Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	AVR
Core Size	8-Bit
Speed	20MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	32
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Through Hole
Package / Case	40-DIP (0.600", 15.24mm)
Supplier Device Package	40-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atmega644p-20pq

Table of Contents

Introduction....1

Feature....1

1. Description....10

2. Configuration Summary....11

3. Ordering Information....12

4. Block Diagram....14

5. Pin Configurations....15

   5.1. Pinout....15
   5.2. Pin Descriptions....16

6. I/O Multiplexing....18

7. General Information....20

   7.1. Resources....20
   7.2. Data Retention....20
   7.3. About Code Examples....20
   7.4. Capacitive Touch Sensing....20

8. AVR CPU Core....21

   8.1. Overview....21
   8.2. ALU – Arithmetic Logic Unit....22
   8.3. Status Register....22
   8.4. General Purpose Register File....24
   8.5. Stack Pointer....25
   8.6. Accessing 16-bit Registers....27
   8.7. Instruction Execution Timing....28
   8.8. Reset and Interrupt Handling....28

9. AVR Memories....31

   9.1. Overview....31
   9.2. In-System Reprogrammable Flash Program Memory....31
   9.3. SRAM Data Memory....32
   9.4. EEPROM Data Memory....33
   9.5. I/O Memory....34
   9.6. Register Description....34

10. System Clock and Clock Options....43

   10.1. Clock Systems and Their Distribution....43
   10.2. Clock Sources....44
   10.3. Low Power Crystal Oscillator....46

8.5.2. Extended Z-pointer Register for ELPM/SPM

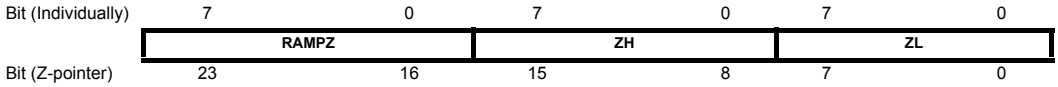
When using the I/O specific commands IN and OUT, the I/O addresses 0x00 - 0x3F must be used. When addressing I/O Registers as data space using LD and ST instructions, 0x20 must be added to these offset addresses. The device is a complex microcontroller with more peripheral units than can be supported within the 64 locations reserved in Opcode for the IN and OUT instructions. For the Extended I/O space from 0x60 in SRAM, only the ST/STS/STD and LD/LDS/LDD instructions can be used.

Name: RAMPZ
Offset: 0x5B
Reset: 0x0
Property: When addressing I/O Registers as data space the offset address is 0x3B

Bit	7	6	5	4	3	2	1	0
	RAMPZ7	RAMPZ6	RAMPZ5	RAMPZ4	RAMPZ3	RAMPZ2	RAMPZ1	RAMPZ0
Access	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0

Bits 0, 1, 2, 3, 4, 5, 6, 7 – RAMPZn: Extended Z-pointer Register for ELPM/SPM
For ELPM/SPM instructions, the Z-pointer is a concatenation of RAMPZ, ZH, and ZL, as shown in the below figure. Note that LPM is not affected by the RAMPZ setting.

Figure 8-4. The Z-pointer used by ELPM and SPM



The actual number of bits is implementation dependent. Unused bits in an implementation will always read as zero. For compatibility with future devices, be sure to write these bits to zero.

8.6. Accessing 16-bit Registers

The AVR data bus is 8 bits wide, and so accessing 16-bit registers requires atomic operations. These registers must be byte-accessed using two read or write operations. 16-bit registers are connected to the 8-bit bus and a temporary register using a 16-bit bus.

For a write operation, the low byte of the 16-bit register must be written before the high byte. The low byte is then written into the temporary register. When the high byte of the 16-bit register is written, the temporary register is copied into the low byte of the 16-bit register in the same clock cycle.

For a read operation, the low byte of the 16-bit register must be read before the high byte. When the low byte register is read by the CPU, the high byte of the 16-bit register is copied into the temporary register in the same clock cycle as the low byte is read. When the high byte is read, it is then read from the temporary register.

This ensures that the low and high bytes of 16-bit registers are always accessed simultaneously when reading or writing the register.

Interrupts can corrupt the timed sequence if an interrupt is triggered and accesses the same 16-bit register during an atomic 16-bit read/write operation. To prevent this, interrupts can be disabled when writing or reading 16-bit registers.

The temporary registers can also be read and written directly from user software.

Note:

1. The 128kHz oscillator is a very low power clock source, and is not designed for high accuracy.

When this clock source is selected, start-up times are determined by the SUT Fuses:

Table 10-13. Start-Up Times for the 128kHz Internal Oscillator

Power Conditions	Start-Up Time from Power-down and Power-save	Additional Delay from Reset	SUT[1:0]
BOD enabled	6 CK	14CK	00
Fast rising power	6 CK	14CK + 4ms	01
Slowly rising power	6 CK	14CK + 64ms	10
Reserved			11

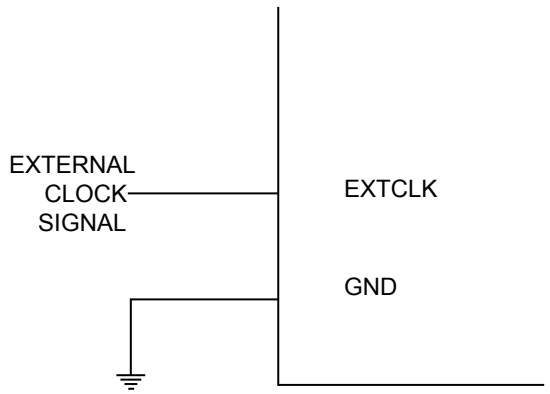
10.8. External Clock

To drive the device from an external clock source, EXTCLK should be driven as shown in the Figure below. To run the device on an external clock, the CKSEL Fuses must be programmed to '0000':

