AT90 Microcontroller

Reference for Register Information

The purpose of this booklet is to provide convenient access to register information relating to the Atmel AT90 microcontroller.

This edition is customized for use with the AUT microcontroller laboratory equipment. Where the AUT hardware setup does not allow a particular register enabled feature the text relating to the feature has been printed in gray. The register bits corresponding with the gray text sections should be cleared to zero when the register is initialized.

INDEX

Parallel Port Register Addresses

The Mega128 microcontroller has six 8-bit parallel ports (A, B, C, D, E, F)

All of the parallel ports operate in the same way.
In the information below “n” indicates which port (A to F) is referred to

Port Data Register – PORTn

Bit	7	6	5	4	3	2	1	0	
	PORTA7	**PORTA6**	**PORTA5**	**PORTA4**	**PORTA3**	**PORTA2**	**PORTA1**	**PORTA0**	**PORTA**
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

What appears in the data register will appear on the output pin if the pin is configured for output

Port Data Direction Register – DDRn

Bit	7	6	5	4	3	2	1	0	
	DDA7	**DDA6**	**DDA5**	**DDA4**	**DDA3**	**DDA2**	**DDA1**	**DDA0**	**DDRA**
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

Writing a one to the data direction register sets the corresponding bit in the port to output, writing a zero sets the bit for input.

Port Input Pins Address – PINn

Bit	7	6	5	4	3	2	1	0	
	PINA7	**PINA6**	**PINA5**	**PINA4**	**PINA3**	**PINA2**	**PINA1**	**PINA0**	**PINA**
Read/Write	R	R	R	R	R	R	R	R	
Initial value	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	

Reading the Input PIN address will show what is on the pin, regardless of whether configured for input or output.

Bit Addressing of Parallel Ports

Individual bits of ports A to E can be addressed using the bit addressing form.

Eg PORTA_Bit2 refers to bit 2 of the **output data** register of port A
DDRB_Bit4 refers to bit 4 of the **data direction** register of port B
PINE_Bit7 refers to bit 7 of the **input register** of port E

The registers for port F does not support bit addressing, To address any of the registers for Port F bit masking method must be used.

Eg To set the output from bit 5 of Port F leaving the other bits unchanged the form is

```
PORTF = PORTF | 0b00100000;
```

To clear bit 2 of the data direction register for port F leaving the other bits unchanged the form is

```
DDRF = DDRF & 0b11111011;
```

Analog to Digital Converter Registers

ADC Data Register – ADCL and ADCH

ADLAR = 0: (right adjusted)

Bit	15	14	13	12	11	10	9	8	
	–	**–**	**–**	**–**	**–**	**–**	**ADC9**	**ADC8**	**ADCH**
	ADC7	**ADC6**	**ADC5**	**ADC4**	**ADC3**	**ADC2**	**ADC1**	**ADC0**	**ADCL**
	7	6	5	4	3	2	1	0	

ADLAR = 1: (left adjusted)

Bit	15	14	13	12	11	10	9	8	
	ADC9	**ADC8**	**ADC7**	**ADC6**	**ADC5**	**ADC4**	**ADC3**	**ADC2**	**ADCH**
	ADC1	**ADC0**	**–**	**–**	**–**	**–**	**–**	**–**	**ADCL**
	7	6	5	4	3	2	1	0	

When an ADC conversion is complete, the result is found in these two registers. The two registers can be accessed by using ADCH and ADCL register names, or if wanting to access them both use the ADC register name. If the result is left adjusted (see the ADLAR bit in the ADMUX register) and no more than 8-bit precision (7-bit + sign bit for differential input channels) is required, it is sufficient to read ADCH. (Note reading just the ADCL will prevent the ADC Data Register from being updated).

ADC Control and Status Register B – ADCSRB

Bit	7	6	5	4	3	2	1	0	
	ADHSM	ACME	-	-	-	ADTS2	ADTS1	ADTS0	ACSRB
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 – ADHSM: ADC High Speed Mode

Writing this bit to one enables the ADC High Speed mode. This mode enables higher conversion rate at the expense of higher power consumption.

• Bit 2:0 – ADTS2:0: ADC Auto Trigger Source

If ADATE in ADCSRA is written to one, the value of these bits selects which source will trigger an ADC conversion. If ADATE is cleared, the ADTS2:0 settings will have no effect

ADTS2	ADTS1	ADTS0	Trigger Source
0	0	0	Free running mode
0	0	1	Analogue Comparator
0	1	0	External interrupt 0
0	1	1	Timer 0 compare match
1	0	0	Timer 0 overflow
1	0	1	Timer 1 compare match
1	1	0	Timer 1 overflow
1	1	1	Timer 1 input capture

ADC Control and Status Register A – ADCSRA

Bit	7	6	5	4	3	2	1	0	
	ADEN	**ADSC**	**ADATE**	**ADIF**	**ADIE**	**ADPS2**	**ADPS1**	**ADPS0**	**ADCSRA $06 ($26)**
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 ADEN: ADC Enable

Setting this bit enables the ADC, clearing the bit turns the ADC off.

Bit 6 ADSC ADC Start Conversion

Set this bit to start each conversion. When the conversion is complete, the bit returns to zero. Note in Free Running Mode, set this bit to start the first conversion.

Bit 5 ADATE ADC Auto Trigger Enable

When this bit is zero, the ADC is in single conversion mode, when it is one, auto triggering is enables.

Bit 4 ADIF ADC Interrupt Flag

This bit is set when an ADC conversion is completed. The bit is cleared by hardware when the interrupt routine is executed. Alternatively, writing a logical one to the flag can clear the bit.

Bit 3 ADIE ADC Interrupt Enable

When this bit is set and the I-bit in SREG is set, the ADC Conversion Complete Interrupt is activated.

Bits 2:0 ADPS2: 0 ADC Prescaler Select Bits

These bits determine the conversion speed of the ADC related to the system clock.

ADPS2	ADPS1	ADPS0	Division Factor
0	0	0	2
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

ADC Multiplexer Selection Register – ADMUX

Bit	7	6	5	4	3	2	1	0		
	REFS1	**REFS0**	**ADLAR**	**MUX4**	**MUX3**	**MUX2**	**MUX1**	**MUX0**	**ADMUX**	**$07 ($27)**
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 7:6 (REFS1: 0) Reference Selection Bits

These are used to alter the voltage reference selection of the ADC.

REFS1	REFS0	Voltage Reference Selection
0	0	AREF, Internal Vref turned off
0	1	AVCC with external capacitor at AREF pin
1	0	Reserved
1	1	Internal 2.56V Voltage Reference with external capacitor at AREF pin

Bit 5 ADLAR ADC Left Adjust Result

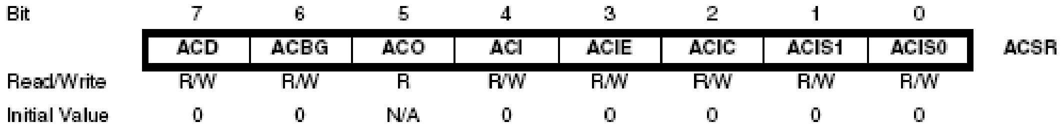
Writing a one to this bit will left adjust the ADC conversion result in the ADC data register, other wise the result is right adjusted.

Bits 4:0 MUX4: 0 Analog Channel and Gain Selection Bits

This selects which channel will be converted, and whether they are single or differential and which gain setting is used.

	Single Ended	Positive Differential	Negative Differential	
MUX4.0	**Input**	**Input**	**Input**	**Gain**
00000	ADC0	N/A		
00001	ADC1			
00010	ADC2			
00011	ADC3			
00100	ADC4			
00101	ADC5			
00110	ADC6			
00111	ADC7			
01000	N/A	ADC0	ADC0	10x
01001		ADC1	ADC0	10x
01010		ADC0	ADC0	200X
01011		ADC1	ADC0	200x
01100		ADC2	ADC2	10x
01101		ADC3	ADC2	10x
01110		ADC2	ADC2	200x
01111		ADC3	ADC2	200x
10000		ADC0	ADC1	1x
10001		ADC1	ADC1	1x
10010		ADC2	ADC1	1x
10011		ADC3	ADC1	1x
10100		ADC4	ADC1	1x
10101		ADC5	ADC1	1x
10110		ADC6	ADC1	1x
10111		ADC7	ADC1	1x
11000		ADC0	ADC2	1x
11001		ADC1	ADC2	1x
11010		ADC2	ADC2	1x
11011		ADC3	ADC2	1x
11100		ADC4	ADC2	1x
11101		ADC5	ADC2	1x
11110	1.22 V (V_{BG})	N/A		
11111	0 V (GND)			

The Analog Comparator   Control Register Functions



Bit	7	6	5	4	3	2	1	0	
	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	ACSR
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	N/A	0	0	0	0	0	

Bit 7 – ACD: Analog Comparator Disable
When this bit is written logic one, the power to the Analog Comparator is switched off.

