



Welcome to [E-XFL.COM](https://www.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	10MHz
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	23
Program Memory Size	16KB (8K x 16)
Program Memory Type	FLASH
EEPROM Size	512 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	32-VFQFN Exposed Pad
Supplier Device Package	32-VQFN (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atmega168v-10mu

Table of Contents

Introduction.....	1
Feature.....	1
1. Description.....	10
2. Configuration Summary.....	11
3. Ordering Information	12
3.1. ATmega48/V.....	12
3.2. ATmega88/V.....	13
3.3. ATmega168/V.....	14
4. Block Diagram.....	15
5. Pin Configurations.....	16
5.1. Pin-out.....	16
5.2. Pin Descriptions.....	19
6. I/O Multiplexing.....	21
7. Resources.....	23
8. Data Retention.....	24
9. About Code Examples.....	25
10. Capacitive Touch Sensing.....	26
10.1. QTouch Library.....	26
11. AVR CPU Core.....	27
11.1. Overview.....	27
11.2. ALU – Arithmetic Logic Unit.....	28
11.3. Status Register.....	28
11.4. General Purpose Register File.....	30
11.5. Stack Pointer.....	31
11.6. Instruction Execution Timing.....	33
11.7. Reset and Interrupt Handling.....	34
12. AVR Memories.....	36
12.1. Overview.....	36
12.2. In-System Reprogrammable Flash Program Memory.....	36
12.3. SRAM Data Memory.....	37
12.4. EEPROM Data Memory.....	39
12.5. I/O Memory.....	40
12.6. Register Description.....	41

17.2.7. Pin Change Mask Register 1

Name: PCMSK1

Offset: 0x6C

Reset: 0x00

Property: -

Bit	7	6	5	4	3	2	1	0
		PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT8
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 0, 1, 2, 3, 4, 5, 6 – PCINT8, PCINT9, PCINT10, PCINT11, PCINT12, PCINT13, PCINT14: Pin Change Enable Mask

Each PCINT[15:8]-bit selects whether pin change interrupt is enabled on the corresponding I/O pin. If PCINT[15:8] is set and the PCIE1 bit in PCICR is set, pin change interrupt is enabled on the corresponding I/O pin. If PCINT[15:8] is cleared, pin change interrupt on the corresponding I/O pin is disabled.

18.4.2. 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.

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]

18.4.5. Port C 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.

Name: PORTC

Offset: 0x28

Reset: 0x00

Property: When addressing as I/O Register: address offset is 0x08

Bit	7	6	5	4	3	2	1	0
		PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 6:0 – PORTCn: Port C Data [n = 6:0]

The setup of the OC1x should be performed before setting the Data Direction Register for the port pin to output. The easiest way of setting the OC1x value is to use the Force Output Compare (FOC1x) strobe bits in Normal mode. The OC1x Registers keep their values even when changing between Waveform Generation modes.

Be aware that the TCCR1A.COM1x[1:0] bits are not double buffered together with the compare value. Changing the TCCR1A.COM1x[1:0] bits will take effect immediately.

19.6. Compare Match Output Unit

The Compare Output mode bits in the Timer/Counter Control Register A (TCCR0A.COM0x) have two functions:

