performance and Low Power RISC Architecture**
- **89 Powerful Instructions - Most Single Clock Cycle Execution**
- **1K bytes of In-System Reprogrammable Downloadable Flash**
  - **SPI Serial Interface for Program Downloading**
  - **Endurance: 1,000 Write/Erase Cycles**
- **64 bytes EEPROM**
  - **Endurance: 100,000 Write/Erase Cycles**
- **32 x 8 General Purpose Working Registers**
- **15 Programmable I/O Lines**
- **V_{CC}: 2.7 - 6.0V**
- **Fully Static Operation, 0 - 16 MHz**
- **Instruction Cycle Time: 62.5 ns @ 16 MHz**
- **One 8-Bit Timer/Counter with Separate Prescaler**
- **External and Internal Interrupt Sources**
- **Programmable Watchdog Timer with On-Chip Oscillator**
- **On-Chip Analog Comparator**
- **Low Power Idle and Power Down Modes**
- **Programming Lock for Software Security**
- **20-Pin Device**
- **Selectable On-Chip RC Oscillator for Zero External Components**

Description

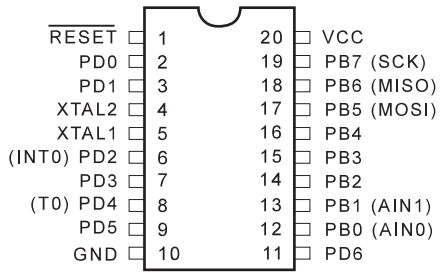
The AT90S1200 is a low-power CMOS 8-bit microcontroller based on the *AVR*® enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the AT90S1200 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

The AVR core combines a rich instruction set with the 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.

(continued)

Pin Configuration

PDIP/SOIC/SSOP



8-Bit AVR® Microcontroller with 1K bytes Downloadable Flash

Preliminary

0838A

Block Diagram

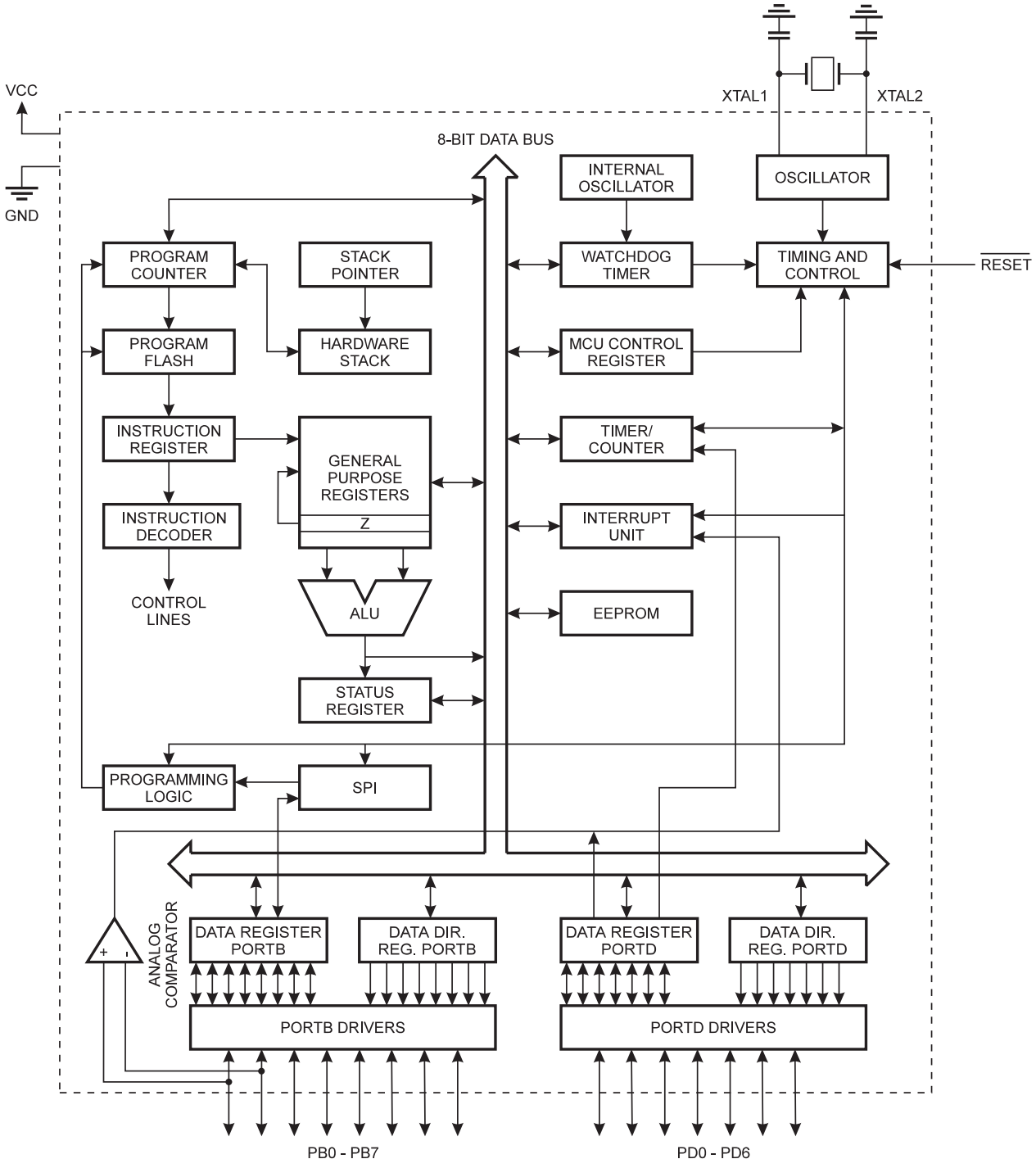


Figure 1. The AT90S1200 Block Diagram

Description (Continued)

The architecture supports high level languages efficiently as well as extremely dense assembler code programs. The AT90S1200 provides the following features: 1K bytes of Downloadable Flash, 64 bytes EEPROM, 15 general purpose I/O lines, 32 general purpose working registers, internal and external interrupts, programmable Watchdog Timer with internal oscillator, an SPI serial port for program downloading and two software selectable power saving modes. The Idle Mode stops the CPU while allowing the registers, timer/counter, watchdog and interrupt system to continue functioning. The power down mode saves the register contents but freezes the oscillator, disabling all other chip functions until the next external interrupt or hardware reset.

The device is manufactured using Atmel's high density 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 AT90S1200 is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications.

The AT90S1200 *AVR* is supported with a full suite of program and system development tools including: 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 (I_{IL}) if the internal pullups are activated.

Port B also serves the functions of various special features of the AT90S1200 as listed on Page 2-27.

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 AT90S1200 as listed on Page 2-31.

$\overline{\text{RESET}}$

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.

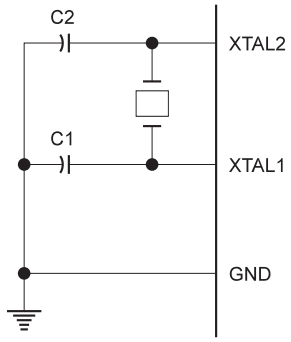


Figure 2. Oscillator Connections

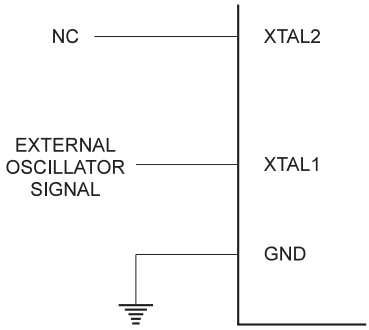


Figure 3. External Clock Drive Configuration

On-Chip RC Oscillator

An on-chip RC oscillator running at a fixed frequency of 1 MHz can be selected as the MCU clock source. If enabled, the AT90S1200 can operate with no external components. A control bit - RCEN in the Flash Memory selects the on-chip RC oscillator as the clock source when programmed (‘0’). The AT90S1200 is normally shipped with this bit unprogrammed (‘1’). Parts with this bit programmed can be ordered as AT90S1200A. The RCEN-bit can be changed by parallel programming only. When using the on-chip RC oscillator for serial program downloading, the RCEN bit must programmed in parallel programming mode first.

AT90S1200 *AVR* Enhanced RISC Microcontroller CPU

The AT90S1200 *AVR* RISC microcontroller is upward compatible with the *AVR* Enhanced RISC Architecture. The programs written for the AT90S1200 MCU are 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.

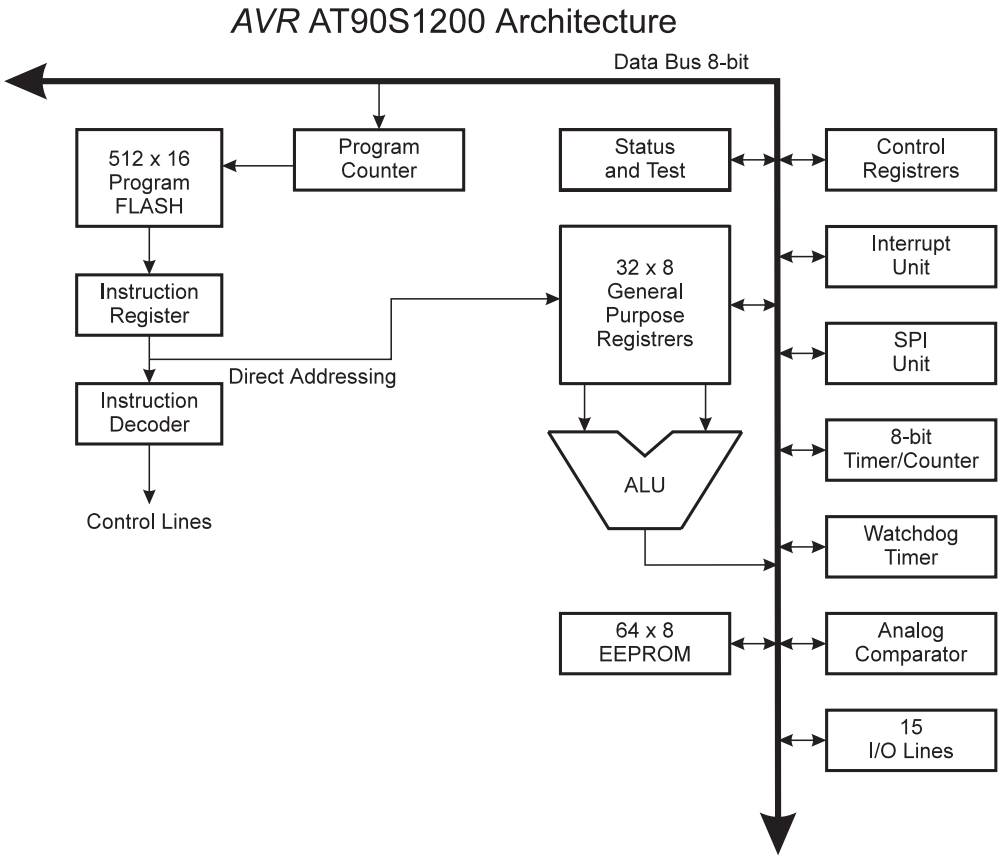


Figure 4. The AT90S1200 *AVR* Enhanced RISC Architecture

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 AT90S1200 *AVR* Enhanced RISC microcontroller architecture. The *AVR* uses a Harvard architecture concept - with separate memories and buses for program and data memories. 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 relative call instructions, the whole 512 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 a 3 level deep hardware stack dedicated for subroutines and interrupts.

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.

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.

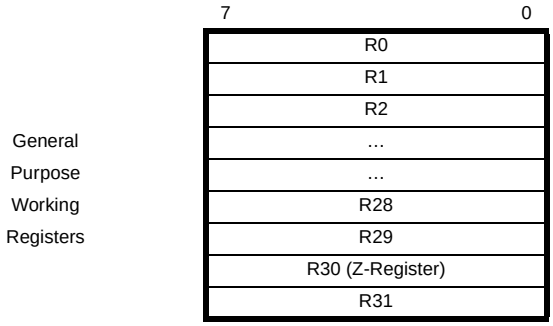


Figure 5. *AVR* CPU General Purpose Working Registers

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.