• Bit 6 – ACBG: Analog Comparator Bandgap Select
When this bit is set, a fixed bandgap reference voltage replaces the positive input to the Analog Comparator. When this bit is cleared, AIN0 is applied to the positive input of the Analog Comparator.

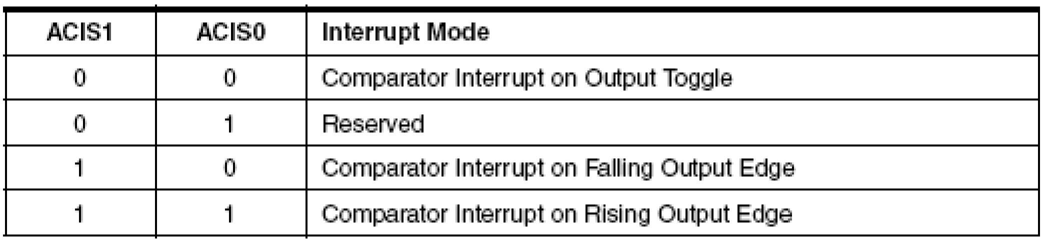
• Bit 5 – ACO: Analog Comparator Output
The output of the Analog Comparator is connected to ACO.

• Bit 4 – ACI: Analog Comparator Interrupt Flag
This bit is set by hardware when a comparator output event triggers the interrupt mode defined by ACIS1 and ACIS0. 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 written logic one and the I-bit in the Status Register is set, the Analog Comparator interrupt is activated. When written logic zero, the interrupt is disabled.

• Bit 2 – ACIC: Analog Comparator Input Capture Enable
When written logic 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 written logic zero, no connection between the analog comparator and the Input Capture function exists. To make the comparator trigger the timer/Counter1 Input Capture interrupt, the TICIE1 bit in the Timer Interrupt Mask Register (TIMSK) must be set.

• 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 below.



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

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.

Analog Comparator Control and Status Register (ACSR)

Multiplexed Input

It is possible to select any of the ADC7..0 pins to replace the negative input to the Analog Comparator. The ADC multiplexer is used to select this input, and consequently, the ADC must be switched off to utilize this feature. If the Analog Comparator Multiplexer Enable bit (ACME in SFIOR) is set and the ADC is switched off (ADEN in ADCSRA is zero), MUX2..0 in ADMUX select the input pin to replace the negative input to the Analog Comparator, as shown in Table 94. If ACME is cleared or ADEN is set, AIN1 is applied to the negative input to the Analog Comparator.

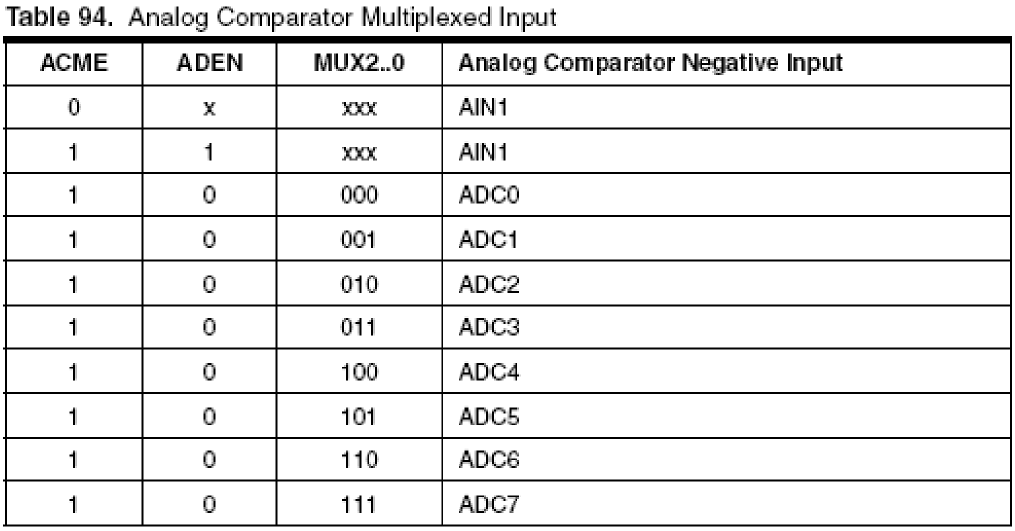
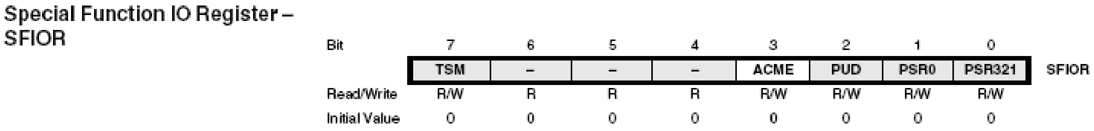


Table 94. Analog Comparator Multiplexed Input

ACME	ADEN	MUX2..0	Analog Comparator Negative Input
0	x	xxx	AIN1
1	1	xxx	AIN1
1	0	000	ADC0
1	0	001	ADC1
1	0	010	ADC2
1	0	011	ADC3
1	0	100	ADC4
1	0	101	ADC5
1	0	110	ADC6
1	0	111	ADC7



Special Function IO Register – SFIOR

Bit	7	6	5	4	3	2	1	0	
	TSM	–	–	–	ACME	PUD	PSR0	PSR321	SFIOR
Read/Write	R/W	R	R	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

• Bit 3 – ACME: Analog Comparator Multiplexer Enable

When this bit is written logic one and the ADC is switched off (ADEN in ADCSRA is zero), the ADC multiplexer selects the negative input to the Analog Comparator. When this bit is written logic zero, AIN1 is applied to the negative input of the Analog Comparator.

For a detailed description of this bit, see “Analog Comparator Multiplexed Input”.

Useful Library Functions

Delay functions #include <intrinsics.h>

- __delay_cycles(no of clock cycles)

Standard Input / Output Functions #include <stdio.h>

- void sprintf(char array, “format_string”[,arg1, arg2,…])

  This function puts the values specified by the arguments (arg1, arg2, ..) into a character array. The format of characters in the array is specified by the “format string”.

LCD Display Driver. #include <labboard.h>

- SLCDInit() : Initiate the SLCD system by initiating the TWI. Call this function first
- SLCDDisplayOn(); DisplayOn : Turn on the display. Until this function is called, nothing will be displayed.
- SLCDDisplayOff() : Turns off the display, but what is saved in display memory is not
- SLCDClearScreen() : Clear the screen
- SLCDHomeCursor() : Set position to 0,0
- // SLCDSetCursorPosition(1,0): Set the position where the next character will be displayed. The first number is the line (0-3) the second is the position in the line(0-19)
-

Universal Synchronous Asynchronous Receiver Transmitter

The AT90 has one USARTS,   called USART1.

Register Description

USART I/O Data Register – UDR1

Bit	7	6	5	4	3	2	1	0	
	RXB[7:0]								**UDR0 (Read) $0C**
	TXB[7:0]								**UDR0 (Write) $0C**
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 USART Transmit Data Buffer register and USART Receive Data Buffer registers share the same I/O address referred to as USART Data Register or UDR1. Writing to this address will send the data to the transmit data buffer register (TXB). Reading the UDR register location will return the contents of the receive data buffer register (RXB).

When data is written to the transmit buffer, and the transmitter is enabled, the transmitter will load the data into the transmit shift register when the shift register is empty. Then the data will be serially transmitted on the TxD pin.

The receive buffer consists of a two level FIFO. The FIFO will change its state whenever the receive buffer is accessed.