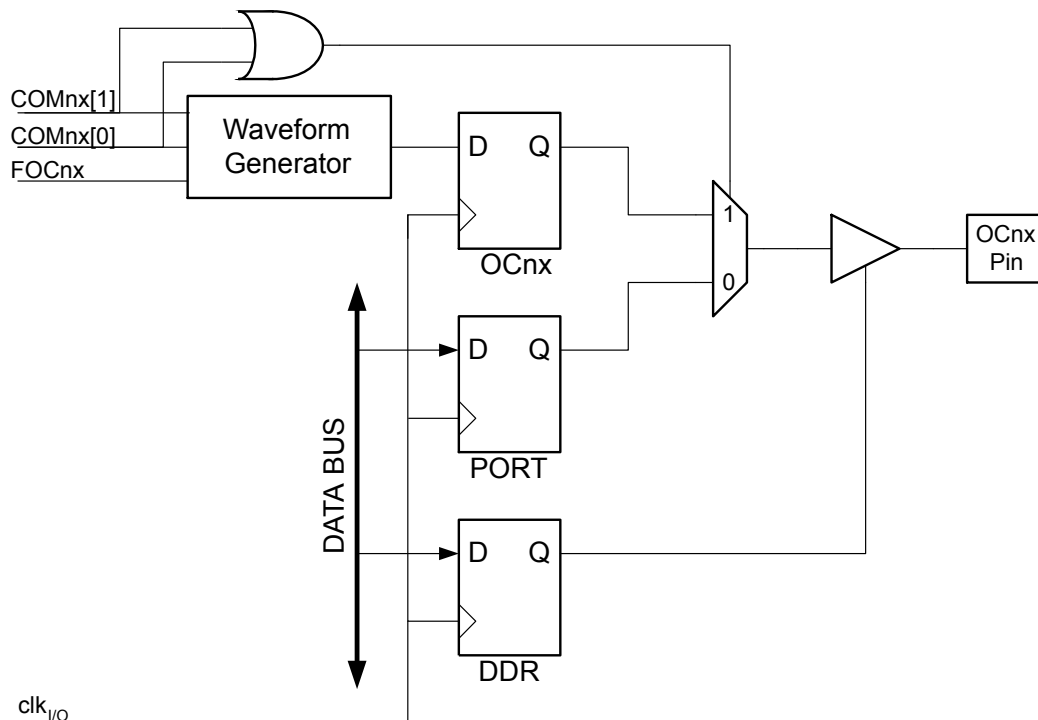
- The Waveform Generator uses the COM0x bits for defining the Output Compare (OC0x) register state at the next compare match.
- The COM0x bits control the OC0x pin output source

The figure below shows a simplified schematic of the logic affected by COM0x. The I/O Registers, I/O bits, and I/O pins in the figure are shown in bold. Only the parts of the general I/O port control registers that are affected by the COM0x bits are shown, namely PORT and DDR.

On system reset the OC0x Register is reset to 0x00.

Note: 'OC0x state' is always referring to internal OC0x *registers*, not the OC0x *pin*.

Figure 19-4. Compare Match Output Unit, Schematic



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 general I/O port function is overridden by the Output Compare (OC0x) from the Waveform Generator if either of the COM0x[1:0] bits are set. However, the OC0x pin direction (input or output) is still controlled by the Data Direction Register (DDR) for the port pin. In the Data Direction Register, the bit for the OC1x

19.9.1. TC0 Control Register A

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.

Name: TCCR0A

Offset: 0x44

Reset: 0x00

Property: When addressing as I/O Register: address offset is 0x24

Bit	7	6	5	4	3	2	1	0
	COM0A1	COM0A0	COM0B1	COM0B0			WGM01	WGM00
Access	R/W	R/W	R/W	R/W			R/W	R/W
Reset	0	0	0	0			0	0

Bits 7:6 – COM0An: Compare Output Mode for Channel A [n = 1:0]

These bits control the Output Compare pin (OC0A) behavior. If one or both of the COM0A[1:0] bits are set, the OC0A 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 OC0A pin must be set in order to enable the output driver.

When OC0A is connected to the pin, the function of the COM0A[1:0] bits depends on the WGM0[2:0] bit setting. The table below shows the COM0A[1:0] bit functionality when the WGM0[2:0] bits are set to a normal or CTC mode (non- PWM).

Table 19-3. Compare Output Mode, non-PWM

COM0A1	COM0A0	Description
0	0	Normal port operation, OC0A disconnected.
0	1	Toggle OC0A on Compare Match.
1	0	Clear OC0A on Compare Match.
1	1	Set OC0A on Compare Match .

The table below shows the COM0A[1:0] bit functionality when the WGM0[1:0] bits are set to fast PWM mode.

Table 19-4. Compare Output Mode, Fast PWM⁽¹⁾

COM0A1	COM0A0	Description
0	0	Normal port operation, OC0A disconnected.
0	1	WGM02 = 0: Normal Port Operation, OC0A Disconnected WGM02 = 1: Toggle OC0A on Compare Match
1	0	Clear OC0A on Compare Match, set OC0A at BOTTOM (non-inverting mode)
1	1	Set OC0A on Compare Match, clear OC0A at BOTTOM (inverting mode)

Note:

Accessing the low byte triggers the 16-bit read or write operation: When the low byte of a 16-bit register is written by the CPU, the high byte that is currently stored in TEMP and the low byte being written are both copied into the 16-bit register in the same clock cycle. When the low byte of a 16-bit register is read by the CPU, the high byte of the 16-bit register is copied into the TEMP register in the same clock cycle as the low byte is read, and must be read subsequently.