Table 10-14. External Clock Frequency

Frequency	CKSEL[3:0]
0 - 20MHz	0000

Figure 10-3. External Clock Drive Configuration



When this clock source is selected, start-up times are determined by the SUT Fuses:

Table 10-15. Start-Up Times for the External Clock Selection - SUT

Power Conditions	Start-Up Time from Power-down and Power-save	Additional Delay from Reset (V_{CC} = 5.0V)	SUT[1:0]
BOD enabled	6 CK	14CK	00
Fast rising power	6 CK	14CK + 4.1ms	01
Slowly rising power	6 CK	14CK + 65ms	10
Reserved			11

- Brown-out Reset
- 2-wire Serial Interface address match
- Timer/Counter2 interrupt
- SPM/EEPROM ready interrupt
- External level interrupt on INT
- Pin change interrupt

Note: 1. Timer/Counter2 will only keep running in asynchronous mode.

Related Links

8-bit Timer/Counter2 with PWM and Asynchronous Operation on page 189

11.6. Power-Down Mode

When the SM[2:0] bits are written to '010', the SLEEP instruction makes the MCU enter Power-Down mode. In this mode, the external Oscillator is stopped, while the external interrupts, the 2-wire Serial Interface address watch, and the Watchdog continue operating (if enabled).

Only one of these events can wake up the MCU:

- External Reset
- Watchdog System Reset
- Watchdog Interrupt
- Brown-out Reset
- 2-wire Serial Interface address match
- External level interrupt on INT
- Pin change interrupt

This sleep mode basically halts all generated clocks, allowing operation of asynchronous modules only.

Note: If a level triggered interrupt is used for wake-up from Power-Down, the required level must be held long enough for the MCU to complete the wake-up to trigger the level interrupt. If the level disappears before the end of the Start-up Time, the MCU will still wake up, but no interrupt will be generated. The start-up time is defined by the SUT and CKSEL Fuses.

When waking up from Power-Down mode, there is a delay from the wake-up condition occurs until the wake-up becomes effective. This allows the clock to restart and become stable after having been stopped. The wake-up period is defined by the same CKSEL Fuses that define the Reset Time-out period.

Related Links

Clock Sources on page 44
EXINT - External Interrupts on page 84

11.7. Power-save Mode

When the SM[2:0] bits are written to 011, the SLEEP instruction makes the MCU enter Power-save mode. This mode is identical to Power-down, with one exception:

If Timer/Counter2 is enabled, it will keep running during sleep. The device can wake up from either Timer Overflow or Output Compare event from Timer/Counter2 if the corresponding Timer/Counter2 interrupt enable bits are set in TIMSK2, and the Global Interrupt Enable bit in SREG is set.

If Timer/Counter2 is not running, Power-down mode is recommended instead of Power-save mode.

- OC0A: Output Compare Match A output. The PB3 pin can serve as an external output for the Timer/Counter0 Output Compare. The pin has to be configured as an output (DDB3 set “1”) to serve this function. The OC0A pin is also the output pin for the PWM mode timer function.
- PCINT11: Pin Change Interrupt source 11. The PB3 pin can serve as an external interrupt source.

- AIN0/INT2/PCINT10 – Port B, Bit 2
  - AIN0: Analog Comparator Positive input. This pin is directly connected to the positive input of the Analog Comparator.
  - INT2: External Interrupt source 2. The PB2 pin can serve as an External Interrupt source to the MCU.
  - PCINT10: Pin Change Interrupt source 10. The PB2 pin can serve as an external interrupt source.
- T1/CLKO/PCINT9 – Port B, Bit 1
  - T1: Timer/Counter1 counter source.
  - CLKO: Divided System Clock: The divided system clock can be output on the PB1 pin. The divided system clock will be output if the CKOUT Fuse is programmed, regardless of the PORTB1 and DDB1 settings. It will also be output during reset.
  - PCINT9: Pin Change Interrupt source 9. The PB1 pin can serve as an external interrupt source.
- T0/XCK0/PCINT8 – Port B, Bit 0
  - T0: Timer/Counter0 counter source.
  - XCK0: USART0 External clock. The Data Direction Register (DDB0) controls whether the clock is output (DDB0 set “1”) or input (DDB0 cleared). The XCK0 pin is active only when the USART0 operates in Synchronous mode.
  - PCINT8: Pin Change Interrupt source 8. The PB0 pin can serve as an external interrupt source.

Table 15-7 and Table 15-8 relate the alternate functions of Port B to the overriding signals shown in Figure 15-5. SPI MSTR INPUT and SPI SLAVE OUTPUT constitute the MISO signal, while MOSI is divided into SPI MSTR OUTPUT and SPI SLAVE INPUT.

Table 15-7. Overriding Signals for Alternate Functions in PB[7:4]

Signal Name	PB7/SCK/PCINT15	PB6/MISO/PCINT14	PB5/MOSI/PCINT13	PB4/$\overline{SS}$/OC0B/PCINT12
PUOE	SPE • $\overline{MSTR}$	SPE • MSTR	SPE • $\overline{MSTR}$	SPE • $\overline{MSTR}$
PUOV	PORTB7 • $\overline{PUD}$	PORTB6 • $\overline{PUD}$	PORTB5 • $\overline{PUD}$	PORTB4 • $\overline{PUD}$
DDOE	SPE • $\overline{MSTR}$	SPE • MSTR	SPE • $\overline{MSTR}$	SPE • $\overline{MSTR}$
DDOV	0	0	0	0
PVOE	SPE • MSTR	SPE • $\overline{MSTR}$	SPE • MSTR	OC0B ENABLE
PVOV	SCK OUTPUT	SPI SLAVE OUTPUT	SPI MSTR OUTPUT	OC0B
DIEOE	PCINT15 • PCIE1	PCINT14 • PCIE1	PCINT13 • PCIE1	PCINT12 • PCIE1
DIEOV	1	1	1	1

15.4.5. Port B Data Register

When addressing I/O Registers as data space using LD and ST instructions, the provided offset must be used. When using the I/O specific commands IN and OUT, the offset is reduced by 0x20, resulting in an I/O address offset within 0x00 - 0x3F.