USART Control and Status Register A – UCSR1A

Bit	7	6	5	4	3	2	1	0	
	RXC	**TXC**	**UDRE**	**FE**	**DOR**	**UPE**	**U2X**	**MPCM**	**UCSR1A**
Read/Write	R	R/W	R	R	R	R	R/W	R/W	
Initial value	0	0	1	0	0	0	0	0	

• Bit 7 - RXC: USART Receive Complete
This flag bit is set when there is unread data in the receive buffer and cleared when the receive buffer has been read. (empty).

• Bit 6 - TXC: USART Transmit Complete
This flag bit is set when the entire date in the transmit buffer (UDR1), and the transmit shift register has been sent

• Bit 5 - UDRE: USART Data Register Empty
This flag is set, when the transmit buffer is empty and ready to receive new data.

• Bit 4 - FE: Frame Error
This bit is set if the data just received had a Frame Error. i.e. when the stop bit was not detected after the data and parity bits. This bit is valid until the receive buffer (UDR) is read.

• Bit 3 - DOR: Data Over Run
This bit is set if a data overrun condition is detected. A data overrun occurs when the receive buffer is full (contains two new characters), and a new character is waiting in the receive shift register, and a new start bit is detected. This bit is valid until the receive buffer (UDR) is read. Always set this bit to zero when writing to UCSRA.

• Bit 2 - UPE: Parity Error
This bit is set if the new data in the receive buffer had a Parity Error when received and the parity checking was enabled at that point (UPM1 = 1). This bit is valid until the receive buffer (UDR) is read. Always set this bit to zero when writing to UCSRA.

• Bit 1 - U2X: Double the USART Transmission Speed
This bit only has effect for the asynchronous operation. Clear this bit when using synchronous operation. Setting this bit will double the transfer rate for asynchronous communication.

• Bit 0 - MPCM: Multi-Processor Communication Mode
This bit enables the Multi-processor Communication Mode. When the bit is set, all the incoming frames received by the USART receiver that do not contain address information will be ignored. The transmitter is unaffected by the MPCM setting.

USART Control and Status Register B – UCSR1B

Bit	7	6	5	4	3	2	1	0	
	RXCIE	**TXCIE**	**UDRIE**	**RXEN**	**TXEN**	**UCSZ2**	**RXB8**	**TXB8**	**UCSR1B**
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	
Initial value	0	0	0	0	0	0	0	0	

• Bit 7 - RXCIE: Receive Complete Interrupt Enable
Setting this bit enables interrupt on the RXC flag. A receive complete interrupt will occur when the RXCIE bit is set, the global interrupt flag in SREG is set and a new byte of data has been received (receiver buffer register is full).

• Bit 6 - TXCIE: Transmit Complete Interrupt Enable
Setting this bit enables interrupt on the TXC flag. A transmit complete interrupt will occur when the TXCIE bit is set, the global interrupt flag in SREG is set and the transmit buffer and transmit shift register is empty.

• Bit 5 - UDRIE: USART Data Register Empty Interrupt Enable
Setting this bit enables interrupt on the UDRE flag. A Data Register Empty interrupt will occur when the UDRIE bit is set, the global interrupt flag in SREG is set and the transmit buffer register UDR1 is empty.

• Bit 4 - RXEN: Receiver Enable
Setting this bit turns on the USART receiver..

• Bit 3 - TXEN: Transmitter Enable
Setting this bit turns on the USART transmitter.

• Bit 2 - UCSZ2: Character Size
This bit combined with the UCSZ1:0 bits in UCSRC sets the number of data bits (character size) in a frame the receiver and transmitter use.

• Bit 1 - RXB8: Receive Data Bit 8
RXB8 is the 9th data bit of the received character when operating with serial frames with 9 data bits. Must be read before reading the low bits from UDR.

• Bit 0 - TXB8: Transmit Data Bit 8
TXB8 is the 9th data bit in the character to be transmitted when operating with serial frames with 9 data bits. Must be written before writing the low bits to UDR.

USART Control and Status Register C – UCSR1C

Bit	7	6	5	4	3	2	1	0	
	–	**UMSEL**	**UPM1**	**UPM0**	**USBS**	**UCSZ1**	**UCSZ0**	**UCPOL**	**UCSR1C**
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	1	1	0	

• Bit 7 - Reserved Bit This bit is reserved for future use.

• Bit 6 - UMSEL: USART Mode Select

This bit selects between asynchronous and synchronous mode of operation.

UMSEL	Mode
0	Asynchronous Operation
1	Synchronous Operation

• Bit 5:4 - UPM1:0: Parity Mode

These bits enable and set type of parity generation and check. If enabled, the transmitter will automatically generate and send the parity of the transmitted data bits within each frame. The receiver will generate a parity value for the incoming data and compare it to the UPM0 setting. If a mismatch is detected, the UPE flag in UCSRA will be set.

UPM1	UPM0	Parity Mode
0	0	Disabled
0	1	(Reserved)
1	0	Enabled, Even Parity
1	1	Enabled, Odd Parity

• Bit 3 - USBS: Stop Bit Select

This bit selects the number of stop bits to be inserted by the transmitter. The receiver ignores this setting.

USBS	**Stop Bit(s)**
0	1-bit
1	2-bits

• Bit 2:1 - UCSZ1:0: Character Size

These bits combined with the UCSZ2 bit in UCSRB sets the number of data bits (character size) in a frame the receiver and transmitter use.

UCSZ2	**UCSZ1**	**UCSZ0**	**Character Size**
0	0	0	5-bit
0	0	1	6-bit
0	1	0	7-bit
0	1	1	8-bit
1	0	0	(reserved)
1	0	1	(reserved)
1	1	0	(reserved)
1	1	1	9-bit

• Bit 0 - UCPOL: Clock Polarity

This bit is used for synchronous mode only. Clearing this bit when asynchronous mode is used. The UCPOL bit sets the relationship between data output change and data input sample, and the synchronous clock (XCK).

UCPOL	**Transmitted Data Changed (Output of TxD Pin)**	**Received Data Sampled (Input on RxD Pin)**
0	Falling XCK Edge	Rising XCK Edge
1	Rising XCK Edge	Falling XCK Edge

USART Baud Rate Registers – UBRR1L and UBRR1H

Bit	15	14	13	12	11	10	9	8	
	–	–	–	–	UBRR[11:8]				**UBRR0H ($90)**
	UBRR[7:0]								**UBRR0L ($09)**
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	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	

• Bit 15:12 - Reserved Bits
These bits are reserved for future use. For compatibility with future devices, these bit must be written to zero when UBRRH is written.

• Bit 11:0 - UBRR11:0: USART Baud Rate Register
This is a 12-bit register which contains the USART baud rate. The UBRRH contains the 4 most significant bits, and the UBRRL contains the 8 least significant bits of the USART baud rate. Ongoing transmissions by the transmitter and receiver will be corrupted if the baud rate is changed. Writing UBRRL will trigger an immediate update of the baud rate prescaler.

UBRR settings for a range of baud rates are given below for 4 and 8 MHz system clocks.

Note: 1. The values given are decimal (not hexadecimal)
2. **U2X** = 0 gives normal USART transmission speed
U2X = 1 gives **double** USART transmission speed

Baud Rate (bps)	**^{f}osc = 4.0000 MHz**				**^{f}osc = 8.0000 MHz**			
	U2X = 0		**U2X = 1**		**U2X = 0**		**U2X = 1**	
	UBRR	Error	UBRR	Error	UBRR	Error	UBRR	Error
2400	103	0.2%	207	0.2%	207	0.2%	416	-0.1%
4800	51	0.2%	103	0.2%	103	0.2%	207	0.2%
9600	25	0.2%	51	0.2%	51	0.2%	103	0.2%
14.4k	16	2.1%	34	-0.8%	34	-0.8%	68	0.6%
19.2k	12	0.2%	25	0.2%	25	0.2%	51	0.2%
28.8k	8	-3.5%	16	2.1%	16	2.1%	34	-0.8%
38.4k	6	-7.0%	12	0.2%	12	0.2%	25	0.2%
57.6k	3	8.5%	8	-3.5%	8	-3.5%	16	2.1%
76.8k	2	8.5%	6	-7.0%	6	-7.0%	12	0.2%
115.2k	1	8.5%	3	8.5%	3	8.5%	8	-3.5%
230.4k	0	8.5%	1	8.5%	1	8.5%	3	8.5%
250k	0	0.0%	1	0.0%	1	0.0%	3	0.0%
0.5M	–	–	0	0.0%	0	0.0%	1	0.0%
1M	–	–	–	–	–	–	0	0.0%
Max (1)	250 kbps		0.5 Mbps		0.5 Mbps		1 Mbps	

To calculate the UBRR value for a required baud rate the formula below applies.