Note: To perform a 16-bit write operation, the high byte must be written before the low byte. For a 16-bit read, the low byte must be read before the high byte.

Not all 16-bit accesses use the temporary register for the high byte. Reading the OCR1A/B 16-bit registers does not involve using the temporary register.

16-bit Access

The following code examples show how to access the 16-bit Timer Registers assuming that no interrupts update the temporary register. The same principle can be used directly for accessing the OCR1A/B and ICR1 Registers. Note that when using C, the compiler handles the 16-bit access.

Assembly Code Example⁽¹⁾

```
...
; Set TCNT1 to 0x01FF
ldi    r17,0x01
ldi    r16,0xFF
out    TCNT1H,r17
out    TCNT1L,r16
; Read TCNT1 into r17:r16
in     r16,TCNT1L
in     r17,TCNT1H
...
```

The assembly code example returns the TCNT1 value in the r17:r16 register pair.

C Code Example⁽¹⁾

```
unsigned int i;
...
/* Set TCNT1 to 0x01FF */
TCNT1 = 0x1FF;
/* Read TCNT1 into i */
i = TCNT1;
...
```

Note:

1. The example code assumes that the part specific header file is included. For I/O Registers located in extended I/O map, “IN”, “OUT”, “SBIS”, “SBIC”, “CBI”, and “SBI” instructions must be replaced with instructions that allow access to extended I/O. Typically “LDS” and “STS” combined with “SBR”, “SBRC”, “SBR”, and “CBR”.

Atomic Read

It is important to notice that accessing 16-bit registers are atomic operations. If an interrupt occurs between the two instructions accessing the 16-bit register, and the interrupt code updates the temporary register by accessing the same or any other of the 16-bit Timer Registers, then the result of the access outside the interrupt will be corrupted. Therefore, when both the main code and the interrupt code update the temporary register, the main code must disable the interrupts during the 16-bit access.

The following code examples show how to perform an atomic read of the TCNT1 Register contents. The OCR1A/B or ICR1 Registers can be read by using the same principle.

the data direction for the port pin is set as output (DDR_OC1x). The PWM waveform is generated by setting (or clearing) the OC1x Register at the compare match between OCR1x and TCNT1 when the counter increments, and clearing (or setting) the OC1x Register at compare match between OCR1x and TCNT1 when the counter decrements. The PWM frequency for the output when using Phase Correct PWM can be calculated by the following equation:

$$f_{\text{OCnxPCPWM}} = \frac{f_{\text{clk_I/O}}}{2 \cdot N \cdot \text{TOP}}$$

N represents the prescale divider (1, 8, 64, 256, or 1024).

The extreme values for the OCR1x Register represent special cases when generating a PWM waveform output in the Phase Correct PWM mode. If the OCR1x is set equal to BOTTOM the output will be continuously low and if set equal to TOP the output will be continuously high for non-inverted PWM mode. For inverted PWM the output will have the opposite logic values. If OCR1A is used to define the TOP value (WGM1[3:0]=0xB) and COM1A[1:0]=0x1, the OC1A output will toggle with a 50% duty cycle.

20.12.5. Phase and Frequency Correct PWM Mode

The phase and frequency correct Pulse Width Modulation, or phase and frequency correct PWM mode (WGM1[3:0] = 0x8 or 0x9) provides a high resolution phase and frequency correct PWM waveform generation option. The phase and frequency correct PWM mode is, like the phase correct PWM mode, based on a dual-slope operation. The counter counts repeatedly from BOTTOM (0x0000) to TOP and then from TOP to BOTTOM. In non-inverting Compare Output mode, the Output Compare (OC1x) is cleared on the compare match between TCNT1 and OCR1x while up-counting, and set on the compare match while down-counting. In inverting Compare Output mode, the operation is inverted. The dual-slope operation gives a lower maximum operation frequency compared to the single-slope operation. However, due to the symmetric feature of the dual-slope PWM modes, these modes are preferred for motor control applications.