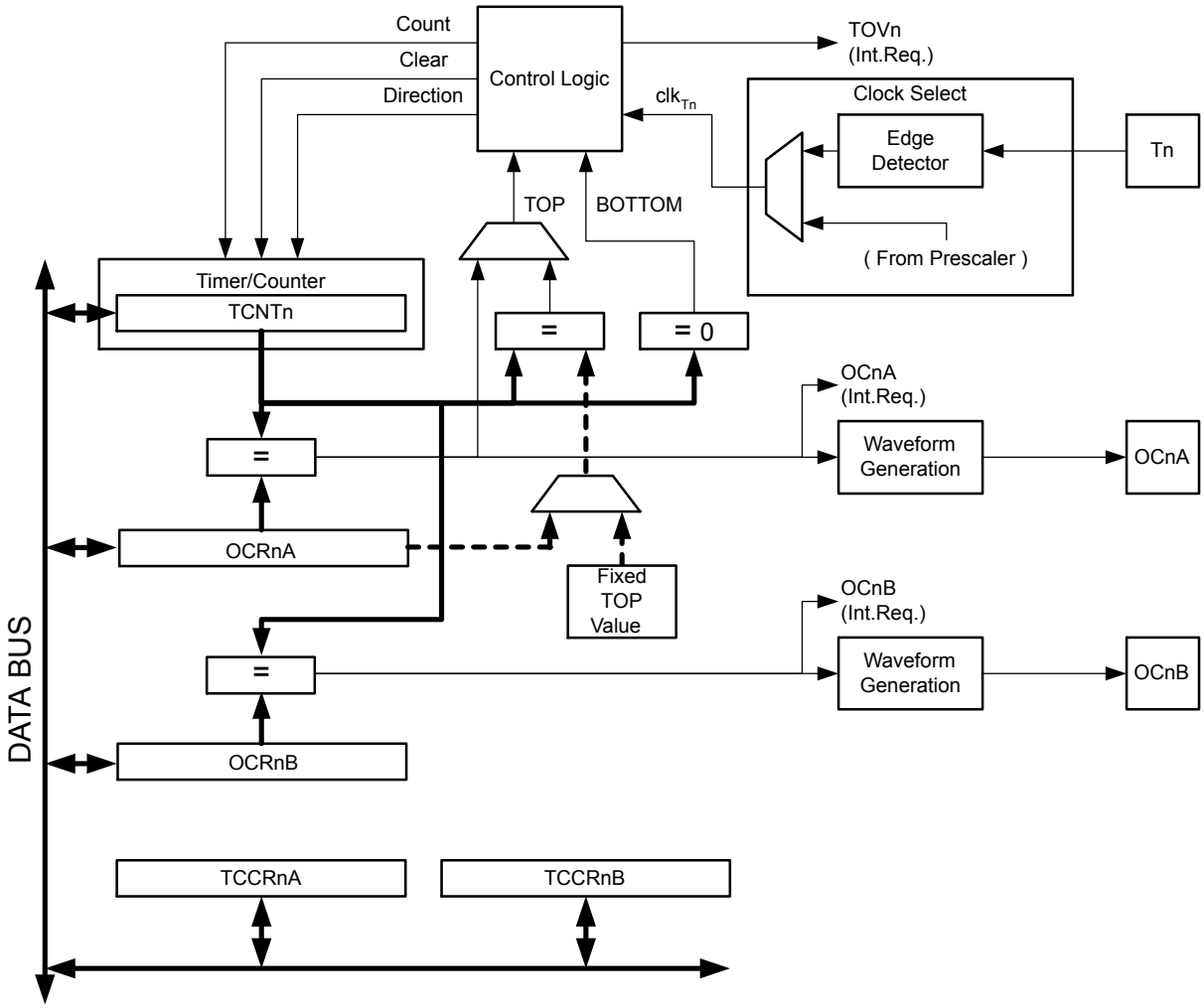
The device is a complex microcontroller with more peripheral units than can be supported within the 64 locations reserved in Opcode for the IN and OUT instructions. For the Extended I/O space from 0x60 in SRAM, only the ST/STS/STD and LD/LDS/LDD instructions can be used.

Name: PORTB
Offset: 0x25
Reset: 0x00
Property: When addressing as I/O Register: address offset is 0x05

Bit	7	6	5	4	3	2	1	0
	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 7:0 – PORTBn: Port B Data [n = 0:7]

Figure 16-1. 8-bit Timer/Counter Block Diagram



16.2.1. Definitions

Many register and bit references in this section are written in general form:

- n=0 represents the Timer/Counter number
- x=A,B represents the Output Compare Unit A or B