Asynchronous normal mode $\text{UBRR} = (f_{osc}/(16 * \text{BAUD})) -1$

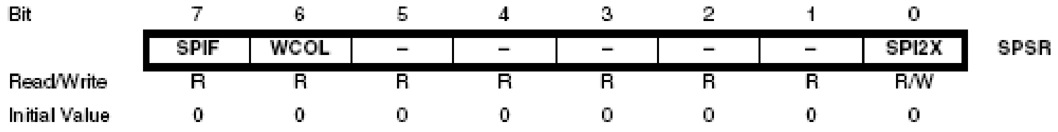
Where f_{osc} = the frequency of the system clock

Serial Peripheral Interface Registers

The status register of the SPI has one bit (SPIF) to indicate when a receive or transmit is complete and another (WCOL) to indicate a write collision.
The control register provides the following interrupt enable, SPI enable, Data order selection (MSB or LSB first), Master or slave select, Clock polarity in idle mode, Active clock edge, Clock prescale.

The same data register address is used for transmit and receive. (SPDR).

SPI Status Register – SPSR



Bit	7	6	5	4	3	2	1	0	
	SPIF	WCOL	–	–	–	–	–	SPI2X	SPSR
Read/Write	R	R	R	R	R	R	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

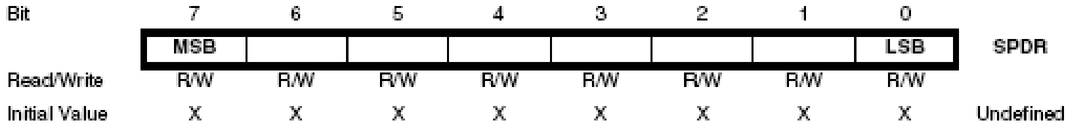
• Bit 7 – SPIF: SPI Interrupt Flag
When a serial transfer is complete, the SPIF flag is set. An interrupt is generated if SPIE in SPCR is set and global interrupts are enabled. If SS is an input and is driven low when the SPI is in Master mode, this will also set the SPIF flag. SPIF is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, the SPIF bit is cleared by first reading the SPI Status Register with SPIF set, then accessing the SPI Data Register (SPDR).

• Bit 6 – WCOL: Write Collision flag
The WCOL bit is set if the SPI Data Register (SPDR) is written during a data transfer. The WCOL bit (and the SPIF bit) are cleared by first reading the SPI Status Register with WCOL set, and then accessing the SPI Data Register.

• Bit 5..1 – Res: Reserved Bits
These bits are reserved bits in the ATmega128 and will always read as zero.

• Bit 0 – SPI2X: Double SPI Speed Bit
When this bit is written logic one the SPI speed (SCK Frequency) will be doubled when the SPI is in Master mode (see Table 72). This means that the minimum SCK period will be 2 CPU clock periods. When the SPI is configured as Slave, the SPI is only guaranteed to work at fosc /4 or lower. The SPI interface on the ATmega128 is also used for program memory and EEPROM downloading or uploading. See page 303 for SPI Serial Programming and verification.

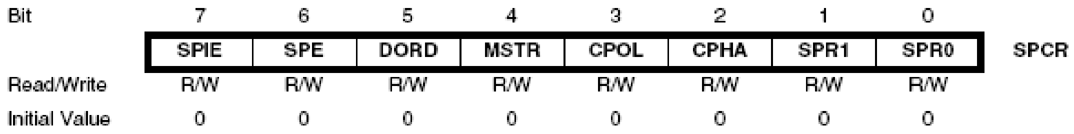
SPI Data Register – SPDR



Bit	7	6	5	4	3	2	1	0	
	MSB							LSB	SPDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	X	X	X	X	X	X	X	X	Undefined

The SPI Data Register is a Read/Write Register used for data transfer between the register file and the SPI Shift Register. Writing to the register initiates data transmission. Reading the register causes the Shift Register Receive buffer to be read.

SPI Control Register – SPCR



Bit	7	6	5	4	3	2	1	0	
	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	SPCR
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 – SPIE: SPI Interrupt Enable

This bit causes the SPI interrupt to be executed if SPIF bit in the SPSR Register is set and the if the global interrupt enable bit in SREG is set.

• Bit 6 – SPE: SPI Enable

When the SPE bit is written to one, the SPI is enabled. This bit must be set to enable any SPI operations.

• Bit 5 – DORD: Data Order

When the DORD bit is written to one, the LSB of the data word is transmitted first. When the DORD bit is written to zero, the MSB of the data word is transmitted first.

• Bit 4 – MSTR: Master/Slave Select

This bit selects Master SPI mode when written to one, and Slave SPI mode when written logic zero. If SS is configured as an input and is driven low while MSTR is set, MSTR will be cleared, and SPIF in SPSR will become set. The user will then have to set MSTR to re-enable SPI Master mode.

• Bit 3 – CPOL: Clock Polarity

When this bit is written to one, SCK is high when idle. When CPOL is written to zero, SCK is low when idle. The CPOL functionality is summarized below:

CPOL	Leading Edge	Trailing Edge
0	Rising	Falling
1	Falling	Rising

• Bit 2 – CPHA: Clock Phase

The settings of the clock phase bit (CPHA) determine if data is sampled on the leading (first) or trailing (last) edge of SCK.
The CPHA functionality is summarized below:

CPHA	Leading Edge	Trailing Edge
0	Sample	Setup
1	Setup	Sample

• Bits 1, 0 – SPR1, SPR0: SPI Clock Rate Select 1 and 0

These two bits control the SCK rate of the device configured as a master. SPR1 and SPR0 have no effect on the slave. The relationship between SCK and the Oscillator Clock frequency f_{osc} is shown in the following table:

Relationship Between SCK and the Oscillator Frequency

SPI2X	SPR1	SPR2	SCK frequency
0	0	0	$f_{osc}/4$
0	0	1	$f_{osc}/16$
0	1	0	$f_{osc}/64$
0	1	1	$f_{osc}/128$
1	0	0	$f_{osc}/2$
1	0	1	$f_{osc}/8$
1	1	0	$f_{osc}/32$
1	1	1	$f_{osc}/64$

8 Bit Timer/Counters TCNT0 (TCNT2)

Timer/Counter Register – TCNT0 (TCNT2)

Bit	7	6	5	4	3	2	1	0	
	TCNT0[7:0]								**TCNT0 $32 ($52)**
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/Counter Register gives direct access, both for read and write operations, to the Timer/Counter unit 8-bit counter.

Output Compare Register – OCR0A, OCROB (OCR2A, OCR2B)

Bit	7	6	5	4	3	2	1	0	
	OCR0[7:0]								**OCR0 $31 ($51)**
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 contains the value that is compared with the counter value (TCNT0). When a match occurs the OCF0 flag is set in TIFR. A match can be used to generate an output compare interrupt, or to generate a waveform output on the OC0 pin.

Timer/Counter Control Register - TCCR0A (TCCR2A)

Bit	7	6	5	4	3	2	1	0	
	COM0A1	**COM0A0**	**COM0B1**	**COM0B0**			**WGM01**	**WGM00**	**TCCR0A**
Read/Write	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 7:6 COM0A1:A0 Compare Match Output Mode
Bit 5:4 COM0B1:B0 Compare Match Output Mode

These bits control the output compare pin (OC0A/B) behavior. If one or both of the COM01:0 bits are set, the OC0 output overrides the normal port functionality of the I/O pin it is connected to. However, note that the Data Direction Register (DDR) bit corresponding to OC0 pin must be set in order to enable the output driver.

When OC0A/B is connected to the pin, the function of the COM01:0 bits depends on the WGM01:0 bit setting.