The main difference between the phase correct, and the phase and frequency correct PWM mode is the time the OCR1x Register is updated by the OCR1x Buffer Register, (see [Figure 20-8](#) and the Timing Diagram below).

The PWM resolution for the phase and frequency correct PWM mode can be defined by either ICR1 or OCR1A. The minimum resolution allowed is 2-bit (ICR1 or OCR1A set to 0x0003), and the maximum resolution is 16-bit (ICR1 or OCR1A set to MAX). The PWM resolution in bits can be calculated using the following equation:

$$R_{\text{PFCPWM}} = \frac{\log(\text{TOP}+1)}{\log(2)}$$

In phase and frequency correct PWM mode the counter is incremented until the counter value matches either the value in ICR1 (WGM1[3:0]=0x8), or the value in OCR1A (WGM1[3:0]=0x9). The counter has then reached the TOP and changes the count direction. The TCNT1 value will be equal to TOP for one timer clock cycle. The timing diagram for the phase correct and frequency correct PWM mode is shown below. The figure shows phase and frequency correct PWM mode when OCR1A or ICR1 is used to define TOP. The TCNT1 value is in the timing diagram shown as a histogram for illustrating the dual-slope operation. The diagram includes non-inverted and inverted PWM outputs. The small horizontal line marks on the TCNT1 slopes represent compare matches between OCR1x and TCNT1. The OC1x Interrupt Flag will be set when a compare match occurs.

20.14.10. Output Compare Register 1 B Low byte

Name: OCR1BL
Offset: 0x8A
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	OCR1BL[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 – OCR1BL[7:0]: Output Compare 1 B Low byte
Refer to [OCR1AL](#).

22.11.4. 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.

22.11.6. TC2 Interrupt Mask Register

Name: TIMSK2

Offset: 0x70

Reset: 0x00

Property: -

Bit	7	6	5	4	3	2	1	0
						OCIEB	OCIEA	TOIE
Access						R/W	R/W	R/W
Reset						0	0	0

Bit 2 – OCIEB: Timer/Counter2, Output Compare B Match Interrupt Enable

When the OCIEB bit is written to '1' and the I-bit in the Status Register is set (one), the Timer/Counter2 Compare Match B interrupt is enabled. The corresponding interrupt is executed if a compare match in Timer/Counter2 occurs, i.e., when the OCFB bit is set in [TIFR2](#).

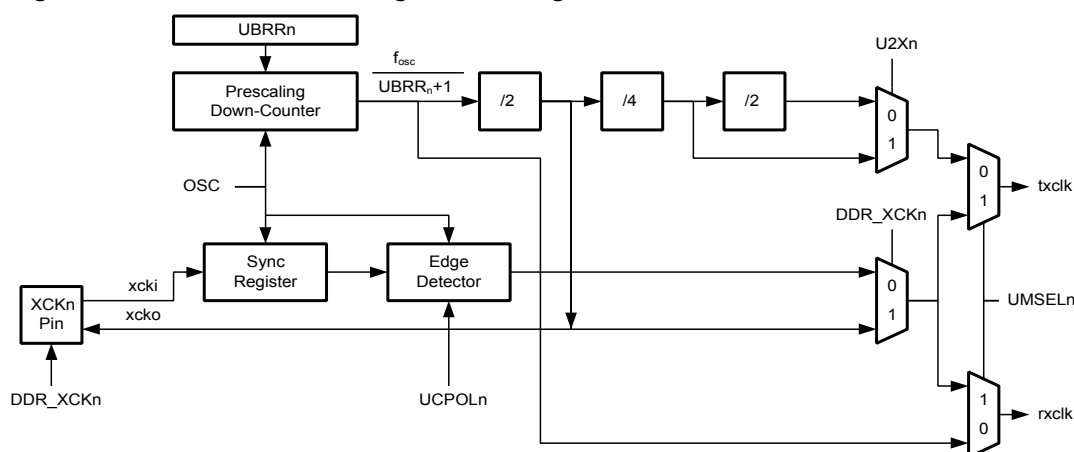
Bit 1 – OCIEA: Timer/Counter2, Output Compare A Match Interrupt Enable

When the OCIEA bit is written to '1' and the I-bit in the Status Register is set (one), the Timer/Counter2 Compare Match A interrupt is enabled. The corresponding interrupt is executed if a compare match in Timer/Counter2 occurs, i.e., when the OCFA bit is set in [TIFR2](#).