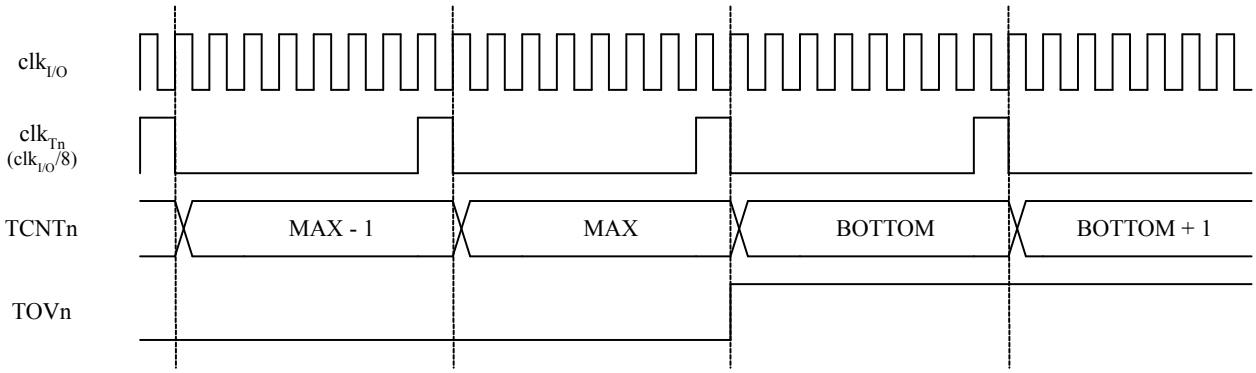
However, when using the register or bit definitions in a program, the precise form must be used, i.e., TCNT0 for accessing Timer/Counter0 counter value.

The following definitions are used throughout the section:

Table 16-1. Definitions

Constant	Description
BOTTOM	The counter reaches the BOTTOM when it becomes zero (0x00 for 8-bit counters, or 0x0000 for 16-bit counters).

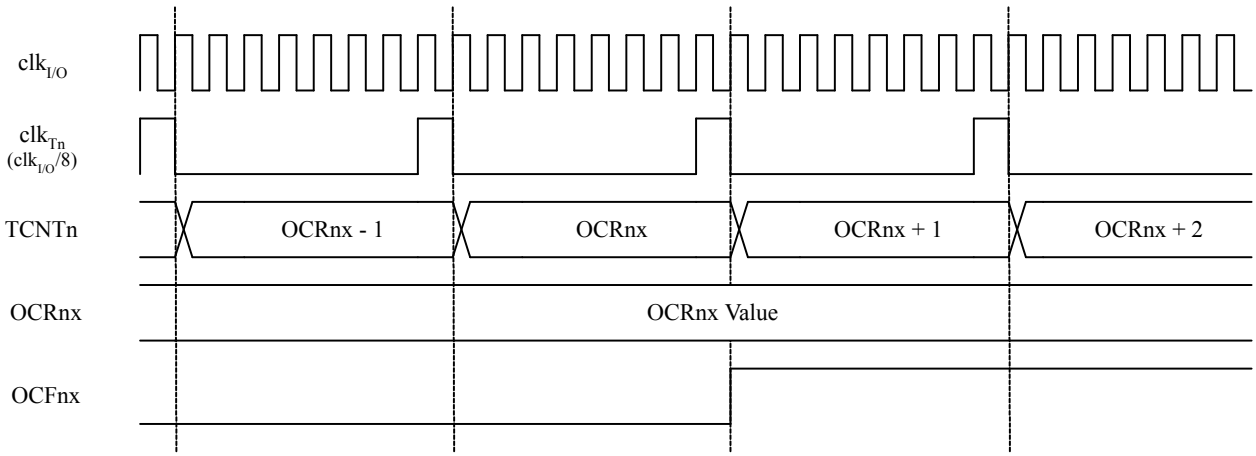
Figure 16-9. Timer/Counter Timing Diagram, with Prescaler ($f_{clk\_I/O}/8$)



Note: The “n” in the register and bit names indicates the device number (n = 0 for Timer/Counter 0), and the “x” indicates Output Compare unit (A/B).

The next figure shows the setting of OCF0B in all modes and OCF0A in all modes (except CTC mode and PWM mode where OCR0A is TOP).

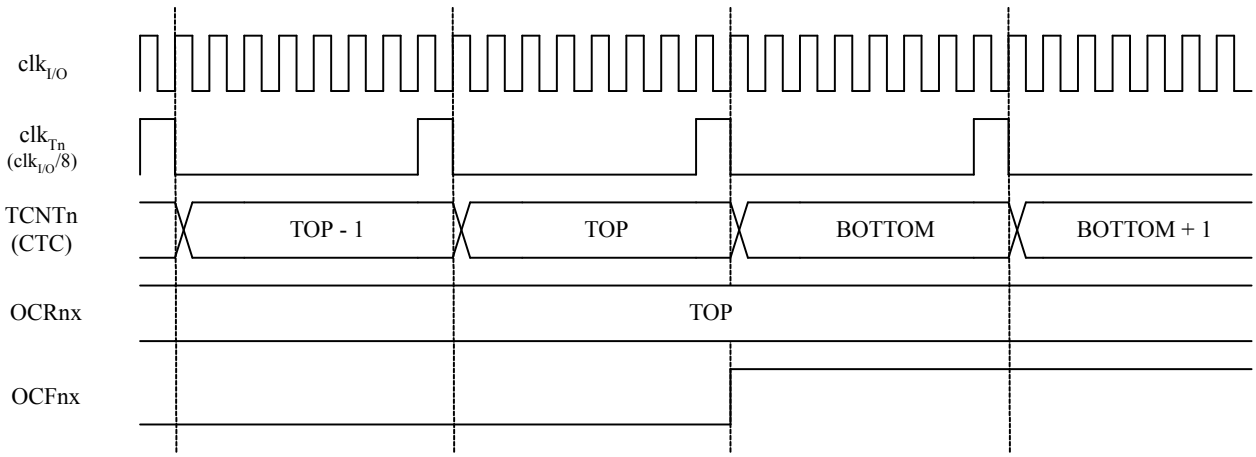
Figure 16-10. Timer/Counter Timing Diagram, Setting of OCF0x, with Prescaler ($f_{clk\_I/O}/8$)



Note: The “n” in the register and bit names indicates the device number (n = 0 for Timer/Counter 0), and the “x” indicates Output Compare unit (A/B).