*Normal or CTC mode (**non-PWM**).*

COM01	COM00	Description
0	0	Normal parallel port operation, **OC0** disconnected from output pin.
0	1	Toggle **OC0** on compare match
1	0	Clear **OC0** on compare match
1	1	Set **OC0** on compare match

*Compare output mode **Fast PWM mode***

COM01	COM00	Description
0	0	Normal parallel port operation, **OC0** disconnected from output pin.
0	1	Reserved
1	0	Clear **OC0** on compare match, set **OC0** at TOP
1	1	Set **OC0** on compare match, clear **OC0** at TOP

*Compare output mode **Phase correct PWM mode***

COM01	COM00	Description
0	0	Normal parallel port operation, **OC0** disconnected from output pin.
0	1	Reserved
1	0	Clear **OC0** on compare match when up-counting. Set **OC0** on compare match when downcounting.
1	1	Set **OC0** on compare match when up-counting. Clear **OC0** on compare match when downcounting.

Bit 3:2 Reserved

Bit 1:0 WGM01:0 Waveform Generation Mode

These bits (including WGM02) control the counting sequence of the counter, the source for the maximum (TOP) counter value, and what type of waveform generation to be used. Modes of operation supported by the Timer/Counter unit are: Normal mode, Clear Timer on Compare match (CTC) mode, and two types of Pulse Width Modulation (PWM) modes.

Waveform generation mode bit description

Mode	WGM 02	WGM 01	WGM 00	Mode of Operation	TOP	Update of OCR0 at	TOV0 Flag Set on
0	0	0	0	Normal	0xFF	Immediate	MAX
1	0	0	1	PWM, Phase Correct	0xFF	TOP	BOTTOM
2	0	1	0	CTC	OCR0	Immediate	MAX
3	0	1	1	Fast PWM	0xFF	TOP	MAX
4	1	0	0	Reserved	----	----	----
5	1	0	1	PWM, Phase Correct	OCRA	TOP	BOTTOM
6	1	1	0	Reserved	----	----	----
7	1	1	1	Fast PWM	OCRA	TOP	TOP

TCCR0B /2B

Bit	7	6	5	4	3	2	1	0	
	FOC0A	**FOC0AB**			**WGM02**	**CS02**	**CS01**	**CS00**	**TCCR0A**
Read/Write	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 7:6 Force output compare A/B

When a one is written to this bit an immediate compare match is forced on the waveform generation

Bit 5:4 Reserved

Bit 3 WGM02 Waveform Generation mode

Used in conjunction with WGM01:WGM00 to set the mode of operation

Bit 2:0 CS02:0 Clock Select

The three clock select bits select the clock source to be used by Timer/Counter 0

CS02	CS01	CS00	Description
0	0	0	No clock source (Timer/counter stopped)
0	0	1	$^{\text{clk}}$T0S/(No prescaling)
0	1	0	$^{\text{clk}}$T0S/8 (From prescaler)
0	1	1	$^{\text{clk}}$T0S/32 (From prescaler)
1	0	0	$^{\text{clk}}$T0S/64 (From prescaler)
1	0	1	$^{\text{clk}}$T0S/128 (From prescaler)
1	1	0	$^{\text{clk}}$T0S/256 (From prescaler)
1	1	1	$^{\text{clk}}$T0S/1024 (From prescaler)

Note: Counter/timer 2 uses the clock select table for 16-bit timer (page 24)

Timer/Counter Interrupt Mask Register – TIMSK0 (2)

Bit	7	6	5	4	3	2	1	0	
						OCIE0B	OCIE0A	TOIE0	TIMSK0 (2)

Bit 7:3 Reserved

Bit 2 OCIE0B Timer/Counter 0 Output Compare B Interrupt Enable

When the OCIE0B bit is written to one, the Timer/Counter0 Compare Match B interrupt is enabled and will be is executed if a Compare Match occurs

Bit 1 OCIE0A Timer/Counter 0 Output Compare A Interrupt Enable

When the OCIE0B bit is written to one, the Timer/Counter0 Compare Match A interrupt is enabled and will be executed if a Compare Match occurs

Bit 0 TOIE0 Timer/Counter 0 Overflow flag interrupt enable

When the TOIE0 bit is written to one, the Timer/Counter0 overflow interrupt is enabled and will be executed when timer goes from FF to 00.

Timer/Counter Interrupt Flag Register – TIFR0 (2)

Bit	7	6	5	4	3	2	1	0	
						OCF0B	OCF0A	TOV0	TIFR0 (2)

Bit 7:3 Reserved

Bit 2 OCF0B Timer/Counter 0 Output Compare B match flag

This bit is set when a compare match occurs between the timer/counter TCNT0 and the output compare register B OCR0B. This flag is automatically cleared if an interrupt occurs, and can be manually cleared by writing a one to it

Bit 1 OCF0A Timer/Counter 0 Output Compare A match flag

This bit is set when a compare match occurs between the timer/counter TCNT0 and the output compare register B OCR0A. This flag is automatically cleared if an interrupt occurs, and can be manually cleared by writing a one to it.

Bit 0 TOV0 Timer/Counter 0 Overflow flag

This bit is set when an overflow occurs in timer 0, e.g. counter goes from 0xFF to 0x00. This flag is automatically cleared if an interrupt occurs, and can be manually cleared by writing a one to it.

16-bit Timer/Counter Register Description

Timer/Counter 1 (3) Control Register A – TCCR1A (TCCR3A)

Bit	7	6	5	4	3	2	1	0	
	COM1A1	COM1A0	COM1B1	COM1B0	COM1C1	COM1C0	WGM11	WGM10	TCCR1A
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 7:6 COMnA1:0 Compare Output Mode for Channel A
Bit 5:4 COMnB1:0 Compare Output Mode for Channel B
Bit 3:2 COMnC1:0 Compare Output Mode for Channel C

The COMnA1:0, COMnB1:0, and COMnC1:0 control the output compare pins (OCnA, OCnB, and OCnC respectively) behavior. If one or both of the COMnx1:0 bits are written to one, the OCnx output overrides the normal port functionality of the I/O pin it is connected to. However, note that the Data Direction Register (DDR) bit corresponding to the OCnA, OCnB or OCnC pin must be set in order to enable the output driver.

When the OCnA, OCnB or OCnC is connected to the pin, the function of the COMnx1:0 bits is dependent of the WGMn3:0 bits setting. The tables below show the COMnx1:0 bit functionality when the WGMn3:0 bits are set to a normal or a CTC mode (non-PWM).

*Compare Output Mode**, Non-PWM***

COMnA1/COMnB1/ COMnC1	COMnA0/COMnB0/ COMnC0	Description
0	0	Normal port operation, OCnA/OCnB/OCnC disconnected.
0	1	Toggle OCnA/OCnB/OCnC on compare match
1	0	Clear OCnA/OCnB/OCnC on compare match (Set output to low level)
1	1	Set OCnA/OCnB/OCnC on compare match (Set output to high level)

*Compare Output mode**, Fast PWM***

COMnA1/COMnB1/ COMnC1	COMnA0/COMnB0/ COMnC0	Description
0	0	Normal port operation, OCnA /OCnB /OCnC disconnected.
0	1	WGMn3=0: Normal port operation, OCnA /OCnB /OCnC disconnected. WGMn3=1: Toggle OCnA on compare match, OCnB/OCnC reserved.
1	0	Clear OCnA /OCnB /OCnC on compare match, set OCnA /OCnB /OCnC at TOP
1	1	Set OCnA/OCnB/OCnC on compare match, clear OCnA/OCnB/OCnC at TOP

*Compare output mode, **phase and frequency correct PWM***

COMnA1/COMnB / COMnC1	COMnA0/COMnB0/ COMnC0	Description
0	0	Normal port operation, OCnA /OCnB /OCnC disconnected.
0	1	WGMn3=0: Normal port operation, OCnA /OCnB /OCnC disconnected. WGMn3=1: Toggle OCnA on compare match, OCnB/OCnC reserved.
1	0	Clear OCnA/OCnB/OCnC on compare match when up-counting. Set OCnA /OCnB /OCnC on compare match when downcounting.
1	1	Set OCnA /OCnB /OCnC on compare match when up-counting. Clear OCnA /OCnB /OCnC on compare match when downcounting.