Bit 0 – TOIE: Timer/Counter2, Overflow Interrupt Enable

When the TOIE bit is written to '1' and the I-bit in the Status Register is set (one), the Timer/Counter2 Overflow interrupt is enabled. The corresponding interrupt is executed if an overflow in Timer/Counter2 occurs, i.e., when the TOV bit is set in [TIFR2](#).

Figure 24-2. Clock Generation Logic, Block Diagram



Signal description:

- txclk: Transmitter clock (internal signal).
- rxclk: Receiver base clock (internal signal).
- xcki: Input from XCKn pin (internal signal). Used for synchronous slave operation.
- xcko: Clock output to XCKn pin (internal signal). Used for synchronous master operation.
- f_{osc} : System clock frequency.

24.4.1. Internal Clock Generation – The Baud Rate Generator

Internal clock generation is used for the asynchronous and the synchronous master modes of operation. The description in this section refers to the Clock Generation Logic block diagram in the previous section..

The USART Baud Rate Register (UBRRn) and the down-counter connected to it function as a programmable prescaler or baud rate generator. The down-counter, running at system clock (f_{osc}), is loaded with the UBRRn value each time the counter has counted down to zero or when the UBRRnL Register is written. A clock is generated each time the counter reaches zero. This clock is the baud rate generator clock output ($= f_{osc}/(UBRRn+1)$). The Transmitter divides the baud rate generator clock output by 2, 8, or 16 depending on mode. The baud rate generator output is used directly by the Receiver's clock and data recovery units. However, the recovery units use a state machine that uses 2, 8, or 16 states depending on mode set by the state of the UMSEL, U2Xn and DDR_XCK bits.

The table below contains equations for calculating the baud rate (in bits per second) and for calculating the UBRRn value for each mode of operation using an internally generated clock source.

Table 24-1. Equations for Calculating Baud Rate Register Setting

Operating Mode	Equation for Calculating Baud Rate ⁽¹⁾	Equation for Calculating UBRRn Value
Asynchronous Normal mode (U2Xn = 0)	$BAUD = \frac{f_{osc}}{16(UBRRn + 1)}$	$UBRRn = \frac{f_{osc}}{16BAUD} - 1$
Asynchronous Double Speed mode (U2Xn = 1)	$BAUD = \frac{f_{osc}}{8(UBRRn + 1)}$	$UBRRn = \frac{f_{osc}}{8BAUD} - 1$
Synchronous Master mode	$BAUD = \frac{f_{osc}}{2(UBRRn + 1)}$	$UBRRn = \frac{f_{osc}}{2BAUD} - 1$

Note: 1. The baud rate is defined to be the transfer rate in bits per second (bps)

Table 24-10. Stop Bit Settings

USBS0	Stop Bit(s)
0	1-bit
1	2-bit

This bit is reserved in Master SPI Mode (MSPIM).

Bit 2 – UCSZ01 / UDORD0: USART Character Size / Data Order

UCSZ0[1:0]: USART Modes: The UCSZ0[1:0] bits combined with the UCSZ02 bit in UCSR0B sets the number of data bits (Character Size) in a frame the Receiver and Transmitter use.

Table 24-11. Character Size Settings

UCSZ0[2:0]	Character Size
000	5-bit
001	6-bit
010	7-bit
011	8-bit
100	Reserved
101	Reserved
110	Reserved
111	9-bit

UDPRD0: Master SPI Mode: When set to one the LSB of the data word is transmitted first. When set to zero the MSB of the data word is transmitted first. Refer to the *USART in SPI Mode - Frame Formats* for details.

Bit 1 – UCSZ00 / UCPHA0: USART Character Size / Clock Phase

UCSZ00: USART Modes: Refer to UCSZ01.