The next figure shows the setting of OCF0A and the clearing of TCNT0 in CTC mode and fast PWM mode where OCR0A is TOP.

Figure 16-11. Timer/Counter Timing Diagram, Clear Timer on Compare Match mode, with Prescaler ($f_{clk\_I/O}/8$)



17.14.9. TC1 Interrupt Flag Register

When addressing I/O Registers as data space using LD and ST instructions, the provided offset must be used. When using the I/O specific commands IN and OUT, the offset is reduced by 0x20, resulting in an I/O address offset within 0x00 - 0x3F.

The device is a complex microcontroller with more peripheral units than can be supported within the 64 locations reserved in Opcode for the IN and OUT instructions. For the Extended I/O space from 0x60 in SRAM, only the ST/STS/STD and LD/LDS/LDD instructions can be used.

Name: TIFR1
Offset: 0x36
Reset: 0x00
Property: When addressing as I/O Register: address offset is 0x16

Bit	7	6	5	4	3	2	1	0
			ICF			OCFB	OCFA	TOV
Access			R/W			R/W	R/W	R/W
Reset			0			0	0	0

Bit 5 – ICF: Timer/Counter1, Input Capture Flag
This flag is set when a capture event occurs on the ICP1 pin. When the Input Capture Register (ICR1) is set by the WGM1[3:0] to be used as the TOP value, the ICF Flag is set when the counter reaches the TOP value.

ICF is automatically cleared when the Input Capture Interrupt Vector is executed. Alternatively, ICF can be cleared by writing a logic one to its bit location.

Bit 2 – OCFB: Timer/Counter1, Output Compare B Match Flag
This flag is set in the timer clock cycle after the counter (TCNT1) value matches the Output Compare Register B (OCR1B).

Note that a Forced Output Compare (FOCB) strobe will not set the OCF1B Flag.

OCFB is automatically cleared when the Output Compare Match B Interrupt Vector is executed. Alternatively, OCF1B can be cleared by writing a logic one to its bit location.

Bit 1 – OCFA: Timer/Counter1, Output Compare A Match Flag
This flag is set in the timer clock cycle after the counter (TCNT1) value matches the Output Compare Register A (OCR1A).

Note that a Forced Output Compare (FOCA) strobe will not set the OCF1A Flag.

OCFA is automatically cleared when the Output Compare Match A Interrupt Vector is executed. Alternatively, OCF1A can be cleared by writing a logic one to its bit location.

Bit 0 – TOV: Timer/Counter1, Overflow Flag
The setting of this flag is dependent of the WGM1[3:0] bits setting. In Normal and CTC modes, the TOV1 Flag is set when the timer overflows. Refer to the Waveform Generation Mode bit description for the TOV Flag behavior when using another WGM1[3:0] bit setting.

TOV1 is automatically cleared when the Timer/Counter 1 Overflow Interrupt Vector is executed. Alternatively, TOV1 can be cleared by writing a logic one to its bit location.

19.11.3. TC2 Counter Value Register

Name: TCNT2
Offset: 0xB2
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	TCNT2[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 7:0 – TCNT2[7:0]: Timer/Counter 2 Counter Value
The Timer/Counter Register gives direct access, both for read and write operations, to the Timer/Counter unit 8-bit counter. Writing to the TCNT2 Register blocks (removes) the Compare Match on the following timer clock. Modifying the counter (TCNT2) while the counter is running, introduces a risk of missing a Compare Match between TCNT2 and the OCR2x Registers.

19.11.4. TC2 Output Compare Register A

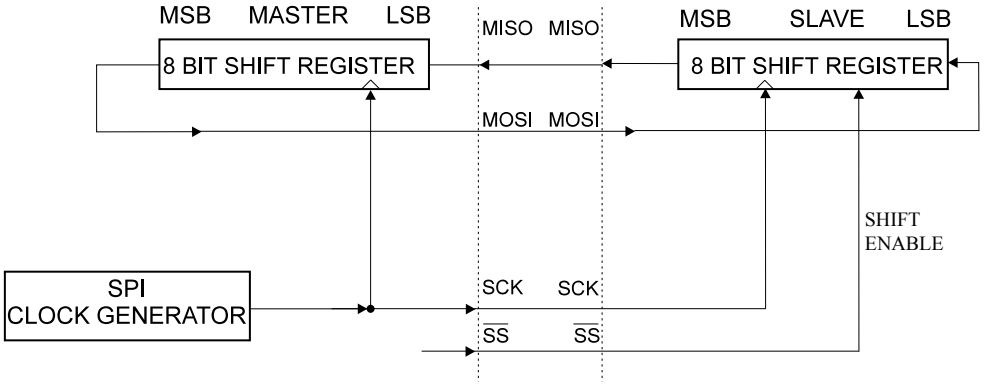
Name: OCR2A
Offset: 0xB3
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	OCR2A[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 7:0 – OCR2A[7:0]: Output Compare 2 A
The Output Compare Register A contains an 8-bit value that is continuously compared with the counter value (TCNT2). A match can be used to generate an Output Compare interrupt, or to generate a waveform output on the OC2A pin.

data to be sent into SPDR before reading the incoming data. The last incoming byte will be kept in the Buffer Register for later use.

Figure 20-2. SPI Master-slave Interconnection



The system is single buffered in the transmit direction and double buffered in the receive direction. This means that bytes to be transmitted cannot be written to the SPI Data Register before the entire shift cycle is completed. When receiving data, however, a received character must be read from the SPI Data Register before the next character has been completely shifted in. Otherwise, the first byte is lost.