Bit 1:0 - WGMn1:0: Waveform Generation Mode

Combined with the WGMn3:2 bits found in the TCCRnB register, these bits control the counting sequence of the counter, the source for maximum (TOP) counter value, and what type of waveform generation to be used. Modes of operation supported by the Timer/Counter unit are: Normal mode (counter), Clear Timer on Compare match (CTC) mode, and three types of Pulse Width Modulation (PWM) modes.

Waveform generation mode bit description

Mode	WGMn3	WGMn2 (CTCn)	WGMn1 (PWMn1)	WGMn0 (PWMn0)	Timer/Counter Mode of Operation	TOP	Update of OCRnx at	TOVn Flag Set on
0	0	0	0	0	Normal	0xFFFF	Immediate	MAX
1	0	0	0	1	PWM, Phase Correct, 8-bit	0x00FF	TOP	BOTTOM
2	0	0	1	0	PWM, Phase Correct, 9-bit	0x01FF	TOP	BOTTOM
3	0	0	1	1	PWM, Phase Correct, 10-bit	0x03FF	TOP	BOTTOM
4	0	1	0	0	CTC	OCRnA	Immediate	MAX (TOP)
5	0	1	0	1	Fast PWM, 8-bit	0x00FF	TOP	TOP
6	0	1	1	0	Fast PWM, 9-bit	0x01FF	TOP	TOP
7	0	1	1	1	Fast PWM, 10-bit	0x03FF	TOP	TOP
8	1	0	0	0	PWM, Phase and Frequency Correct	ICRn	BOTTOM	BOTTOM
9	1	0	0	1	PWM, Phase and Frequency Correct	OCRnA	BOTTOM	BOTTOM
10	1	0	1	0	PWM, Phase Correct	ICRn	TOP	BOTTOM
11	1	0	1	1	PWM, Phase Correct	OCRnA	TOP	BOTTOM
12	1	1	0	0	CTC	ICRn	Immediate	MAX (TOP)
13	1	1	0	1	(Reserved)	–	–	–
14	1	1	1	0	Fast PWM	ICRn	TOP	TOP
15	1	1	1	1	Fast PWM	OCRnA	TOP	TOP

Timer/Counter 1 (and 3) Control Register B – TCCR1B (TCCR3B)

Bit	7	6	5	4	3	2	1	0	
	ICNC1	**ICES1**	**–**	**WGM13**	**WGM12**	**CS12**	**CS11**	**CS10**	**TCCR1B**
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

Bit 7 - ICNCn: Input Capture Noise Canceler
Setting this bit (to one) activates the input capture noise canceler. When the noise canceler is activated, the input from the Input Capture Pin (ICPn) is filtered.

Bit 6 - ICESn: Input Capture Edge Select
This bit selects which edge on the Input Capture Pin (ICPn) that is used to trigger a capture event. When the ICESn bit is written to zero, a falling (negative) edge is used as trigger, and when the ICESn bit is written to one, a rising (positive) edge will trigger the capture.

When a capture is triggered according to the ICESn setting, the counter value is copied into the Input Capture Register (ICRn). The event will also set the Input Capture Flag (ICFn), and this can be used to cause an Input Capture Interrupt, if this interrupt is enabled.

When the ICRn is used as TOP value (see description of the WGMn3:0 bits located in the TCCRnA and the TCCRnB register), the ICPn is disconnected and consequently the input capture function is disabled.

Bit 5 - Reserved Bit
This bit is reserved for future use.

Bit 4:3 - WGMn3:2: Waveform Generation Mode
See TCCRnA register description.

Bit 2:0 - CSn2:0: Clock Select
The three clock select bits select the clock source to be used by the Timer/Counter.

Clock select bit description

CSn2	CSn1	CSn0	Description
0	0	0	No clock source. (Timer/counter stopped)
0	0	1	^{clk}I/O / 1 no prescaling
0	1	0	^{clk}I/O / 8 from prescaler
0	1	1	^{clk}I/O / 64 from prescaler
1	0	0	^{clk}I/O / 256 from prescaler
1	0	1	^{clk}I/O / 1024 from prescaler
1	1	0	External clock source on Tn pin. Clock on falling edge
1	1	1	External clock source on Tn pin. Clock on rising edge

Note: This table also applies to counter/timer 2

Timer/Counter 1 (and 3) Control Register C – TCCR1C   (TCCR3C)

Bit	7	6	5	4	3	2	1	0	
	FOC1A	**FOC1B**	**FOC1C**	**–**	**–**	**–**	**–**	**–**	**TCCR1C**
Read/Write	W	W	W	R	R	R	R	R	
Initial value	0	0	0	0	0	0	0	0	

Bit 7- FOCnA: Force Output Compare for Channel A
Bit 6- FOCnB: Force Output Compare for Channel B
Bit 5- FOCnC: Force Output Compare for Channel C

The FOCnA/FOCnB/FOCnC bits are only active when the WGMn3:0 bits specifies a non-PWM mode. When writing a logical one to the FOCnA/FOCnB/FOCnC bit, an immediate compare match is forced on the waveform generation unit. The OCnA/OCnB/OCnC output is changed according to its COMnx1:0 bits setting. Note that the FOCnA/FOCnB/FOCnC bits are implemented as strobes. Therefore it is the value present in the COMnx1:0 bits that determine the effect of the forced compare.

A FOCnA/FOCnB/FOCnC strobe will not generate any interrupt nor will it clear the timer in clear timer on compare match (CTC) mode using OCRnA as TOP. The FOCnA/FOCnB/FOCnB bits are always read as zero.

• Bit 4:0 - Reserved Bits
These bits are reserved for future use.

The Counters

Reading and Writing 16-bit Timer Registers

Note this action is automatically performed in the IAR C compiler. To ensure that both the high and low bytes are read and written simultaneously when the CPU accesses 16-bit registers, the access is performed using an 8-bit temporary high byte register (TEMP). This temporary register is shared by all the other 16-bit registers.

Because the temporary register saves the high byte of the 16-bits this means **for any** 16-bit register –

- When **writing** -   write the **high byte first**
- When **reading**   read the **low byte first**

Timer/Counter 1 (and 3) – TCNT1H and TCNT1L   (TCNT3H and TCNT3L)

Bit	7	6	5	4	3	2	1	0	
	TCNT1[15:8]								**TCNT1H**
	TCNT1[7:0]								**TCNT1L**
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	

.

16-bit Output Compare registers

The information on this page refers to both 16-bit Timer/Counters
TCNT1 and TCNT3

Output Compare Register 1 A – OCR1AH and OCR1AL (OCR3AH, OCR3AL)

Bit	7	6	5	4	3	2	1	0	
	OCR1A[15:8]								**OCR1AH**
	OCR1A[7:0]								**OCR1AL**
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	

Output Compare Register 1 B – OCR1BH and OCR1BL (OCR3BH, OCR3BL)

Bit	7	6	5	4	3	2	1	0	
	OCR1B[15:8]								**OCR1BH**
	OCR1B[7:0]								**OCR1BL**
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	

Output Compare Register 1 C – OCR1CH and OCR1CL (OCR3CH, OCR3CL)

Bit	7	6	5	4	3	2	1	0	
	OCR1C[15:8]								**OCR1CH**
	OCR1C[7:0]								**OCR1CL**
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 registers contain a 16-bit value that is continuously compared with the counter value (TCNTn). A match can be used to generate an output compare interrupt, or to generate a waveform output on the OCnx pin. The output compare registers are 16 bit in size.

- When **writing** - write the **high byte first**
- When **reading** read the **low byte first**

16-bit Input Capture Register

Input Capture Register 1 – ICR1H and ICR1L (ICR3H, ICR3L)

Bit	7	6	5	4	3	2	1	0	
	ICR1[15:8]								**ICR1H**
	ICR1[7:0]								**ICR1L**
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 input capture is updated with the counter (TCNTn) value each time an event occurs on the ICPn pin (or optionally on the analog comparator output for Timer/Counter1). The input capture can be used for defining the counter TOP value.