UCPHA0: Master SPI Mode: The UCPHA0 bit setting determine if data is sampled on the leading edge (first) or trailing (last) edge of XCK0. Refer to the *SPI Data Modes and Timing* for details.

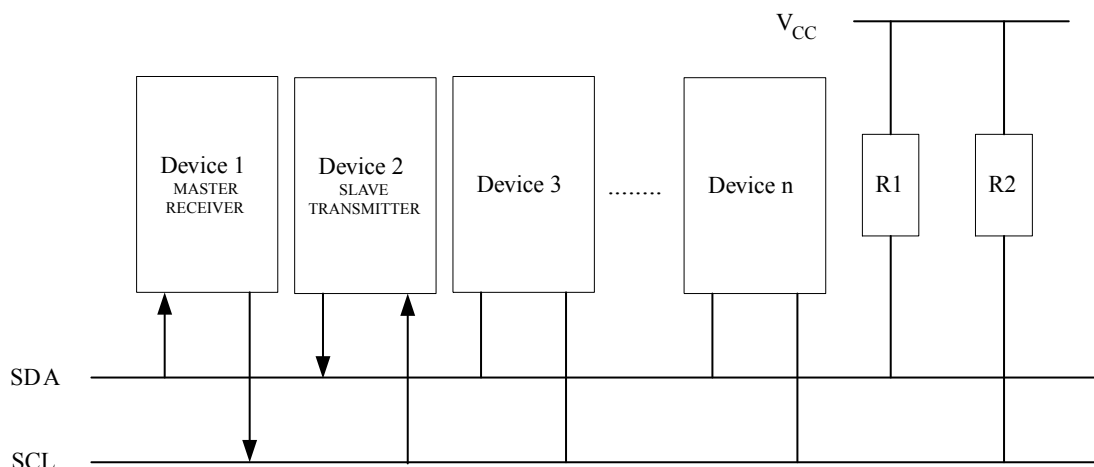
Bit 0 – UCPOL0: Clock Polarity 0

USART0 Modes: This bit is used for synchronous mode only. Write this bit to zero when asynchronous mode is used. The UCPOL0 bit sets the relationship between data output change and data input sample, and the synchronous clock (XCK0).

Table 24-12. USART Clock Polarity Settings

UCPOL0	Transmitted Data Changed (Output of TxD0 Pin)	Received Data Sampled (Input on RxD0 Pin)
0	Rising XCK0 Edge	Falling XCK0 Edge
1	Falling XCK0 Edge	Rising XCK0 Edge

Figure 26-13. Data Transfer in Master Receiver Mode



A START condition is sent by writing to the TWI Control register (TWCRn) a value of the type TWCRn=1x10x10x:

- TWCRn.TWEN must be written to '1' to enable the 2-wire Serial Interface
- TWCRn.TWSTA must be written to '1' to transmit a START condition
- TWCRn.TWINT must be cleared by writing a '1' to it.

The TWI will then test the 2-wire Serial Bus and generate a START condition as soon as the bus becomes free. After a START condition has been transmitted, the TWINT Flag is set by hardware, and the status code in TWSRn will be 0x08 (see Status Code table below). In order to enter MR mode, SLA+R must be transmitted. This is done by writing SLA+R to TWDR. Thereafter, the TWINT flag should be cleared (by writing '1' to it) to continue the transfer. This is accomplished by writing the a value to TWCRn of the type TWCRn=1x00x10x.

When SLA+R have been transmitted and an acknowledgment bit has been received, TWINT is set again and a number of status codes in TWSRn are possible. Possible status codes in Master mode are 0x38, 0x40, or 0x48. The appropriate action to be taken for each of these status codes is detailed in the table below. Received data can be read from the TWDR Register when the TWINT Flag is set high by hardware. This scheme is repeated until the last byte has been received. After the last byte has been received, the MR should inform the ST by sending a NACK after the last received data byte. The transfer is ended by generating a STOP condition or a repeated START condition. A repeated START condition is sent by writing to the TWI Control register (TWCRn) a value of the type TWCRn=1x10x10x again. A STOP condition is generated by writing TWCRn=1x01x10x:

After a repeated START condition (status code 0x10) the 2-wire Serial Interface can access the same Slave again, or a new Slave without transmitting a STOP condition. Repeated START enables the Master to switch between Slaves, Master Transmitter mode and Master Receiver mode without losing control over the bus.