Register 30 also serves as an 8-bit pointer for indirect address of the register file.

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 AT90S1200 contains 1K bytes on-chip downloadable Flash memory for program storage. Since all instructions are single 16-bit words, the Flash is organized as 512 x 16 words. The Flash memory has an endurance of at least 1000 write/erase cycles.

The AT90S1200 Program Counter is 9-bit wide, thus addressing the 512 words Flash program memory.

See Page 2-34 for a detailed description on Flash data downloading.

The Program and Data Addressing Modes

The AT90S1200 *AVR* Enhanced RISC Microcontroller supports powerful and efficient addressing modes. This section describes the different addressing modes supported in the AT90S1200. 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

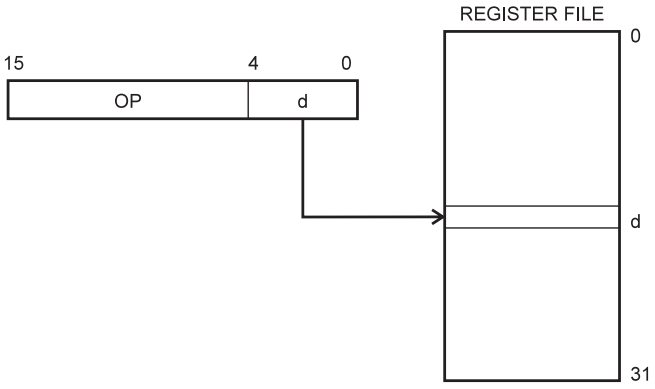


Figure 6. Direct Single Register Addressing

The operand is contained in register d (Rd).

REGISTER INDIRECT

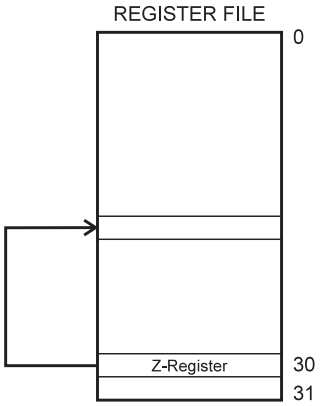


Figure 7. Indirect Register Addressing

The register accessed is the one pointed to by the Z-register (R30).

REGISTER DIRECT, TWO REGISTERS RD AND RR

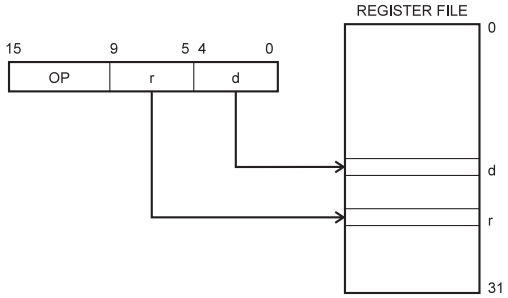


Figure 8. Direct Register Addressing, Two Registers

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

I/O DIRECT

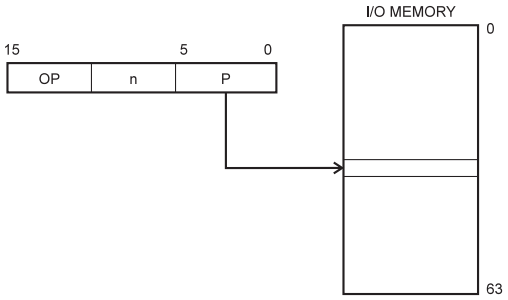


Figure 9. I/O Direct Addressing

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

RELATIVE PROGRAM ADDRESSING, RJMP AND RCALL

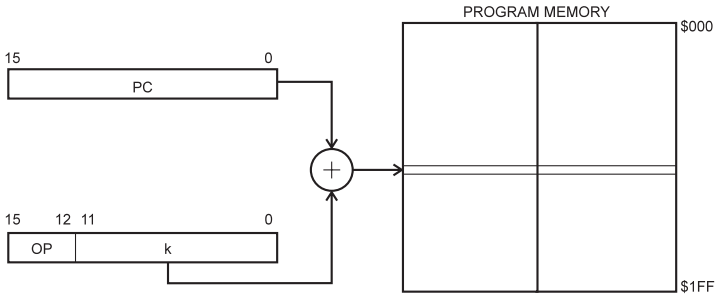


Figure 10. Relative Program Memory Addressing

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

Subroutine and Interrupt Hardware Stack

The AT90S1200 uses a 3 level deep hardware stack for subroutines and interrupts. The hardware stack is 9 bit wide and stores the Program Counter - PC - return address while subroutines and interrupts are executed.

RCALL instructions and interrupts push the PC return address onto stack level 0, and the data in the other stack levels 1-2 are pushed one level deeper in the stack. When a RET or RETI instruction is executed the returning PC is fetched from the stack level 0, and the data in the other stack levels 1-2 are popped one level in the stack.

If more than 3 subsequent subroutine calls or interrupts are executed, the first values written to the stack are overwritten.

The EEPROM Data Memory

The AT90S1200 contains 64 bytes of data EEPROM 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 2-24 specifying the EEPROM address register, the EEPROM data register, and the EEPROM control register. For the SPI data downloading, see Page 2-40 for a detailed description.

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 11 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.

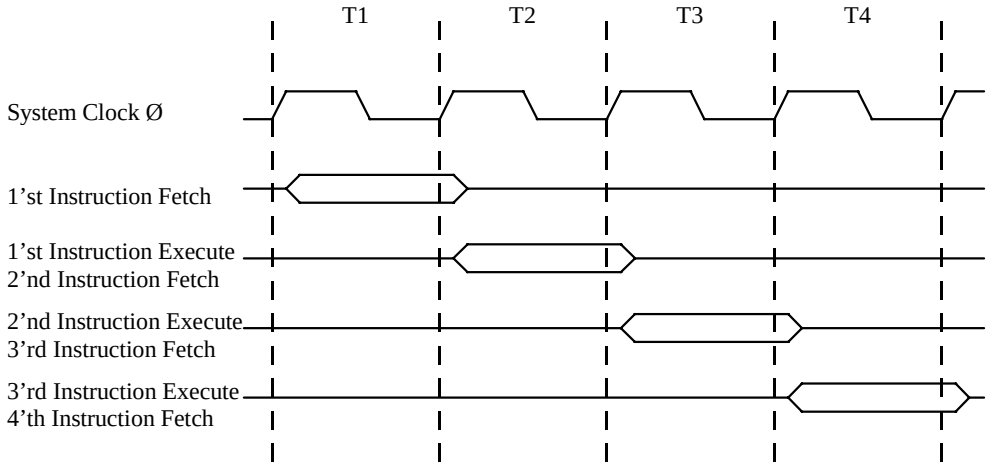


Figure 11. The Parallel Instruction Fetches and Instruction Executions

Figure 12 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.

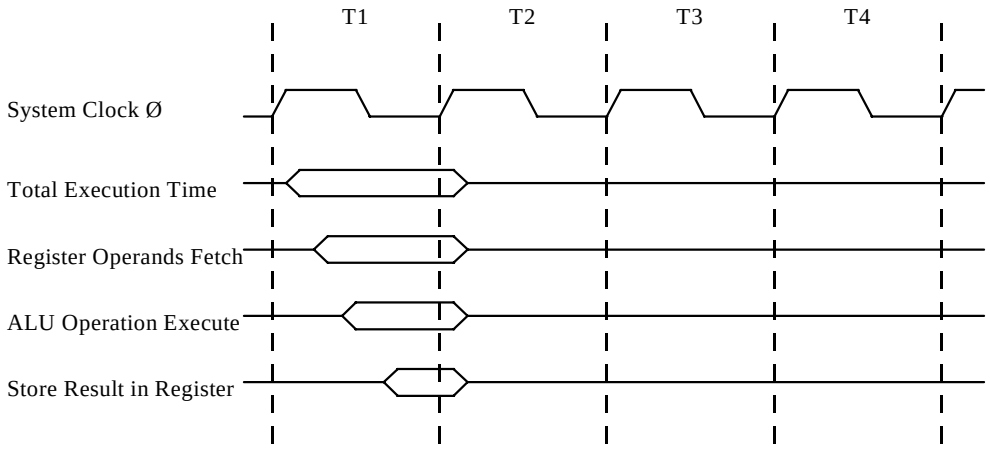


Figure 12. Single Cycle ALU Operation

I/O Memory

The I/O space definition of the AT90S1200 is shown in the following table:

Table 1. The AT90S1200 I/O Space

Address Hex	Name	Function
$3F	SREG	Status REGister
$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)
$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
$08	ACSR	Analog Comparator Control and Status Register

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

All the different AT90S1200 I/Os 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 - $1F are directly bit-accessible using the SBI and CBI instructions. In these registers, the value of single bits can be checked by using the SBIS and SBIC instructions. Refer to the instruction set chapter for more details.

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

THE STATUS REGISTER - SREG

The AV*R* 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/TIMSK. If the global interrupt enable register is cleared (zero), none of the interrupts are enabled, independent of the GIMSK/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⊕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.