- When **writing** - write the **high byte first**
- When **reading** read the **low byte first**

Timer/Counter Interrupt Mask Register – TIMSK1 (3)

Bit	7	6	5	4	3	2	1	0	
			ICIE1		**OCIE1C**	**OCIE1B**	**OCIE1A**	**TOIE1**	**TIMSK1**

Bit 5 ICIE1: Timer/Counter 1, Input Capture Interrupt Enable
When this bit is set the timer/counter 1 input capture interrupt is enabled. The corresponding interrupt vector is executed when the ICF1 flag, located in TIFR, is set

Bit 3 OCIE1C: Timer/Counter 1, Output Compare C Match Interrupt Enable
When the OCIE0B bit is written to one, the Timer/Counter1 Compare Match C interrupt is enabled and will be is executed if a Compare Match occurs

Bit 1 OCIE1B Timer/Counter 1 Output Compare B Interrupt Enable
When the OCIE0B bit is written to one, the Timer/Counter1 Compare Match B interrupt is enabled and will be executed if a Compare Match occurs

Bit 1 OCIE0A Timer/Counter 1 Output Compare A Interrupt Enable
When the OCIE0B bit is written to one, the Timer/Counter1 Compare Match A interrupt is enabled and will be executed if a Compare Match occurs

Bit 0 TOIE1 Timer/Counter 0 Overflow flag interrupt enable
When the TOIE0 bit is written to one, the Timer/Counter1 overflow interrupt is enabled and will be executed when timer goes from FF to 00.

Timer/Counter Interrupt Flag Register – TIFR1 (3)

Bit	7	6	5	4	3	2	1	0	
			ICF1		**OCF1C**	**OCF1B**	**OCF1A**	**TOV1**	**TIFR0 (2)**

Bit 5 ICF1: Timer/Counter 1, Input Capture Flag
This flag is set when a capture event occurs on the ICP1 pin.ICF1 is automatically cleared when the Input Capture Interrupt vector is executed. Alternatively, ICF1 can be cleared by writing a logic one to its bit location

Bit 3 OCF1C Timer/Counter 1 Output Compare C match flag
This bit is set when a compare match on OCR1C occurs. This flag is automatically cleared if an interrupt occurs, and can be manually cleared by writing a one to it

Bit 2 OCF0B Timer/Counter 1 Output Compare B match flag
This bit is set when a compare match on OCR1B occurs. This flag is automatically cleared if an interrupt occurs, and can be manually cleared by writing a one to it

Bit 1 OCF1A Timer/Counter 0 Output Compare A match flag
This bit is set when a compare match on OCR1A occurs. This flag is automatically cleared if an interrupt occurs, and can be manually cleared by writing a one to it.

Bit 0 TOV1 Timer/Counter 0 Overflow flag
This bit is set when an overflow occurs in timer 1, e.g. counter goes from 0xFFFF to 0x0000. This flag is automatically cleared if an interrupt occurs, and can be manually cleared by writing a one to it.

The AVR status register – SREG

Bit	7	6	5	4	3	2	1	0	
	I	**T**	**H**	**S**	**V**	**N**	**Z**	**C**	**SREG $3F ($5F)**
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 for the interrupts to be enabled. The individual interrupt enable control is then performed in separate control registers. If the global interrupt enable register is cleared, none of the interrupts are enabled independent of the individual interrupt enable settings. The I-bit is cleared by hardware after an interrupt has occurred, and is set by the RETI instruction to enable subsequent interrupts. The I-bit can also be set and cleared in software with the SEI and CLI instructions.

• Bit 6 - T: Bit Copy Storage
The bit copy instructions BLD (Bit LoaD) and BST (Bit STore) use the T bit as source or 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. Half carry is useful in BCD arithmetic.

• 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.

• Bit 3 - V: Two's Complement Overflow Flag
The two's complement overflow flag V supports two's complement arithmetic.

• Bit 2 - N: Negative Flag
The negative flag N indicates a negative result in an arithmetic or logic operation.

• Bit 1 - Z: Zero Flag
The zero flag Z indicates a zero result in an arithmetic or logic operation.

• Bit 0 - C: Carry Flag
The carry flag C indicates a carry in an arithmetic or logic operation.

External Interrupt Registers

The External Interrupt Control Registers (EICRA and EICRB) contain **sense control bits**.
The nature of the external input signal which causes the interrupt is determined by the settings in these registers.

External interrupt Control Register A - EICRA

Bit	7	6	5	4	3	2	1	0	
	ISC31	ISC30	ISC21	ISC20	ISC11	ISC10	**ISC01**	**ISC00**	**EICRA $6A**
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	

ISCn1	ISCn0	Description
0	0	The low level of INTn generates an interrupt request
0	1	Reserved
1	0	The falling edge between two samples of INTn generates an interrupt request
1	1	The rising edge between two samples of INTn generates an interrupt request

External Interrupt Control Register B - EICRB

Bit	7	6	5	4	3	2	1	0	
	ISC71	**ISC70**	ISC61	ISC60	**ISC51**	**ISC50**	ISC41	ISC40	**EICRB $5A**
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	

ISCn1	ISCn0	Description
0	0	The low level of INTn generates an interrupt request
0	1	Any logic change on INTn generates an interrupt request.
1	0	The falling edge between two samples of INTn generates an interrupt request
1	1	The rising edge between two samples of INTn generates an interrupt request

External Interrupt Flag Register - EIFR

When an active interrupt signal occurs on any external interrupt pin the corresponding flag in this register is set. The flag is cleared when the interrupt routine is executed, or by writing a 1 to it.

Bit	7	6	5	4	3	2	1	0	
	INTF7	INTF6	**INTF5**	INTF4	INTF3	INTF2	INTF1	**INTF0**	**EIFR $38**
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	

External Interrupt Mask Register - EIMSK

To enable an interrupt from a given external interrupt set the corresponding bit to 1.

Bit	7	6	5	4	3	2	1	0	
	INT7	INT6	**INT5**	INT4	INT3	INT2	INT1	**INT0**	**EIMSK $39**
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	

Interrupt Vector Table

Vector No	Program Address	Source	Interrupt Definition
1	$0000	RESET_vect	Reset
2	$0002	INT0_vect	External Interrupt Request 0
3	$0004	INT1_vect	External Interrupt Request 1
4	$0006	INT2_vect	External Interrupt Request 2
5	$0008	INT3_vect	External Interrupt Request 3
6	$000A	INT4_vect	External Interrupt Request 4
7	$000C	INT5_vect	External Interrupt Request 5
8	$000E	INT6_vect	External Interrupt Request 6
9	$0010	INT7_vect	External Interrupt Request 7
10	$0012	PCINT0_vect	Pin Change Interrupt Request 0
11	$0014	USB_General_vect	USB General Interrupt request
12	$0016	USB_Pipe__vect	USB Endpoint/Pipe Interrupt request
13	$0018	WDT_vect	Watchdog Time-out Interrupt
14	$001A	TIMER2_COMPA_vect	Timer/Counter2 Compare Match A
15	$001C	TIMER2_COMPB_vect	Timer/Counter2 Compare Match B
16	$001E	TIMER2_OVF_vect	Timer/Counter2 Overflow
17	$0020	TIMER1_CAPT_vect	Timer/Counter1 Capture Event
18	$0022	TIMER1_COMPA_vect	Timer/Counter1 Compare Match A
19	$0024	TIMER1_COMPB_vect	Timer/Counter1 Compare Match B
20	$0026	TIMER1_COMPC__vect	Timer/Counter1 Compare Match C
21	$0028	TIMER1_OVF_vect	Timer/Counter1 Overflow
22	$002A	TIMER0_COMPA_vect	Timer/Counter0 Compare Match A
23	$002C	TIMER0_COMPB_vect	Timer/Counter0 Compare match B
24	$002E	TIMER0_OVF_vect	Timer/Counter0 Overflow
25	$0030	SPI_STC_vect	SPI Serial Transfer Complete
26	$0032	USART1_RX_vect	USART1 Rx Complete
27	$0034	USART1_UDRE_vect	USART1 Data Register Empty
28	$0036	USART1_TX_vect	USART1 Tx Complete
2 9	$0038	ANALOG_COMP_vect	Analog Comparator
30	$003A	ADC_vect	ADC Conversion Complete
31	$003C	EE_READY_vect	EEPROM Ready
32	$003E	TIMER3_CAPT_vect	Timer/Counter3 Capture Event
33	$0040	TIMER3_COMPA_vect	Timer/Counter3 Compare Match A
34	$0042	TIMER3_COMPB_vect	Timer/Counter3 Compare Match B
35	$0044	TIMER3_COMPC_vect	Timer/Counter3 Compare Match C
36	$0046	TIMER3_OVF_vect	Timer/Counter3 Overflow
37	$0048	TWI_vect	2-wire Serial Interface
38	$004A	SPM_ READY_vect	Store Program Memory Ready