In SPI Slave mode, the control logic will sample the incoming signal of the SCK pin. To ensure correct sampling of the clock signal, the minimum low and high periods should be longer than two CPU clock cycles.

When the SPI is enabled, the data direction of the MOSI, MISO, SCK, and $\overline{SS}$ pins is overridden according to the table below. For more details on automatic port overrides, refer to the IO Port description.

Table 20-1. SPI Pin Overrides

Pin	Direction, Master SPI	Direction, Slave SPI
MOSI	User Defined	Input
MISO	Input	User Defined
SCK	User Defined	Input
$\overline{SS}$	User Defined	Input

Note: 1. See the IO Port description for how to define the SPI pin directions.

The following code examples show how to initialize the SPI as a Master and how to perform a simple transmission. `DDR_SPI` in the examples must be replaced by the actual Data Direction Register controlling the SPI pins. `DD_MOSI`, `DD_MISO` and `DD_SCK` must be replaced by the actual data direction bits for these pins. E.g. if MOSI is placed on pin PB5, replace `DD_MOSI` with `DDB5` and `DDR_SPI` with `DDRB`.

Assembly Code Example

```
SPI_MasterInit:
   ; Set MOSI and SCK output, all others input
   ldi     r17,(1<<DD_MOSI)|(1<<DD_SCK)
   out     DDR_SPI,r17
   ; Enable SPI, Master, set clock rate fck/16
   ldi     r17,(1<<SPE)|(1<<MSTR)|(1<<SPR0)
   out     SPCR,r17
   ret
```

21.12.1. USART I/O Data Register n

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 UDRn. The Transmit Data Buffer Register (TXB) will be the destination for data written to the UDR1 Register location. Reading the UDRn Register location will return the contents of the Receive Data Buffer Register (RXB).

For 5-, 6-, or 7-bit characters the upper unused bits will be ignored by the Transmitter and set to zero by the Receiver.

The transmit buffer can only be written when the UDRE Flag in the UCSRnA Register is set. Data written to UDRn when the UCSRnA.UDRE Flag is not set, will be ignored by the USART Transmitter n. 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 TxDn pin.

The receive buffer consists of a two level FIFO. The FIFO will change its state whenever the receive buffer is accessed. Due to this behavior of the receive buffer, do not use Read-Modify-Write instructions (SBI and CBI) on this location. Be careful when using bit test instructions (SBIC and SBIS), since these also will change the state of the FIFO.

Name: UDRn
Offset: 0xC6 + n*0x08 [n=0..1]
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	TXB / RXB[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 7:0 – TXB / RXB[7:0]: USART Transmit / Receive Data Buffer

Table 24-1. Analog Comparator Multiplexed Input

ACME	ADEN	MUX[2:0]	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

24.3. Register Description

28. MEMPROG- Memory Programming

28.1. Program And Data Memory Lock Bits

The devices provides Lock bits. These can be left unprogrammed ('1') or can be programmed ('0') to obtain the additional features listed in Table. Lock Bit Protection Modes in this section. The Lock bits can only be erased to “1” with the Chip Erase command.

Table 28-1. Lock Bit Byte(1)

Lock Bit Byte	Bit No.	Description	Default Value
	7	–	1 (unprogrammed)
	6	–	1 (unprogrammed)
BLB12	5	Boot Lock bit	1 (unprogrammed)
BLB11	4	Boot Lock bit	1 (unprogrammed)
BLB02	3	Boot Lock bit	1 (unprogrammed)
BLB01	2	Boot Lock bit	1 (unprogrammed)
LB2	1	Lock bit	1 (unprogrammed)
LB1	0	Lock bit	1 (unprogrammed)

Note:

1. '1' means unprogrammed, '0' means programmed.

Table 28-2. Lock Bit Protection Modes(1)(2)

Memory Lock Bits			Protection Type
LB Mode	LB2	LB1	
1	1	1	No memory lock features enabled.
2	1	0	Further programming of the Flash and EEPROM is disabled in Parallel and Serial Programming mode. The Fuse bits are locked in both Serial and Parallel Programming mode.(1)
3	0	0	Further programming and verification of the Flash and EEPROM is disabled in Parallel and Serial Programming mode. The Boot Lock bits and Fuse bits are locked in both Serial and Parallel Programming mode.(1)

Note:

1. Program the Fuse bits and Boot Lock bits before programming the LB1 and LB2.
2. '1' means unprogrammed, '0' means programmed.

High Fuse Byte	Bit No.	Description	Default Value
EESAVE	3	EEPROM memory is preserved through the Chip Erase	1 (unprogrammed), EEPROM not reserved
BODLEVEL2(4)	2	Brown-out Detector trigger level	1 (unprogrammed)
BODLEVEL1(4)	1	Brown-out Detector trigger level	1 (unprogrammed)
BODLEVEL0(4)	0	Brown-out Detector trigger level	1 (unprogrammed)

Note:

1. Please refer to *Alternate Functions of Port C* in I/O-Ports chapter for description of RSTDISBL Fuse.
2. The SPIEN Fuse is not accessible in serial programming mode.
3. Please refer to *WDTCSR – Watchdog Timer Control Register* for details.
4. Please refer to Table BODLEVEL Fuse Coding in *System and Reset Characteristics* for BODLEVEL Fuse decoding.