Reset and Interrupt Handling

The AT90S1200 provides 3 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
4	$002	TIMER0, OVF0	Timer/Counter0 Overflow
5	$003	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     TIM0_OVF     ; Timer0 overflow handle
$003              rjmp     ANA_COMP     ; Analog Comparator Handle
;
$004      MAIN:   <instr>  xxx          ; Main program start
  …         …        …      …
```

RESET SOURCES

The AT90S1200 has three sources of reset:

- Power-On Reset. The MCU is reset when a supply voltage is applied to the V_{CC} 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 13 shows the reset logic. Table 3 defines the timing and electrical parameters of the reset circuitry. Note that Power On Reset timing is clocked by the internal RC oscillator. Refer to characterization dat-data for RC oscillator frequency at other V_{CC} voltages.

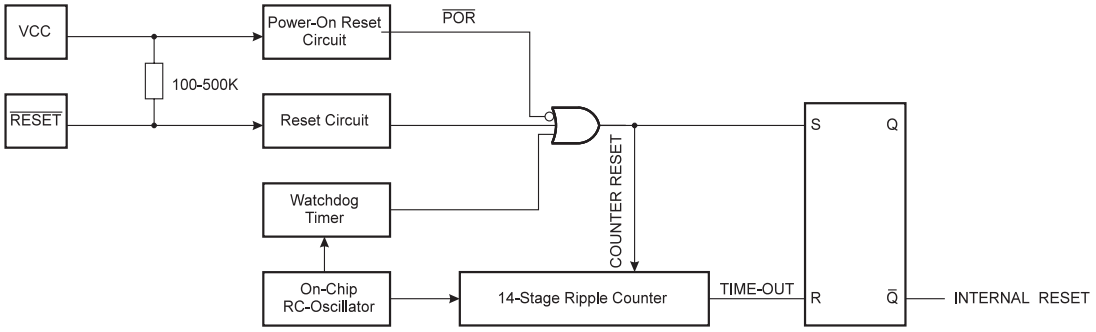


Figure 13. Reset Logic

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

Symbol	Parameter	Min	Typ	Max	Units
V_{POT}	Power-On Reset Threshold Voltage	1.8	2	2.2	V
V_{RST}	Pin Threshold Voltage		$V_{CC}/2$		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 V_{CC} has reached a safe level. As shown in Figure 13, an internal timer clocked from the Watchdog timer oscillator prevents the MCU from starting until after a certain period after V_{CC} has reached the Power-On Threshold voltage - V_{POT}, regardless of the V_{CC} rise time (see Figure 14 and Figure 15). The total reset period is the Power-On Reset period - t_{POR} + the Delay Time-out period - t_{TOUT}.

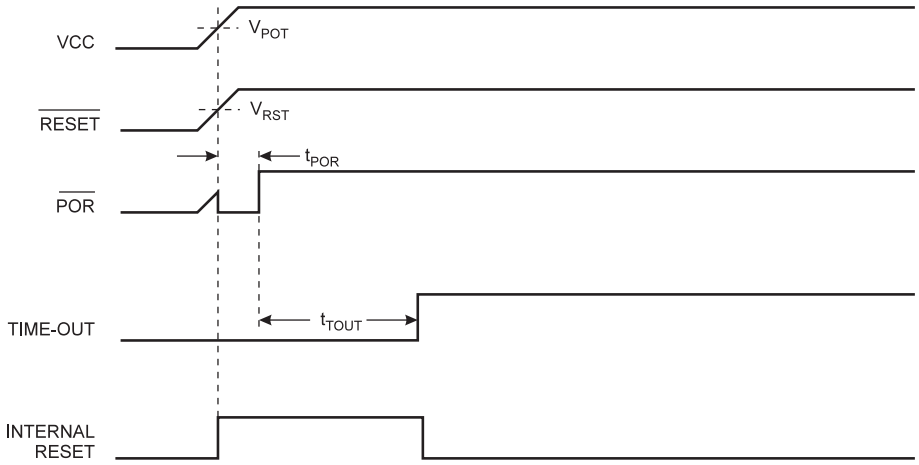


Figure 14. MCU Start-Up, $\overline{\text{RESET}}$ Tied to V_{CC} or Unconnected. Rapidly Rising V_{CC}.

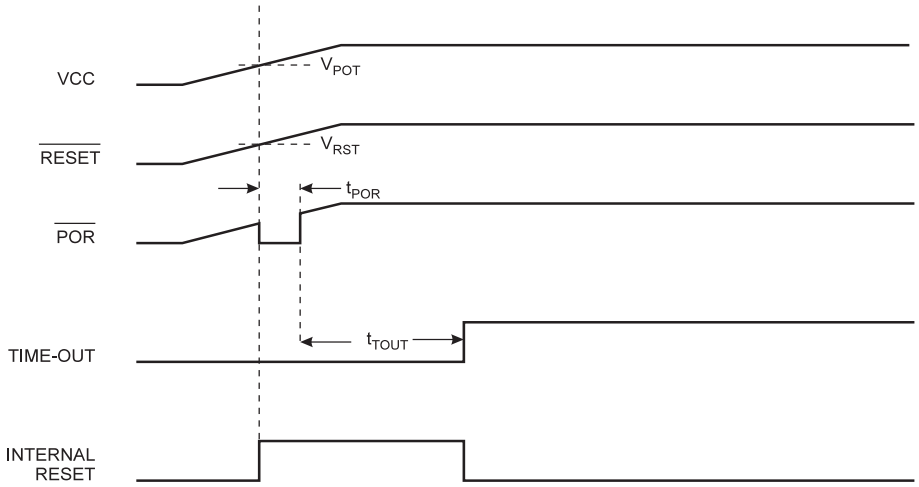


Figure 15. MCU Start-Up, $\overline{\text{RESET}}$ Tied to V_{CC} or Unconnected. Slowly Rising V_{CC}

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 V_{CC} will have the same effect. By holding the $\overline{\text{RESET}}$ pin low for a period after V_{CC} has been applied, the Power-On Reset period can be extended. Refer to Figure 16 for a timing example on this.

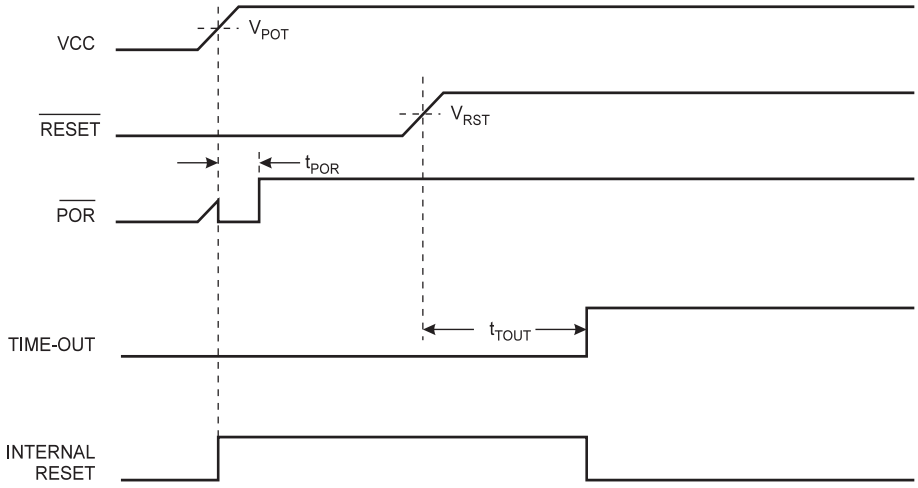


Figure 16. MCU Start-Up, $\overline{\text{RESET}}$ Controlled Externally

EXTERNAL RESET

An external reset is generated by a low level on the pin. The pin must be held low for at least two crystal clock cycles. When 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.

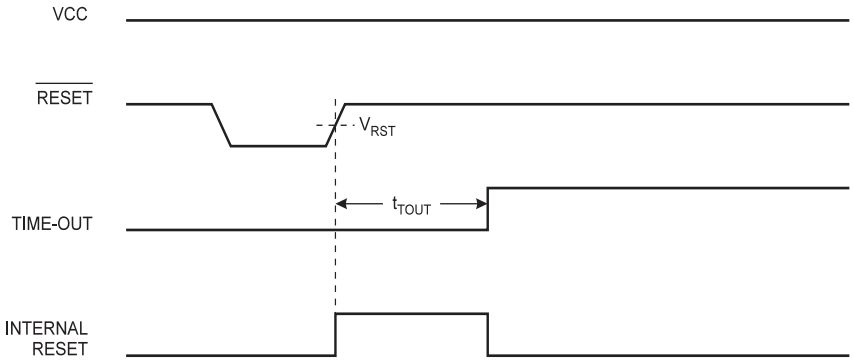


Figure 17. External Reset During Operation

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 2-23 for details on operation of the Watchdog.

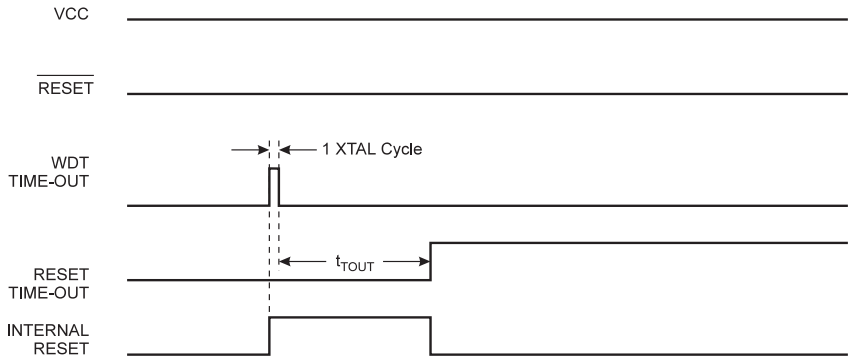


Figure 18. Watchdog Reset During Operation

INTERRUPT HANDLING

The AT90S1200 has two Interrupt Mask control registers GIMSK - General Interrupt MASK register - at I/O space address $3B and the TIMSK - Timer/Counter Interrupt MaSK register at I/O address $39.

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.

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	-	INT0	-	-	-	-	-	-	GIMSK
Read/Write	R	R/W	R	R	R	R	R	R	
Initial value	0	0	0	0	0	0	0	0	

Bit 7 - Res : Reserved bit:

This bit is a reserved bit in the AT90S1200 and always read zero.

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 bit 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. INT0 can activated even if the pin is configured as an output. See also Page 2-18.

Bits 5..0 - Res : Reserved bits:

These bits are reserved bits in the AT90S1200 and always read as zero.

THE TIMER/COUNTER INTERRUPT MASK REGISTER - TIMSK

Bit	7	6	5	4	3	2	1	0	
$39	-	-	-	-	-	-	TOIE0	-	TIMSK
Read/Write	R	R	R	R	R	R	R/W	R	
Initial value	0	0	0	0	0	0	0	0	

Bits 7..2 - Res : Reserved bits:
These bits are reserved bits in the AT90S1200 and always read 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 $002) 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 - Res: Reserved bit:
This bits is a reserved bit in the AT90S1200 and always reads zero.

THE TIMER/COUNTER INTERRUPT FLAG REGISTER - TIFR

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

Bits 7..2 - Res : Reserved bits:
These bits are reserved bits in the AT90S1200 and always read 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 - Res: Reserved bit:
This bits is a reserved bit in the AT90S1200 and always reads zero.

EXTERNAL INTERRUPTS

The external interrupt is triggered by the INT0 pin. The interrupt can trigger on rising edge, falling edge or level. This is set up as described in the specification for the MCU control register - MCUCR. When INT0 is level triggered, the interrupt is pending as long as INT0 is held low.

The interrupt is triggered even if INT0 is configured as an output. This provides a way to generate a software interrupt.

The interrupt flag can not be directly accessed by the user. If an external edge triggered interrupt is suspected to be pending, the flag can be cleared as follows.

1. Disable the external interrupt by clearing the INT0 flag in GIMSK.
2. Select level triggered interrupt.
3. Select desired interrupt edge.
4. Re-enable the external interrupt by setting INT0 in GIMSK.

INTERRUPT RESPONSE TIME

The interrupt execution response for all the enabled *AVR* interrupts is 4 clock cycles minimum. 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 (9 bits) is pushed onto the Stack. The vector is a relative jump to the interrupt routine, and this jump takes 2 clock cycles. If an interrupt occurs during execution of a multi-cycle instruction, this instruction is completed before the interrupt is served.

A return from an interrupt handling routine takes 4 clock cycles. During these 4 clock cycles, the Program Counter (9 bits) is popped back from the Stack. When *AVR* exits from an interrupt, it will always return to the main program and execute one more instruction before any pending interrupt is served.

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