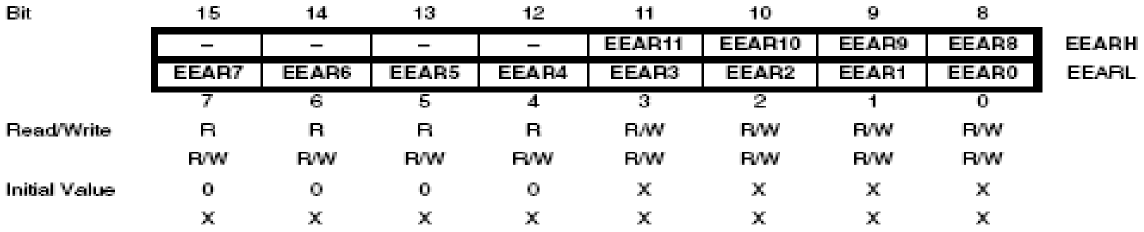
Note: example of declaring an interrupt service routine.

```
#pragma vector = TIMER0_COMPA_vect
__interrupt void real_time(void)          // interrupt on compare match {
```

Using EEPROM Memory

The AT90 contains 4K bytes of data EEPROM memory. It is organized as a separate data space, in which single bytes can be read and written. Access between the EEPROM and the CPU, is performed by programming the EEPROM Address Registers, the EEPROM Data Register, and the EEPROM Control Register which are in the I/O space. This is more like communicating with an I/O device than with data memory RAM.

EEPROM Address Register –



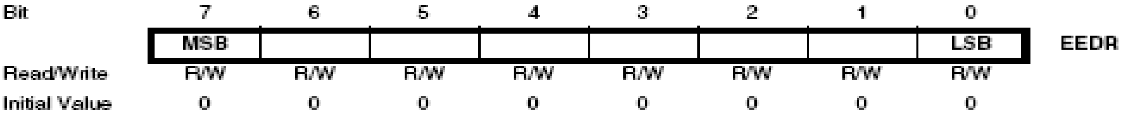
Bit	15	14	13	12	11	10	9	8	
	–	–	–	–	EEAR11	EEAR10	EEAR9	EEAR8	EEARH
	EEAR7	EEAR6	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	EEARL
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	X	X	X	X	
	X	X	X	X	X	X	X	X	

- **Bits 15..12 – Res: Reserved Bits**

These are reserved bits and will always read as zero. When writing to this address location, write these bits to zero for compatibility with future devices.

• Bits 11.-.0 – EEAR11.-.0: EEPROM Address
The EEPROM Address Registers – EEARH and EEARL – specify the EEPROM address in the 4K bytes EEPROM space. The EEPROM data bytes are addressed linearly between 0 and 4096. The initial value of EEAR is undefined. A proper value must be written before the EEPROM may be accessed.

EEPROM Data Register –

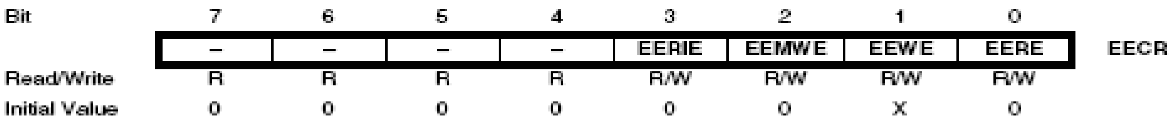


Bit	7	6	5	4	3	2	1	0	
	MSB							LSB	EEDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **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.

EEPROM Control Register –



Bit	7	6	5	4	3	2	1	0	
	–	–	–	–	EERIE	EEMWE	EEWE	EERE	EECR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	X	0	

- **Bits 7..4 – Res: Reserved Bits**

• Bit 3 – EERIE: EEPROM Ready Interrupt Enable
Writing EERIE to one enables the EEPROM Ready Interrupt if the I-bit in SREG is set. Writing EERIE to zero disables the interrupt. The EEPROM Ready interrupt generates a constant interrupt when EEWE is cleared.

• Bit 2 – EEMWE: EEPROM Master Write Enable
The EEMWE bit determines whether setting EEWE to one causes the EEPROM to be written. When EEMWE is written to one, writing EEWE to one within four clock cycles will write data to the EEPROM at the selected address. If EEMWE is zero, writing EEWE to one will have no effect. When EEMWE has been written to one by software, hardware clears the bit to zero after four clock cycles. See the description of the EEWE bit for an EEPROM write procedure.

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. The EEMWE bit must be set when the logical one is written to EEWE, otherwise no EEPROM write takes place. The following procedure should be followed when writing the EEPROM (the order of steps 3 and 4 is not essential):

1. Wait until EEWE becomes zero.
2. Wait until SPMEN in SPMCSR becomes zero.
3. Write new EEPROM address to EEAR (optional).
4. Write new EEPROM data to EEDR (optional).
5. Write a logical one to the EEMWE bit while writing a zero to EEWE in EECR.
6. Within four clock cycles after setting EEMWE, write a logical one to EEWE.

Caution: An interrupt between step 5 and step 6 will make the write cycle fail, since the EEPROM Master Write Enable will time-out. If an interrupt routine accessing the EEPROM is interrupting another EEPROM access, the EEAR or EEDR Register will be modified, causing the interrupted EEPROM access to fail. It is recommended to have the global interrupt flag cleared during the four last steps to avoid these problems. When the write access time has elapsed, the EEWE bit is cleared 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 written to a logic one to trigger the EEPROM read. The EEPROM read access takes one instruction, and the requested data is available immediately. When the EEPROM is read, the CPU is halted for four cycles before the next instruction is executed. The user should poll the EEWE bit before starting the read operation. If a write operation is in progress, it is neither possible to read the EEPROM, nor to change the EEAR Register.

Connection Summary for AUT Microcontroller Applications Board

	Alternate	*Parallel port*	
	Function	*Function*	
Switches, Buttons and Keypad			
I/P		PA0-7	8x toggle switches - PINA
I/P		PA0-7	8x push buttons (normally high) - PINA
I/P		PA0-7	3x4 keypad - PINA
O/P		PE2	Selects active device on PINA
O/P		PE3	Only one device active at a time
LEDs and Seven Segment Displays			
O/P		PC0-7	8x LEDs – display binary value of PORTC
O/P		PC0-7	2x hexadecimal 7 segment displays - PORTC
Analog Inputs – Temperature, light, humidity, microphone, potentiometers			
I/P	ADC0	PF0	Light sensor (LDR)
I/P	ADC1	PF1	Potentiometer (VR1) 0-5V
I/P	ADC1	PF1	Relative Humidity Sensor
I/P	ADC2	PF2	Potentiometer (VR2) 0-5V
I/P	ADC2	PF2	Microphone
I/P	ADC3	PF3	Temperature sensor
Miscellaneous Inputs			
I/P	IC3, T2	PE7, PD7	Optical Interrupter on fan
I/P	IC1, T1	PD4, PD6	Timer 1 – input capture and external input
I/P	INT0	PD0	Motion sensor – external interrupt
Miscellaneous Outputs			
O/P	USART1	PD2,3	LCD Display - 2 rows of 24 characters
O/P	OC1A	PB5	Motor with fan (variable speed with PWM)
O/P		PG0	Motor direction
O/P		PG1	Heater resistor
O/P	OC1B	PB6	Incandescent lamp
O/P	OC1C,OC2	PB7	Speaker
O/P		PD1	Piezo buzzer

PE3	PE2
0	0
0	1
1	0

ADC1 switch selects between these

ADC2 switch selects between these

Notes

1. When the alternate function of an I/O pin is used it is still necessary to set the direction, input or output, by storing a value in the corresponding bit or bits of the data Direction register.
   For example, when using output compare pin (OC1B) to drive the incandescent lamp initialise with DDRB.6 = 1.
   Because the default value in all data direction registers is zero alternate function inputs such as ADC will usually work without DDR initialisation to zero but it is good practice to do so.

2. Bit addressing cannot be used with registers PING, PORTG and DDRG. The method shown at the bottom of page 1 must be used for initialising and operating the heater and the motor direction relay.