Table 28-8. Fuse Low Byte

Low Fuse Byte	Bit No.	Description	Default Value
CKDIV8(4)	7	Divide clock by 8	0 (programmed)
CKOUT(3)	6	Clock output	1 (unprogrammed)
SUT1	5	Select start-up time	1 (unprogrammed)(1)
SUT0	4	Select start-up time	0 (programmed)(1)
CKSEL3	3	Select Clock source	0 (programmed)(2)
CKSEL2	2	Select Clock source	0 (programmed)(2)
CKSEL1	1	Select Clock source	1 (unprogrammed)(2)
CKSEL0	0	Select Clock source	0 (programmed)(2)

Note:

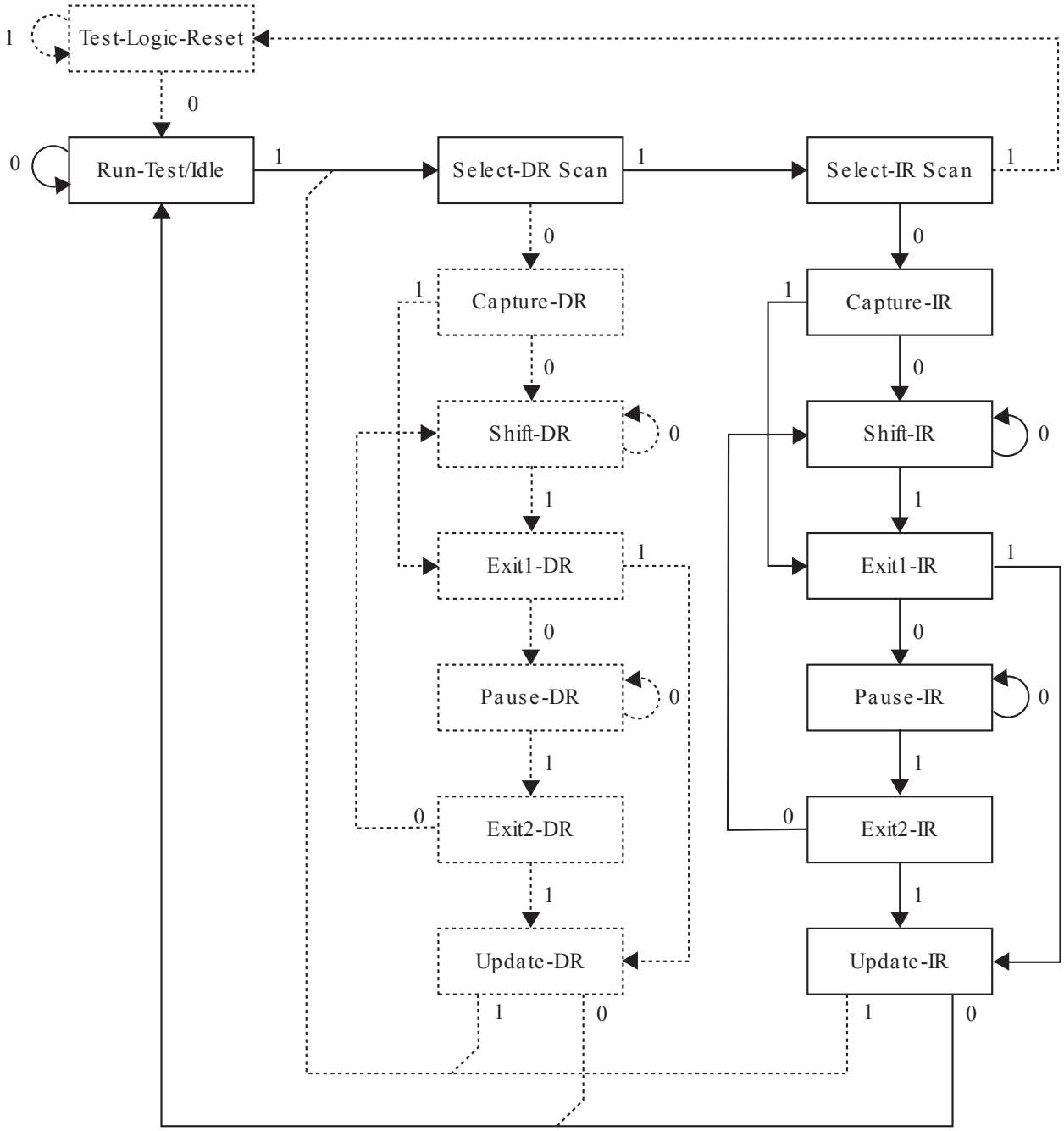
1. The default value of SUT[1:0] results in maximum start-up time for the default clock source. See Table. Start-up times for the internal calibrated RC Oscillator clock selection in *Calibrated Internal RC Oscillator* of System Clock and Clock Options chapter for details.
2. The default setting of CKSEL[3:0] results in internal RC Oscillator @ 8MHz. See Table 'Internal Calibrated RC Oscillator Operating Modes' in *Calibrated Internal RC Oscillator* of the System Clock and Clock Options chapter for details.
3. The CKOUT Fuse allows the system clock to be output on PORTB0. Please refer to *Clock Output Buffer* section in the System Clock and Clock Options chapter for details.
4. Please refer to *System Clock Prescaler* section in the System Clock and Clock Options chapter for details.

The status of the Fuse bits is not affected by Chip Erase. Note that the Fuse bits are locked if Lock bit1 (LB1) is programmed. Program the Fuse bits before programming the Lock bits.

Related Links

Calibration Accuracy of Internal RC Oscillator on page 400
System and Reset Characteristics on page 401

Figure 28-9. State Machine Sequence for Changing the Instruction Word



28.10.2. AVR_RESET (0xC)

The AVR specific public JTAG instruction for setting the AVR device in the Reset mode or taking the device out from the Reset mode. The TAP controller is not reset by this instruction. The one bit Reset Register is selected as Data Register. Note that the reset will be active as long as there is a logic 'one' in the Reset Chain. The output from this chain is not latched.