Note that the Subroutine and Interrupt Stack is a 3-level true hardware stack, and if more than 3 nested subroutines and interrupts are executed, only the most recent 3 return addresses are stored.

THE MCU CONTROL REGISTER - MCUCR

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

Bit	7	6	5	4	3	2	1	0	
$35	-	-	SE	SM	-	-	ISC01	ISC00	MCUCR
Read/Write	R	R	R/W	R/W	R	R	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 AT90S1200 and always read 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 - Res : Reserved bits:

These bits are reserved bits in the AT90S1200 and always read zero.

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 in the GIMSK register is set. The level and edges on the external INT0 pin that activate the interrupt are defined as:

Table 4. 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 interrupt routine, and resumes execution from the instruction following SLEEP. The contents of the register file and the I/O memory are unaltered. If a reset occurs during sleep mode, the MCU wakes up and executes from the Reset vector.

Note that if a *level* triggered interrupt is used for wake-up from power down, 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 external 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 level triggered interrupt can wake up the MCU.

Timer / Counter

The AT90S1200 provides one general purpose 8-bit Timer/Counter. The Timer/Counter gets the prescaled clock from the 10-bit prescaling timer. The Timer/Counter 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 19 shows the general Timer/Counter prescaler.

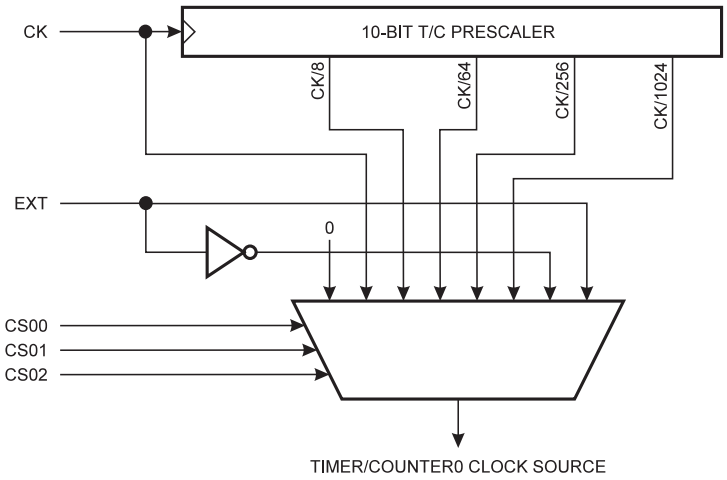


Figure 19. Timer/Counter0 Prescaler

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

The 8-bit Timer/Counter0

Figure 20 shows the block diagram for Timer/Counter0.

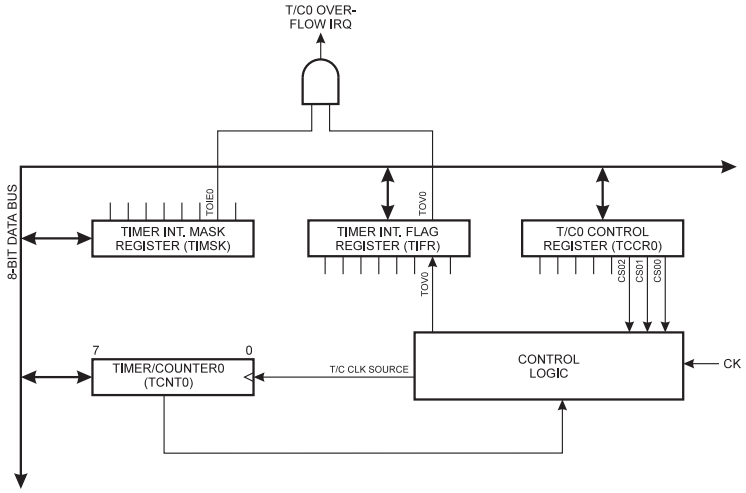


Figure 20. Timer/Counter 0 Block Diagram

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 overflow status flag is found in the Timer/Counter Interrupt Flag Register - TIFR. 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 between two external clock transitions must be at least one internal CPU clock period. The external clock signal is sampled on the rising edge of the internal CPU clock.

The 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 CONTROL REGISTER - TCCR0

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

Bits 7..3 - Res : Reserved bits:

These bits are reserved bits in the AT90S1200 and always read zero.

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

The Clock Select0 bits 2,1 and 0 define the prescaling source of Timer0.

Table 5. Clock 0 Prescale Select

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

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 AT90S1200 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 timer clock cycle following the write operation.

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 cycles. Refer to RC Oscillator Frequency graph on page 2-46. These values apply at V_{CC} = 5V. See characterization data for RC oscillator frequency at other V_{CC} voltages. The WDR - Watchdog Reset - instruction resets the Watchdog Timer. Eight different clock cycle periods can be selected to determine the maximum period between two WDR instructions to avoid that the Watchdog Timer resets the MCU. If the reset period expires without another WDR instruction, the AT90S1200 resets and executes from the reset vector. For timing details on the Watchdog reset, refer to Page 2-17.

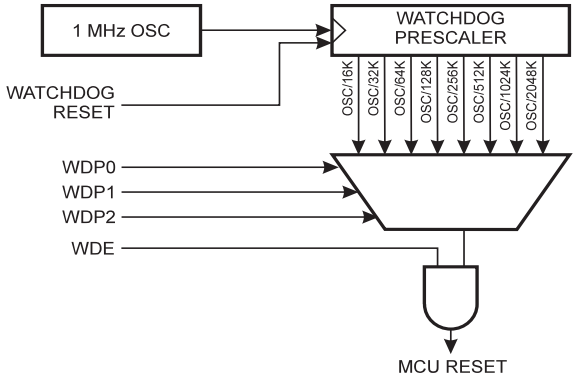


Figure 21. Watchdog Timer

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 AT90S1200 and will always read as zero.

Bit 3 - WDE : Watchdog 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..0 : Watchdog Timer Prescaler 2,1 and 0:
The WDP2..0 determine the Watchdog Timer prescaling when the Watchdog Timer is enabled. The different prescaling values and their corresponding Timeout Periods are shown in Table 6.

Table 6. Watchdog Timer Prescale Select (Typical Values at V_{CC} = 5.0V)

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

EEPROM Read/Write Access

The EEPROM access registers are accessible in the I/O space.

The write access time is in the range of 2.5 - 4ms, depending on the V_{CC} voltages. A self-timing function, however, lets the user software detect when the next byte can be written. An EEPROM brown-out detection prevents writing to the EEPROM if V_{CC} is below a certain level.

When the EEPROM is read or written, the CPU is halted for two clock cycles before the next instruction is executed.

THE EEPROM ADDRESS REGISTER - EEAR

Bit	7	6	5	4	3	2	1	0	
$1E	-	-	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	EEAR
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	

Bit 7,6 - Res : Reserved bits:

These bits are reserved bit in the AT90S1200 and will always read as zero.

Bits 5..0 - EEAR..0 : EEPROM Address:

The EEPROM Address Register - EEAR5..0 - specifies the EEPROM address in the 64-byte EEPROM space. The EEPROM data bytes are addressed linearly between 0 and 63.

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	

Bits 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	

Bits 7..2 - Res : Reserved bits:

These bits are reserved bits in the AT90S1200 and will always be 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 and 4ms at V_{CC} = 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. When EEWE has been set, the CPU is halted for two cycles before the next instruction is executed.

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 takes one instruction and there is no need to poll the EERE bit. When EERE has been set, the CPU is halted for two cycles before the next instruction is executed.

The Analog Comparator