The active states are:

- Shift-DR: The Reset Register is shifted by the TCK input.

4.2. The sum of all I_{OH}, for ports PA0-PA3, PC0-PC7 should not exceed 100mA. If I_{OH} exceeds the test condition, V_{OH} may exceed the related specification. Pins are not guaranteed to source current greater than the listed test condition.

Related Links

Minimizing Power Consumption on page 60

29.2.1. Power Consumption

Table 29-3. ATmega644P DC Characteristics - T_A = -40°C to 85°C, V_{CC} = 1.8V to 5.5V (unless otherwise noted)

Symbol	Parameter	Condition	Min.	Typ.(2)	Max.	Units
I_{CC}	Power Supply Current(1)	Active 1MHz, V_{CC} = 2V	-	0.41	0.5	mA
		Active 4MHz, V_{CC} = 3V	-	2.0	2.7	
		Active 8MHz, V_{CC} = 5V	-	7.5	9.0	
		Idle 1MHz, V_{CC} = 2V	-	0.1	0.15	
		Idle 4MHz, V_{CC} = 3V	-	0.5	0.7	
		Idle 8MHz, V_{CC} = 5V	-	1.6	4.0	
	Power-save mode(3)	32 kHz TOSC enabled, V_{CC} = 1.8V	-	0.5	-	
		32 kHz TOSC enabled, V_{CC} = 3V	-	0.6	-	
	Power-down mode(3)	WDT enabled, V_{CC} = 3V	-	-	8.0	µA
		WDT disabled, V_{CC} = 3V	-	-	2.0	

Note:

1. All bits set in the ”PRR – Power Reduction Register ”.
2. Typical values at 25°C. Maximum values are test limits in production.
3. The current consumption values include input leakage current.

Table 29-4. ATmega644P DC Characteristics - T_A = -40°C to 105°C, V_{CC} = 1.8V to 5.5V (unless otherwise noted)

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
I_{CC}	Power Supply Current(1)	Active 1MHz, V_{CC} = 2V	-	-	0.7	mA
		Active 4MHz, V_{CC} = 3V	-	-	3.0	
		Active 8MHz, V_{CC} = 5V	-	-	11.0	
		Idle 1MHz, V_{CC} = 2V	-	-	0.2	
		Idle 4MHz, V_{CC} = 3V	-	-	0.85	
		Idle 8MHz, V_{CC} = 5V	-	-	6.0	
	Power-down mode(2)	WDT enabled, V_{CC} = 3V	-	-	20	µA
		WDT disabled, V_{CC} = 3V	-	-	10	

Note:

1. All bits set in the ”PRR – Power Reduction Register ”
2. The current consumption values include input leakage current.

PRR bit	Additional Current consumption compared to Active with external clock (See Figure 30-1 and Figure 30-2)	Additional Current consumption compared to Idle with external clock (See Figure 30-6 and Figure 30-7)
PRTIM2	3.3%	16.1%
PRTIM1	1.8%	8.9%
PRTIM0	0.9%	4.0%
PRADC	3.5%	17.2%
PRSPI	2.5%	12.3%

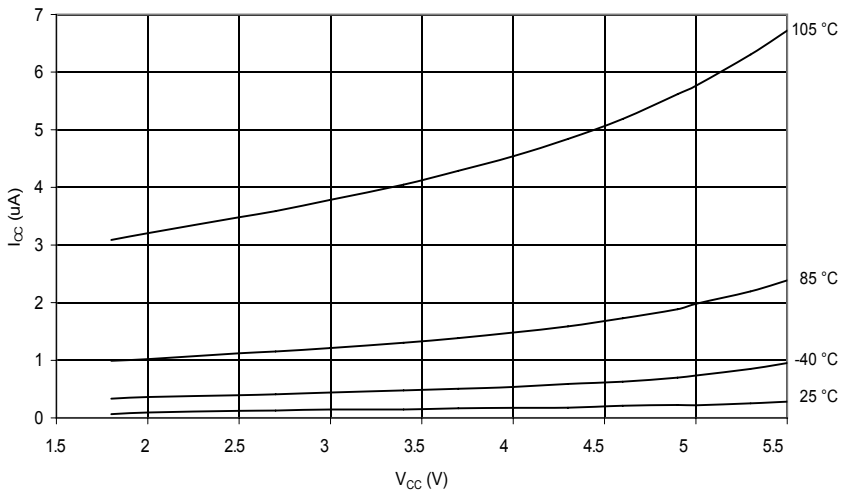
It is possible to calculate the typical current consumption based on the numbers from Table 30-2 for other V_{CC} and frequency settings than listed in Table 30-1.

Calculate the expected current consumption in idle mode with TIMER1, ADC, and SPI enabled at V_{CC} = 2.0V and F = 1MHz. From Table 30-2, third column, we see that we need to add 8.9% for the TIMER1, 17.2% for the ADC, and 12.3% for the SPI module. Reading from Figure 30-6, we find that the idle current consumption is ~0.1 mA at V_{CC} = 2.0V and F = 1MHz. The total current consumption in idle mode with TIMER1, ADC, and SPI enabled, gives:

$$I_{CCtotal} \approx 0.1mA \cdot (1 + 0.089 + 0.172 + 0.123) \approx 0.138mA$$

30.4. Power-down Supply Current

Figure 30-11. Power-down Supply Current vs. V_{CC} (Watchdog Timer Disabled)



Atmel | Enabling Unlimited Possibilities®

Atmel Corporation 1600 Technology Drive, San Jose, CA 95110 USA **T:** (+1)(408) 441.0311 **F:** (+1)(408) 436.4200 | **www.atmel.com**