The analog comparator compares the input values on the positive pin PB0 (AIN0) and the negative pin PB1 (AIN1). When the voltage on the positive pin PB0 (AIN0) is higher than the voltage on the negative pin PB1 (AIN1), the Analog Comparator Output, ACO is set (one). The comparator's output can be set to trigger the Analog Comparator interrupt. 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 22.

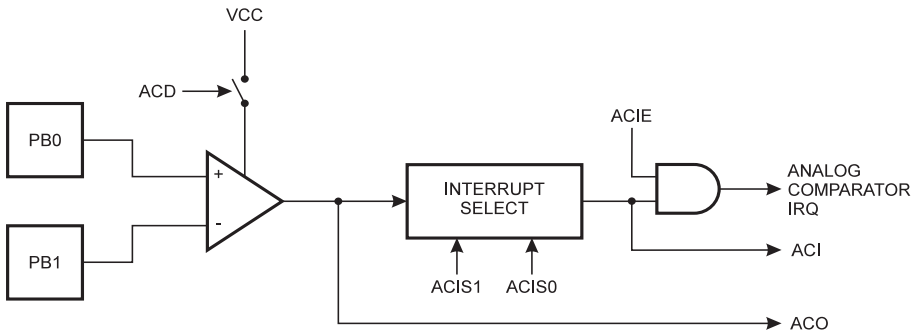


Figure 22. Analog Comparator Block Diagram

THE ANALOG COMPARATOR CONTROL AND STATUS REGISTER - ACSR

Bit	7	6	5	4	3	2	1	0	
$08	**ACD**	**-**	**ACO**	**ACI**	**ACIE**	**-**	**ACIS1**	**ACIS0**	**ACSR**
Read/Write	R/W	R	R	R/W	R/W	R	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 AT90S1200 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.

Bit 2 - Res : Reserved bit:

This bit is a reserved bit in the AT90S1200 and will always read as zero.

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 7.

Table 7. 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 (I_{IL}) if the internal pullups are activated.

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

Table 8. Port B Pins Alternate Functions

Port Pin	Alternate Functions
PB0	AIN0 (Analog comparator positive input)
PB1	AIN1 (Analog comparator negative input)
PB5	MOSI (Data input line for memory downloading)
PB6	MISO (Data output line for memory uploading)
PB7	SCK (Serial 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 PORTB 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 PIN 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) and the pin is configured as an input pin, the MOS pull up resistor is activated. To switch the pull up resistor off, PORTBn has to be cleared (zero) or the pin has to be configured as an output pin.

Table 9. DDBn Effect on PORTB Pins

DDBn	PORTBn	I/O	Pull up	Comment
0	0	Input	No	Tri-state (Hi-Z)
0	1	Input	Yes	PBn will source current (I_{IL}) 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 OF PORTB

The alternate pin functions of Port B are:

SCK - PORTB, Bit 7:

SCK, Clock input pin for Memory up/downloading.

MISO - PORTB, Bit 6:

MISO, Data output pin for Memory uploading.

MOSI - PORTB, Bit 5:

MOSI, Data input pin for Memory downloading.

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.

PORT B SCHEMATICS

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

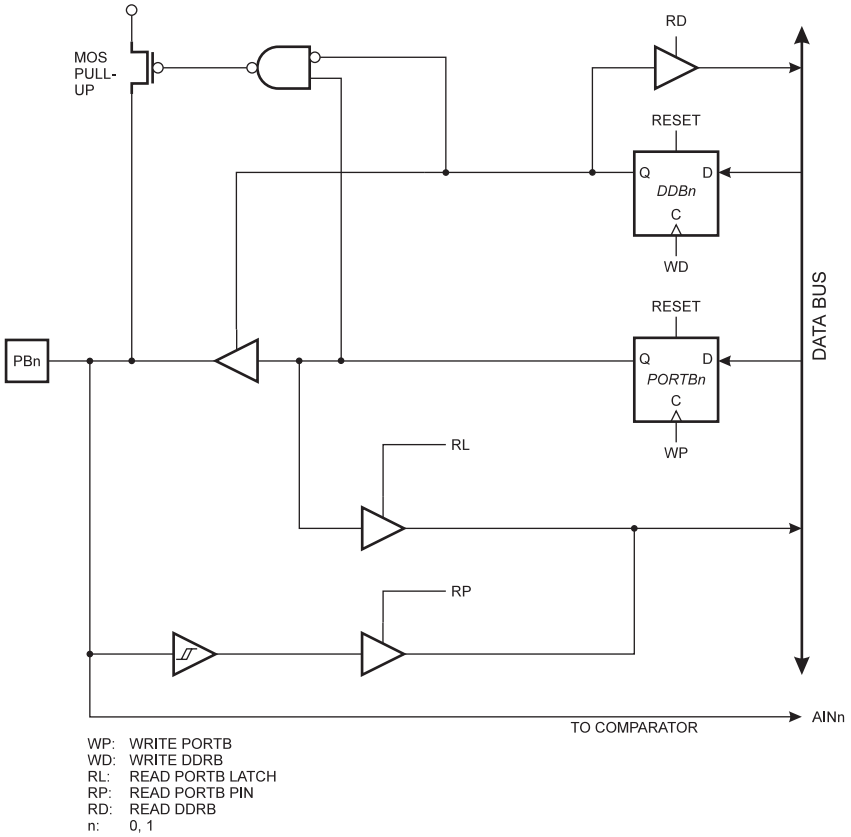


Figure 23. PORTB Schematic Diagram (pins PB0 and PB1)

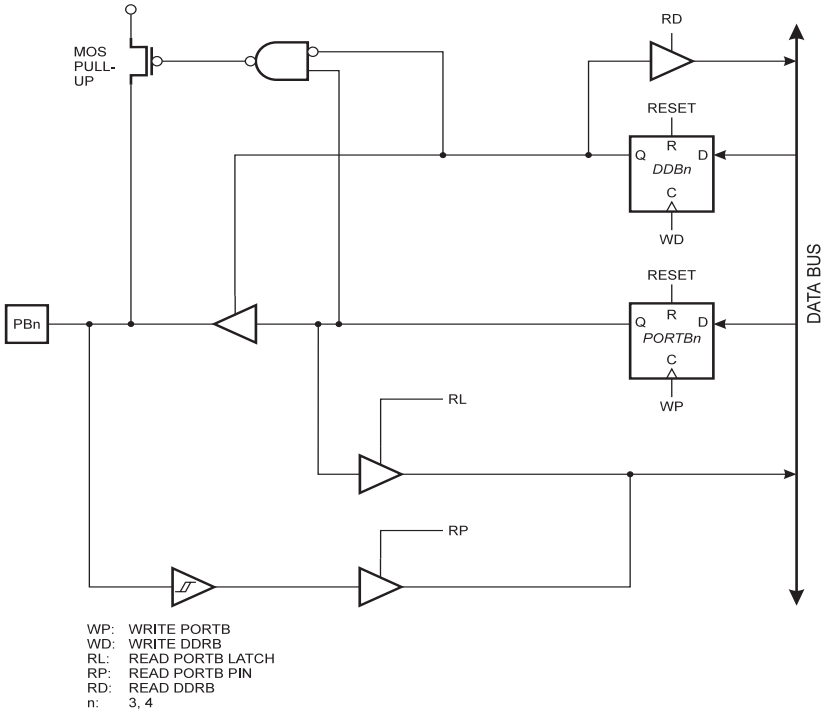


Figure 24. PORTB Schematic Diagram (Pins PB2, PB3 and PB4)

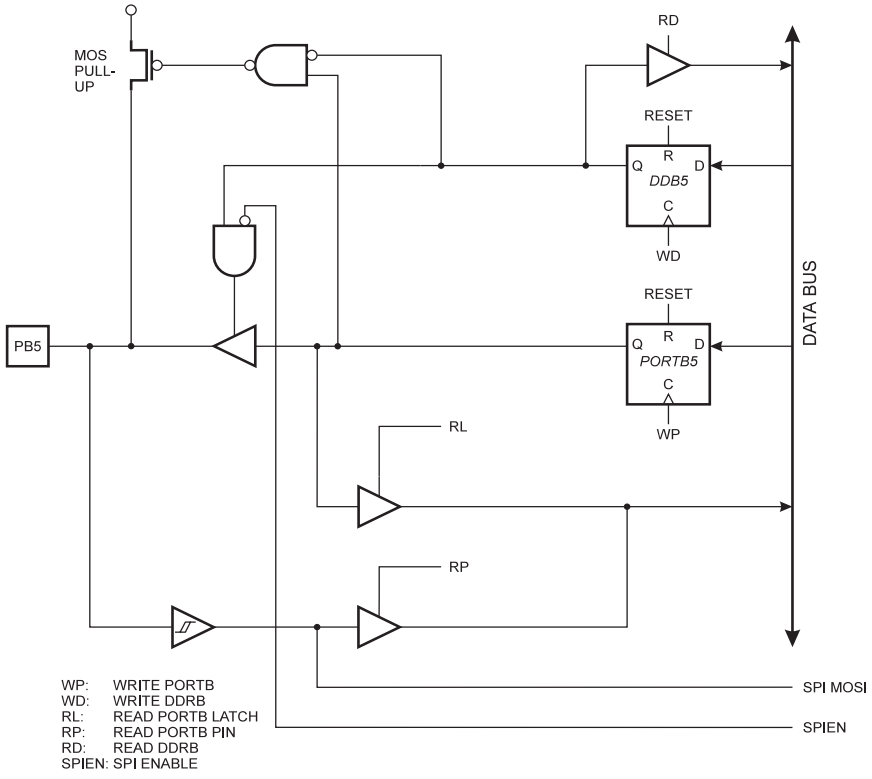


Figure 25. PORTB Schematic Diagram, Pin PB5

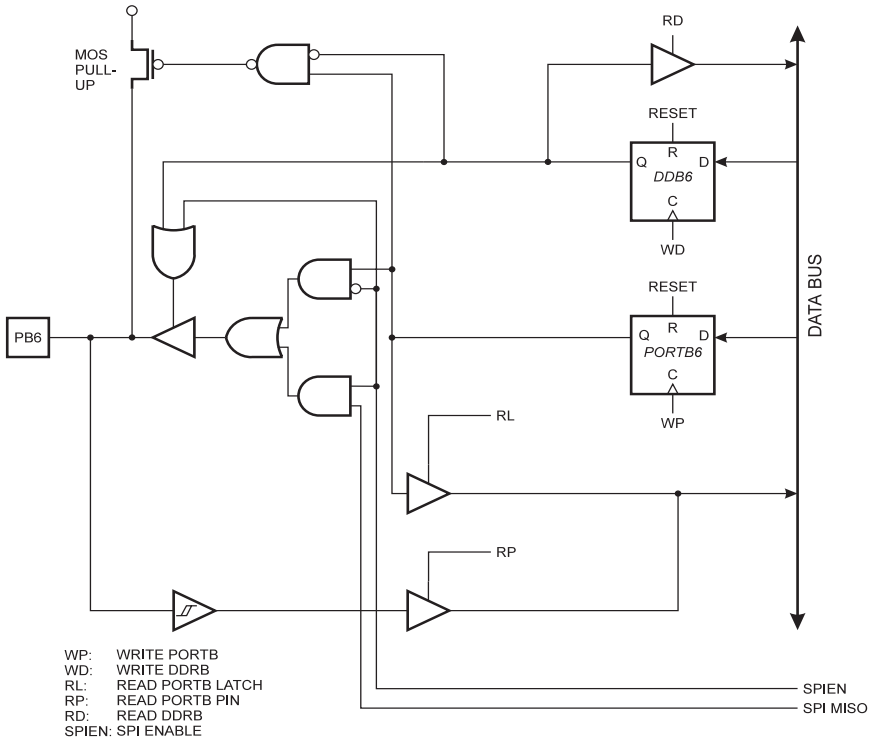


Figure 26. PORTB Schematic Diagram, Pin PB6

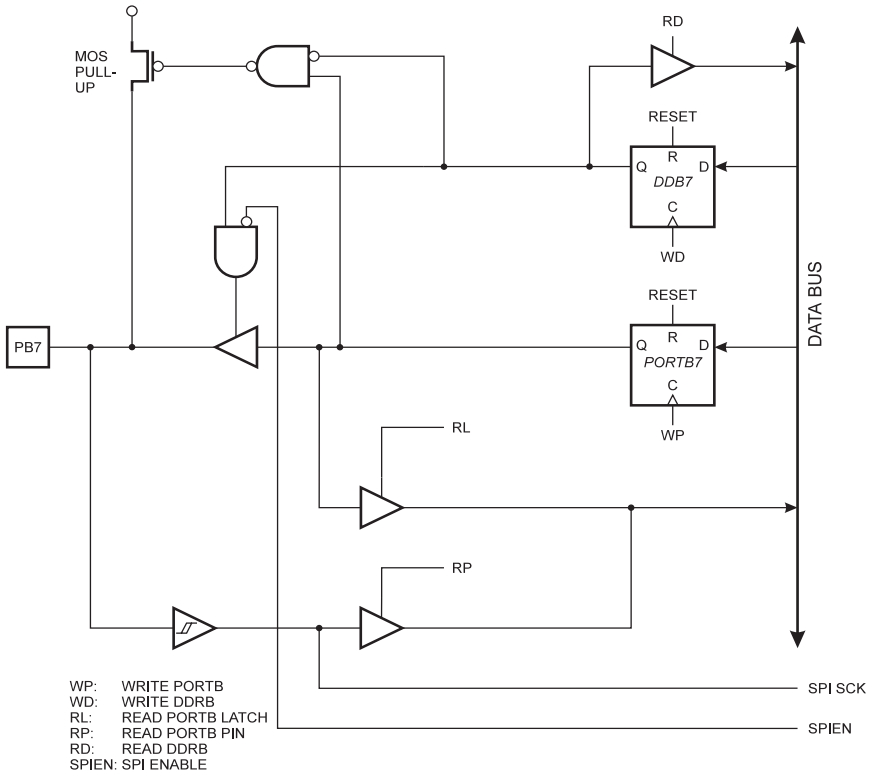


Figure 27. PORTB Schematic Diagram, Pin PB7

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 (I_{IL}) if the pullups are activated.

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

Table 10. Port D Pins Alternate Functions

Port Pin	Alternate Function
PD2	INT0 (External interrupt 0 input)
PD4	T0 (Timer/Counter 0 external input)

THE PORTD 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 - DDRD

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 - PIND

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 DDDn is configured as an input pin, the MOS pull up resistor is activated. To switch the pull up resistor off, the PORTDn bit has to be cleared (zero) or the pin has to be configured as an output pin.

Table 11. DDDn Bits Effect 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 (I_{IL}) 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:

T0 - PORTD, Bit 4:

T0, Timer/Counter0 clock source. See the Timer description for further details.

INT0 - PORTD, Bit 2:

INT0, External Interrupt source 0. See the interrupt description for further details.

PORTD SCHEMATICS

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

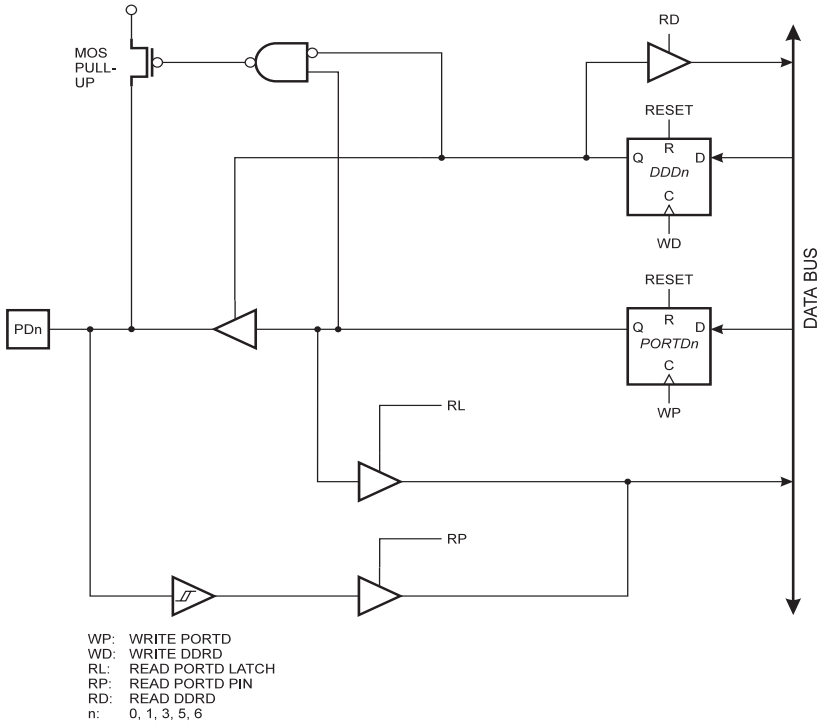


Figure 28. PORTD Schematic Diagram (Pins PD0, PD1, PD3, PD5 and PD6)

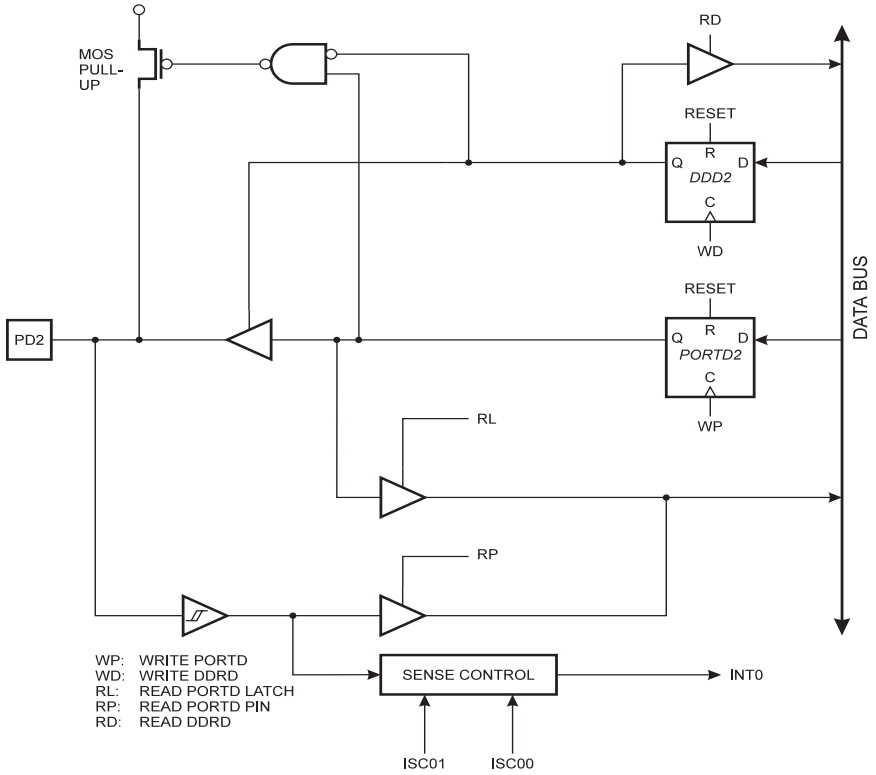


Figure 29. PORTD Schematic Diagram (Pin PD2)

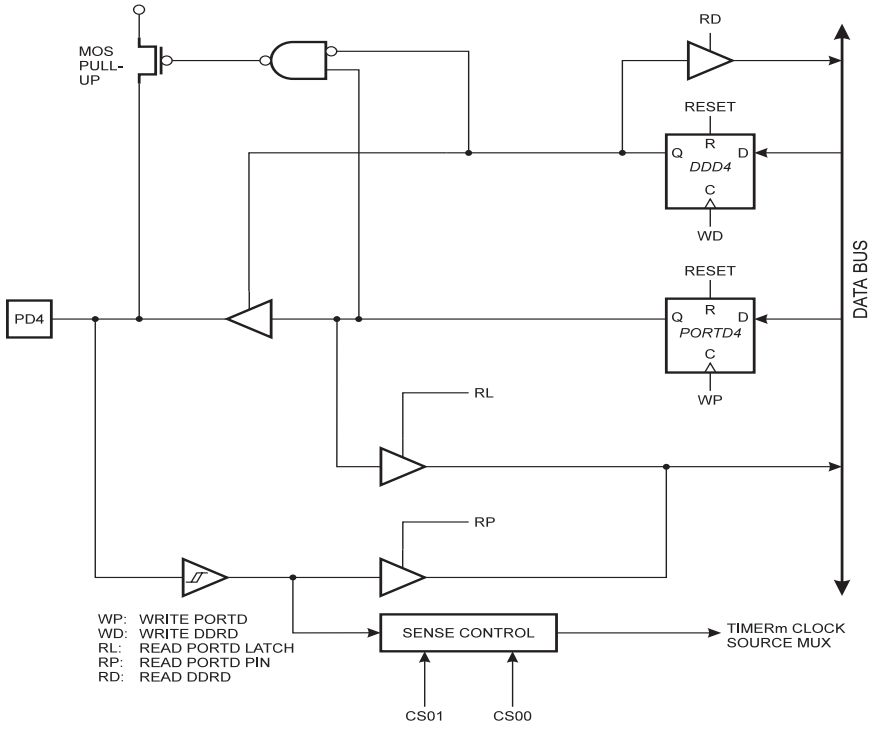


Figure 30. PORTD Schematic Diagram (Pin PD4)

Memory Programming

Program Memory Lock Bits

The AT90S1200 MCU provides two lock bits which can be left unprogrammed (‘1’) or can be programmed (‘0’) to obtain the additional features listed in Table 12.

Table 12. Lock Bit Protection Modes

Program Lock Bits			Protection Type
Mode	**LB1**	**LB2**	
1	1	1	No program lock features
2	0	1	Further programming of the Flash is disabled
3	0	0	Same as mode 2, but verify is also disabled.

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

Fuse Bits

The Fuse bits in the AT90S1200 are RCEN and SPIEN.

When RCEN is programmed (‘0’), MCU clocking from the internal RC oscillator is selected. Default value is erased (‘1’). Order AT90S1200A to get parts with this bit programmed.

When SPIEN is programmed (‘0’), Serial Programming Mode is enabled. Default value is programmed (‘0’).

These bits are not accessible in Serial Programming Mode and are not changed by a chip erase.

Device Code

All Atmel microcontrollers have a three-byte signature code which identifies the device. This code can be read in both serial and parallel mode. The three bytes reside in a separate address space, and for the AT90S1200 they are:

1. $000: $1E (indicates manufactured by Atmel)
2. $001: $90 (indicates 1 kB Flash memory)
3. $002: $01 (indicates 90S1200 device when $001 is $90)

Programming the Flash and EEPROM

Atmel’s AT90S1200 offers 1K bytes of in-system reprogrammable Flash Program memory and 64 bytes of EEPROM Data memory.

The AT90S1200 is normally shipped with the on-chip Flash Program memory and EEPROM Data memory arrays in the erased state (i.e. contents = $FF) and ready to be programmed. This device supports a High-Voltage (12V) Parallel programming mode and a Low-Voltage Serial programming mode. The +12V is used for programming enable only, and no current of significance is drawn by this pin. The serial programming mode provides a convenient way to download the Program and Data into the AT90S1200 inside the user’s system.

The Program and Data memory arrays on the AT90S1200 are programmed byte-by-byte in either programming modes. For the EEPROM, an auto-erase cycle is provided in the serial programming mode.

Parallel Programming

This section describes how to parallel program and verify Flash Program memory, EEPROM Data memory + Program Memory Lock bits and Fuse bits in the AT90S1200.

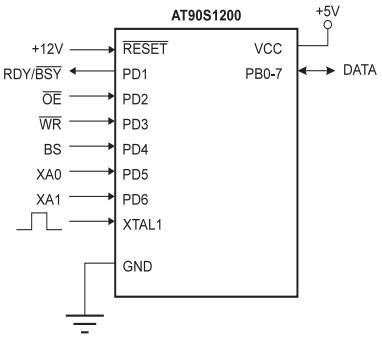


Figure 31. Parallel Programming

SIGNAL NAMES

In this section, some pins of the AT90S1200 are referenced by signal names describing their functionality during parallel programming rather than their pin names. Pins not described in the following table are referenced by pin names.

Table 13. Pin Name Mapping

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

The XA1/XA0 bits determine the action taken when the XTAL1 pin is given a positive pulse. The bit settings are shown in the following table:

Table 14. XA1 and XA0 Coding

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

When pulsing $\overline{WR}$ or $\overline{OE}$, the command loaded determines the action on input or output. The command is a byte where the different bits are assigned functions as shown in the following table:

Table 15. Command Byte Bit Coding

Bit#	Meaning when Set
7	Chip Erase
6	Write Fuse Bits. Located in the data byte at the following bit positions: D5: SPI Fuse, D0: RCEN Fuse (Note: write '0' to program, '1' to erase)
5	Write Lock Bits. Located in the data byte at the following bit positions: D2: LB2, D1: LB1 (Note: write '0' to program)
4	Write Flash or EEPROM (determined by bit 0)
3	Read signature row
2	Read Lock and Fuse Bits. Located in the data byte at the following bits positions: D7: LB1, D6: LB2, D5: SPI Fuse, D0: RCEN Fuse
1	Read from Flash or EEPROM (determined by bit 0)
0	0: Flash Access, 1: EEPROM Access

ENTER PROGRAMMING MODE

The following algorithm puts the device in parallel programming mode:

1. Apply 4.5 - 5.5V between VCC and GND.
2. Set the $\overline{RESET}$ and BS pin to '0' and wait at least 100 ns.
3. Apply 12V to $\overline{RESET}$ and wait at least 100ns before changing BS.

CHIP ERASE

The chip erase will erase the Flash and EEPROM memories plus Lock bits. The lock bits are not reset until the program memory has been completely erased. The Fuse bits are not changed. |A chip erase must be performed before the chip is programmed.

Load Command "Chip Erase"

1. Set XA1, XA0 to '10'. This enables command loading.
2. Set BS to '0'.
3. Set PB(7:0) to '1000 0000'. This is the command for Chip erase.
4. Give XTAL1 a positive pulse. This loads the command, and starts the erase of the Flash and EEPROM arrays. After pulsing XTAL1, give $\overline{WR}$ a negative pulse to enable lock bit erase at the end of the erase cycle. Then wait at least 10 ms for the chip erase cycle to finish. Chip erase does not generate any activity on the RDY/$\overline{BSY}$ signal.

PROGRAMMING THE FLASH

Load Command "Program Flash"

1. Set XA1, XA0 to '10'. This enables command loading.
2. Set BS to '0'
3. Set PB(7:0) to '0001 0000'. This is the command for Flash programming.
4. Give XTAL1 a positive pulse. This loads the command.

Load Address Low byte

1. Set XA1, XA0 to '00'. This enables address loading.
2. Set BS to '0'. This selects Low address.
3. Set PB(7:0) = Address Low byte ($00 - $FF)
4. Give XTAL1 a positive pulse. This loads the Address Low byte.

Load Address High byte

1. Set XA1, XA0 to ‘00’. This enables address loading.
2. Set BS to ‘1’. This selects High address.
3. Set PB(7:0) = Address High byte ($00 - $01)
4. Give XTAL1 a positive pulse. This loads the Address High byte.

Load Data byte

1. Set XA1, XA0 to ‘01’. This enables data loading.
2. Set PB(7:0) = Data Low byte ($00 - $FF)
3. Give XTAL1 a positive pulse. This loads the Data Low byte.

Write Data Low byte

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

Load Data byte

1. Set XA1, XA0 to ‘01’. This enables data loading.
2. Set PB(7:0) = Data High byte ($00 - $FF)
3. Give XTAL1 a positive pulse. This loads the Data High byte.

Write Data High byte

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

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

- The command for Flash programming needs only be loaded before programming of the first byte.
- Address High byte needs only be loaded before programming a new 256 word page in the Flash.

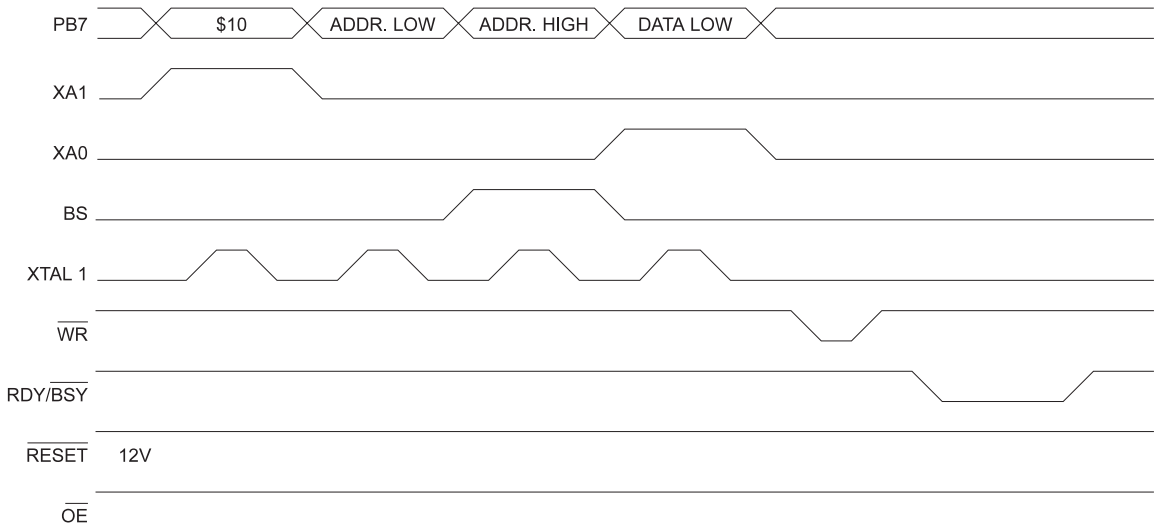


Figure 32. Programming Flash Low Byte

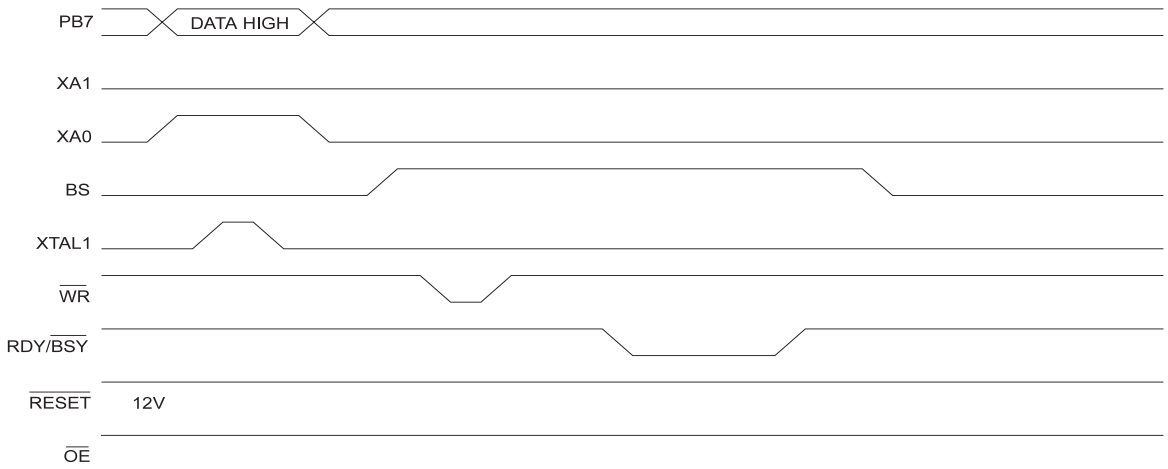


Figure 33. Programming Flash High Byte

PROGRAMMING THE EEPROM

The programming algorithm for the EEPROM data memory is as follows (refer to Flash Programming for details on Command, Address and Data loading):

1. Load Command ‘0001 0001’.
2. Load Low EEPROM Address ($00 - $3F)
3. Load Low EEPROM Data ($00 - $FF)
4. Give $\overline{WR}$ a negative pulse and wait for RDY/$\overline{BSY}$ to go high.

The Command needs only be loaded before programming the first byte.

READING THE FLASH

The algorithm for reading the Flash memory is as follows (refer to Flash Programming for details on Command, Address and Data loading):

1. Load Command ‘0000 0010’.
2. Load Low Address ($00 - $FF)
3. Load High Address ($00 - $01)
4. Set $\overline{OE}$ to ‘0’, and BS to ‘0’. The Low Data byte can now be read at PB(7:0)
5. Set BS to ‘1’. The High Data byte can now be read from PB(7:0)
6. Set $\overline{OE}$ to ‘1’.

The Command needs only be loaded before reading the first byte.

READING THE EEPROM

The algorithm for reading the EEPROM memory is as follows (refer to Flash Programming for details on Command, Address and Data loading):

1. Load Command ‘0000 0011’.
2. Load Low Address ($00 - $3F)
3. Set $\overline{OE}$ to ‘0’, and BS to ‘0’. The EEPROM Data byte can now be read at PB(7:0)
4. Set $\overline{OE}$ to ‘1’.

The Command needs only be loaded before reading the first byte.

PROGRAMMING THE FUSE BITS

The algorithm for programming the Fuse bits is as follows (refer to Flash Programming for details on Command, Address and Data loading):

1. Load Command ‘0100 0000’.
2. Load Data.

   Bit 5 = ‘0’ programs the SPI Fuse bit. Bit 5 = ‘1’ erases the SPI Fuse bit.

   Bit 0 = ‘0’ programs the RCEN Fuse bit. Bit 0 = ‘1’ erases the RCEN Fuse bit.
3. Give $\overline{WR}$ a negative pulse and wait for RDY/$\overline{BSY}$ to go high.

IMPORTANT! $\overline{WR}$ *must be held low for at least 1 ms.*

PROGRAMMING THE LOCK BITS

The algorithm for programming the Lock bits is as follows (refer to Flash Programming for details on Command, Address and Data loading):

1. Load Command ‘0010 0000’.
2. Load Data.

   Bit 2 = ’0’ programs Lock Bit2

   Bit 1 = ’0’ programs Lock Bit1
3. Give $\overline{WR}$ a negative pulse and wait for RDY/$\overline{BSY}$ to go high.

The lock bits can only be cleared by executing a chip erase.

READING THE FUSE AND LOCK BITS

The algorithm for reading the Fuse and Lock bits is as follows (refer to Flash Programming for details on Command, Address and Data loading):

1. Load Command ‘0000 0100’.
2. Set $\overline{OE}$ to ‘0’, and BS to ‘1’. The Status of Fuses an Lock bits can now be read at PB(7:0)

   Bit 7: Lock Bit1 (‘0’ means programmed)

   Bit 6: Lock Bit2 (‘0’ means programmed)

   Bit 5: SPI Fuse (‘0’ means programmed)

   Bit 0: RCEN Fuse (‘0’ means programmed)
3. Set $\overline{OE}$ to ‘1’.

Observe especially that BS needs to be set to ‘1’.

READING THE SIGNATURE BYTES

The algorithm for reading the Signature bytes is as follows (refer to Flash Programming for details on Command, Address and Data loading):

1. Load Command ‘0000 1000’.
2. Load Low address ($00 - $02)

   Set $\overline{OE}$ to ‘0’, and BS to ‘0’. The Selected Signature byte can now be read at PB(7:0)
3. Set $\overline{OE}$ to ‘1’.

The command needs only be programmed before reading the first byte.

Serial Downloading

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

When programming the EEPROM, 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 $3FF for Program Flash memory and $000 to $03F 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 minimum low and high periods for the serial clock (SCK) input are defined as follows:

Low: > 1 XTAL1 clock cycle

High: > 4 XTAL1 clock cycles

SERIAL PROGRAMMING ALGORITHM

To program and verify the AT90S1200 in the serial programming mode, the following sequence is recommended (See four byte instruction formats in Table 16):

1. *Power-up sequence*:

   Apply power between VCC and GND while $\overline{RESET}$ and SCK are set to ‘0’. (If the programmer can not guarantee that SCK is held low during power-up, $\overline{RESET}$ must be given a positive pulse after SCK has been set to ‘0’.) If a crystal is not connected across pins XTAL1 and XTAL2, apply a 0 to 16 MHz clock to the XTAL1 pin.
2. Wait for at least 20 ms and enable serial programming by sending the Programming Enable serial instruction to pin MOSI/PB5. Refer to the above section for minimum low and high periods for the serial clock input, SCK.
3. If a chip erase is performed (must be done to erase the Flash), wait 10ms, give $\overline{RESET}$ a positive pulse and start over again from Step 2.
4. The Flash or EEPROM array is programmed one byte at a time by supplying the address and data together with the appropriate Write instruction. An EEPROM memory location is first automatically erased before new data is written. The next byte can be written after 4 ms.

5. Any memory location can be verified by using the Read instruction which returns the content at the selected address at serial output MISO/PB6.

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 '0' (if a crystal is not used).

   Set $\overline{\text{RESET}}$ to '1'.

   Turn V_{CC} power off.

Table 16. Serial Programming Instruction Set AT90S1200

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 1K & 64 byte memory arrays
Read Program Memory	0010 **H**000	0000 000**a**	**bbbb bbbb**	**oooo oooo**	Read **H**(high or low) data **o** from Program memory at word address **a:b**
Write Program Memory	0100 **H**000	0000 000**a**	**bbbb bbbb**	**iiii iiii**	Write **H**(high or low) data **i** to Program memory at word address **a:b**
Read EEPROM Memory	1010 0000	0000 0000	xx**bb bbbb**	**oooo oooo**	Read data **o** from EEPROM memory at address **b**
Write EEPROM Memory	1100 0000	0000 0000	xx**bb bbbb**	**iiii iiii**	Write data **i** to EEPROM memory at address **b**
Write Lock Bits	1010 1100	111x x***21***x	xxxx xxxx	xxxx xxxx	Write lock bits. Set bits ***1,2***='0' to program lock bits.
Read Device Code	0011 0000	xxxx xxxx	xxxx xx**bb**	**oooo oooo**	Read Device Code **o** from address **b**.

Notes: **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

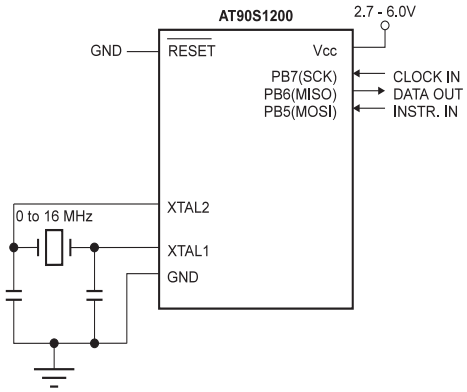


Figure 34. Programming and Verify

When *writing* serial data to the AT90S1200, data is clocked on the *rising* edge of CLK.

When *reading* data from the AT90S1200, data is clocked on the *falling* edge of CLK. See Figure 35 for an explanation.

Programming Characteristics

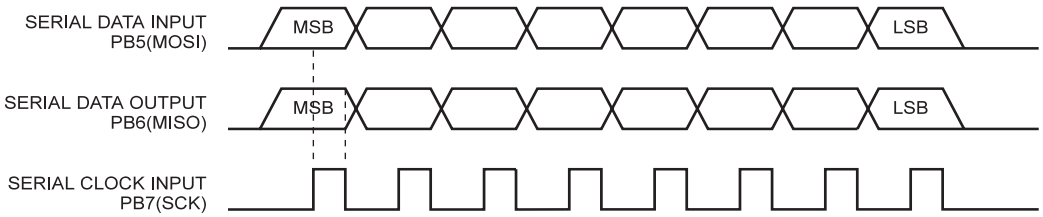


Figure 35. Serial Downloading Waveforms

Absolute Maximum Ratings

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on any Pin except $\overline{\text{RESET}}$ with respect to Ground	-1.0V to +7.0V
Maximum Operating Voltage	6.6V
DC Current per I/O Pin	40.0 mA
DC Current VCC and GND Pins	140.0 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.

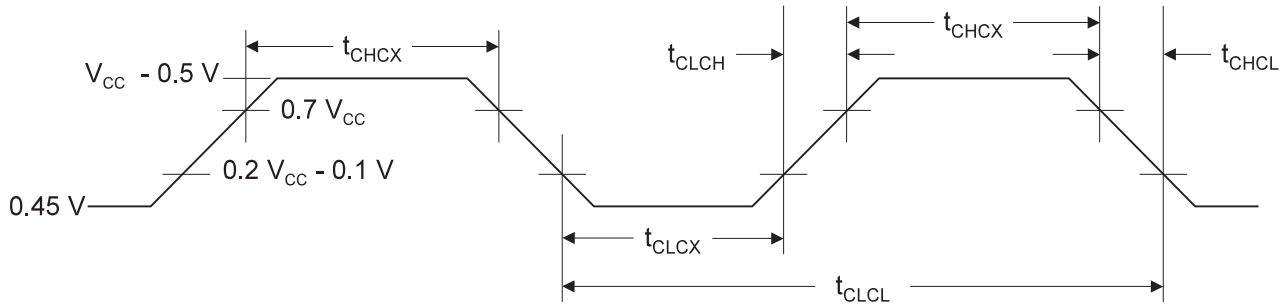
DC Characteristics

T_A = -40°C to 85°C, V_{CC} = 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, $\overline{RESET}$)	0.2 V_{CC} + 0.9		V_{CC} + 0.5	V
V_{IH1}	Input High Voltage	(XTAL1, $\overline{RESET}$)	0.7 V_{CC}		V_{CC} + 0.5	V
V_{OL}	Output Low Voltage (1) (Ports B, D)	I_{IL} = 25 mA, V_{CC} = 5V I_{IL} = 15 mA, V_{CC} = 3V			0.5	V
V_{OH}	Output High Voltage (Ports B,D)	I_{OH} = 3 mA, V_{CC} = 5V I_{OH} = 3 mA, V_{CC} = 3V	V_{CC} - 0.5			V
I_{OH}	Output Source Current (Ports B,D)	V_{CC} = 5V, V_{OH} = 4.5V V_{CC} = 3V, V_{OH} = 2.7V		4 2		mA
I_{IL}	Output Sink Current (Port B,D)	V_{CC} = 5V, V_{OL} = 0.5V V_{CC} = 3V, V_{OL} = 0.3V		28 11		mA
RRST	Reset Pull-Up Resistor		100		500	kΩ
$R_{I/O}$	I/O Pin Pull-Up Resistor		35		120	kΩ
I_{CC}	Power Supply Current	Active Mode, 3V, 4MHz		2		mA
		Idle Mode 3V, 4MHz		500		µA
		Power Down (2) WDT enabled, 3V		10	15 (3)	µA
		Power Down (2) WDT disabled, 3V		0.15	1(3)	µA
V_{ACIO}	Analog Comparator Input Offset Voltage	V_{CC} = 5V			20	mV
I_{ACLK}	Analog Comparator Input Leakage Current	V_{IN} = 1V		10		nA
t_{ACPD}	Analog Comparator Propagation Delay	V_{CC} = 2.7V V_{CC} = 4.0V		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 V_{CC} for Power Down is 2V.
3. Value tested to 45°C

External Clock Drive Waveforms

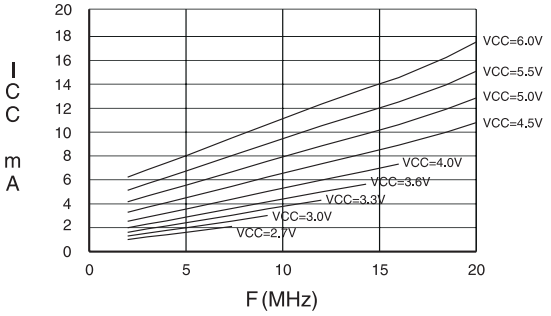


External Clock Drive

Symbol	Parameter	V_{CC} = 2.7V to 6.0V		V_{CC} = 4.0V to 6.0V		Units
		Min	Max	Min	Max	
$1/t_{CLCL}$	Oscillator Frequency	0	4	0	16	MHz
t_{CLCL}	Clock Period	250		62.5		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

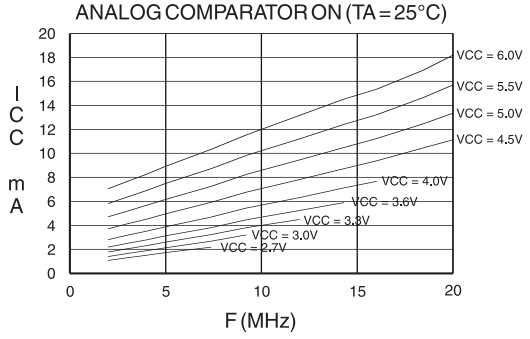
ACTIVE SUPPLY CURRENT

ANALOG COMPARATOR OFF (TA = 25°C)



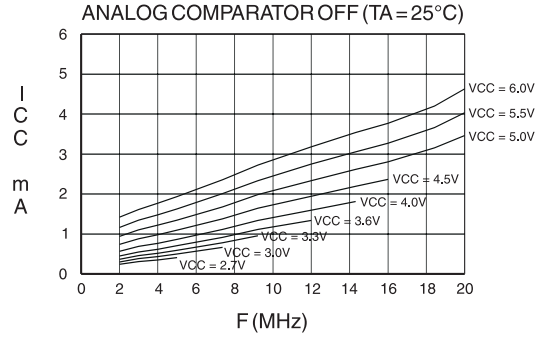
ACTIVE SUPPLY CURRENT

ANALOG COMPARATOR ON (TA = 25°C)



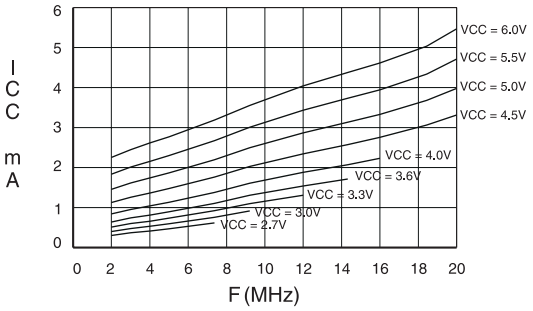
IDLE SUPPLY CURRENT

ANALOG COMPARATOR OFF (TA = 25°C)



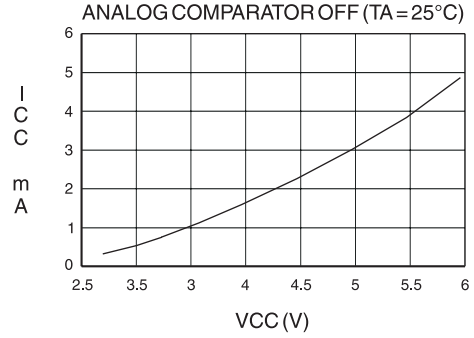
IDLE SUPPLY CURRENT

ANALOG COMPARATOR ON (TA = 25°C)



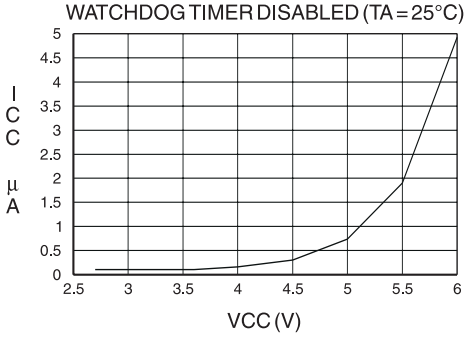
ACTIVE CURRENT RC-OSC

ANALOG COMPARATOR OFF (TA = 25°C)



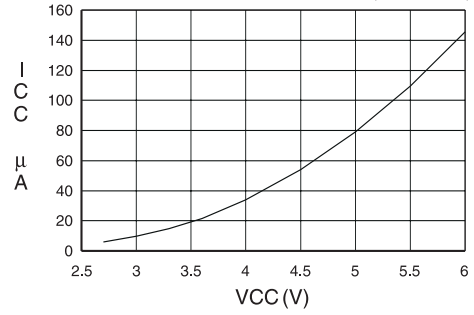
POWER DOWN SUPPLY CURRENT

WATCHDOG TIMER DISABLED (TA = 25°C)



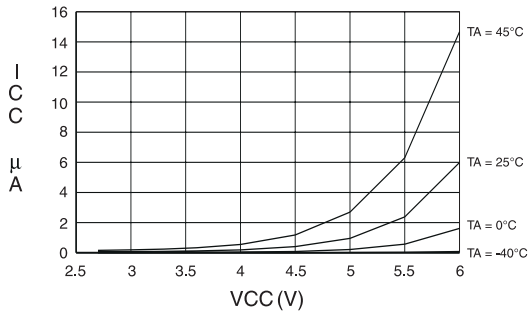
POWER DOWN SUPPLY CURRENT

WATCHDOG TIMER ENABLED (TA = 25°C)

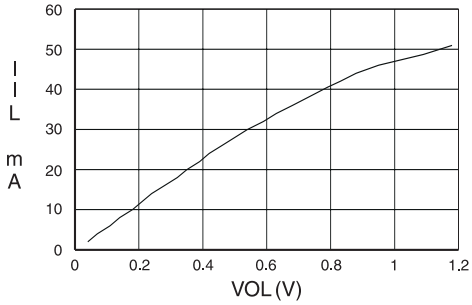


POWER DOWN SUPPLY CURRENT

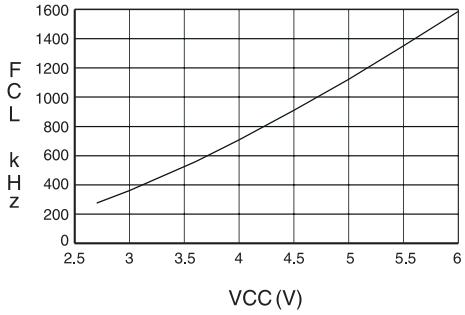
WATCHDOG TIMER DISABLED



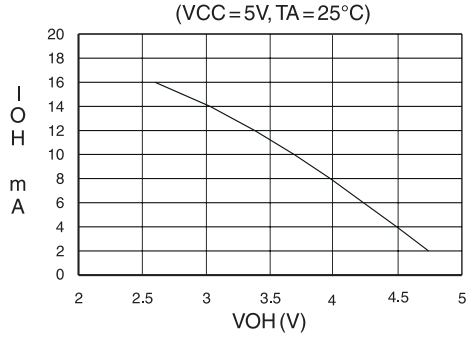
INPUT SINK CURRENT
(VCC = 5V, TA = 25°C)



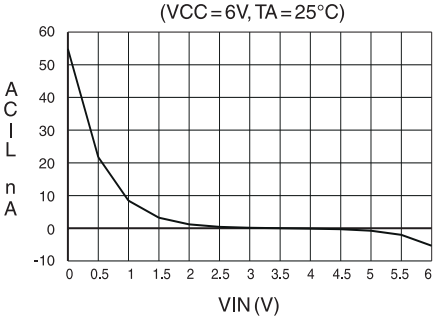
RC OSCILLATOR FREQUENCY
(TA = 25°C)



OUTPUT SOURCE CURRENT
(VCC = 5V, TA = 25°C)



ANALOG COMPARATOR
INPUT LEAKAGE CURRENT
(VCC = 6V, TA = 25°C)



Note: Charts show typical values.

Ordering Information

Speed (MHz)	Power Supply	Ordering Code*	Package	Operation Range
4	2.7 - 6.0V	AT90S1200-4PC AT90S1200-4SC AT90S1200-4YC	20P3 20S 20Y	Commercial (0°C to 70°C)
		AT90S1200-4PI AT90S1200-4SI AT90S1200-4YI	20P3 20S 20Y	Industrial (-40°C to 85°C)
16	4.0 - 6.0V	AT90S1200-16PC AT90S1200-16SC AT90S1200-16YC	20P3 20S 20Y	Commercial (0°C to 70°C)
		AT90S1200-16PI AT90S1200-16SI AT90S1200-16YI	20P3 20S 20Y	Industrial (-40°C to 85°C)

* Order AT90S1200A-XXX for devices with the RCEN fuse programmed.

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)
20Y	20 Lead, 0.150” Wide SSOP

AT90S1200 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	2-12
$3E	Reserved									
$3D	Reserved									
$3C	Reserved									
$3B	GIMSK	-	INT0	-	-	-	-	-	-	2-17
$3A	Reserved									
$39	TIMSK	-	-	-	-	-	-	TOIE0	-	2-18
$38	TIFR	-	-	-	-	-	-	TOV0	-	2-18
$37	Reserved									
$36	Reserved									
$35	MCUCR	-	-	SE	SM	-	-	ISC01	ISC00	2-19
$34	Reserved									
$33	TCCR0	-	-	-	-	-	CS02	CS01	CS00	2-21
$32	TCNT0	Timer/Counter0 (8 Bit)								2-22
$31	Reserved									
$30	Reserved									
$2F	Reserved									
$2E	Reserved									
$2D	Reserved									
$2C	Reserved									
$2B	Reserved									
$2A	Reserved									
$29	Reserved									
$28	Reserved									
$27	Reserved									
$26	Reserved									
$25	Reserved									
$24	Reserved									
$23	Reserved									
$22	Reserved									
$21	WDTCR	-	-	-	-	WDE	WDP2	WDP1	WDP0	2-23
$20	Reserved									
$1F	Reserved									
$1E	EEAR	-	EEPROM Address Register							2-24
$1D	EEDR	EEPROM Data Register								2-24
$1C	EECR	-	-	-	-	-	-	EEWE	EERE	2-24
$1B	Reserved									
$1A	Reserved									
$19	Reserved									
$18	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	2-26
$17	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	2-27
$16	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	2-27
$15	Reserved									
$14	Reserved									
$13	Reserved									
$12	PORTD	-	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	2-31
$11	DDRD	-	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	2-31
$10	PIND	-	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	2-31
$0F	Reserved									
$0E	Reserved									
$0D	Reserved									
$0C	Reserved									
$0B	Reserved									
$0A	Reserved									
$09	Reserved									
$08	ACSR	ACD	-	ACO	ACI	ACIE	-	ACIS1	ACIS0	2-25
…	Reserved									
$00	Reserved									

AT90S1200 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
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
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 • (FFh - 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
RCALL	k	Relative Subroutine Call	PC ← PC + k + 1	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	P, b	Skip if Bit in I/O Register Cleared	if (P(b)=0) PC ← PC + 2 or 3	None	1 / 2
SBIS	P, b	Skip if Bit in I/O Register is Set	if (P(b)=1) PC ← PC + 2 or 3	None	1 / 2
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)

AT90S1200 Instruction Set Summary (Continued)

Mnemonics	Operands	Description	Operation	Flags	#Clocks
DATA TRANSFER INSTRUCTIONS					
LD	Rd,Z	Load Register Indirect	Rd ← (Z)	None	2
ST	Z,Rr	Store Register Indirect	(Z) ← Rr	None	2
MOV	Rd, Rr	Move Between Registers	Rd ← Rr	None	1
LDI	Rd, K	Load Immediate	Rd ← K	None	1
IN	Rd, P	In Port	Rd ← P	None	1
OUT	P, Rr	Out Port	P ← Rr	None	1
BIT AND BIT-TEST INSTRUCTIONS					
SBI	P,b	Set Bit in I/O Register	I/O(P,b) ← 1	None	2
CBI	P,b	Clear Bit in I/O Register	I/O(P,b) ← 0	None	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	3
WDR		Watch Dog Reset	(see specific descr. for WDR/timer)	None	1