26.9.3. TWI (Slave) Address Register

The TWAR should be loaded with the 7-bit Slave address (in the seven most significant bits of TWAR) to which the TWI will respond when programmed as a Slave Transmitter or Receiver, and not needed in the Master modes. In multi master systems, TWAR must be set in masters which can be addressed as Slaves by other Masters.

The LSB of TWAR is used to enable recognition of the general call address (0x00). There is an associated address comparator that looks for the slave address (or general call address if enabled) in the received serial address. If a match is found, an interrupt request is generated.

Name: TWAR

Offset: 0xBA

Reset: 0xFE

Property: -

Bit	7	6	5	4	3	2	1	0
	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	0

Bits 1, 2, 3, 4, 5, 6, 7 – TWAn: TWI (Slave) Address

These seven bits constitute the slave address of the TWI unit.

Bit 0 – TWGCE: TWI General Call Recognition Enable Bit

If set, this bit enables the recognition of a General Call given over the 2-wire Serial Bus.

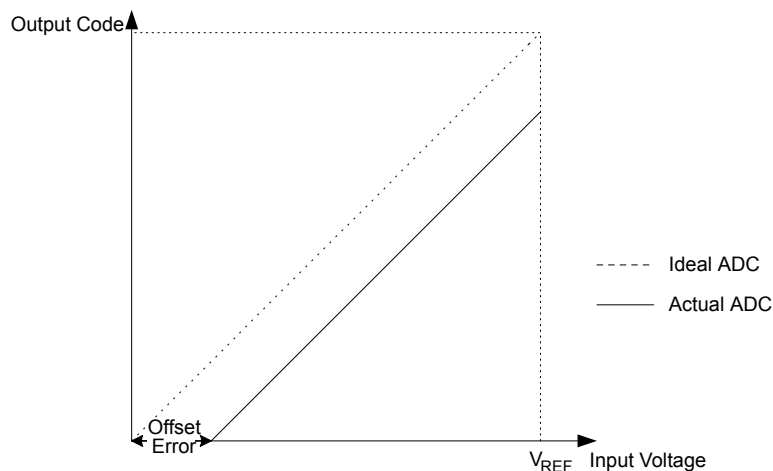
28.6.3. ADC Accuracy Definitions

An n-bit single-ended ADC converts a voltage linearly between GND and V_{REF} in 2^n steps (LSBs). The lowest code is read as 0, and the highest code is read as $2^n - 1$.

Several parameters describe the deviation from the ideal behavior:

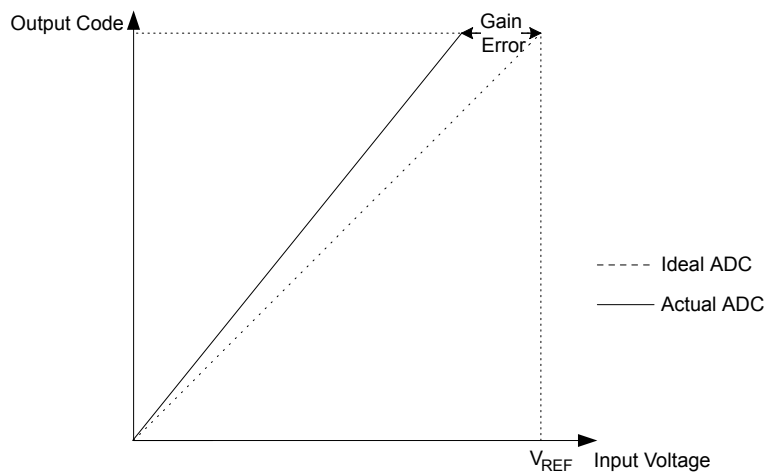
- Offset: The deviation of the first transition (0x000 to 0x001) compared to the ideal transition (at 0.5 LSB). Ideal value: 0 LSB.

Figure 28-10. Offset Error



- Gain error: After adjusting for offset, the gain error is found as the deviation of the last transition (0x3FE to 0x3FF) compared to the ideal transition (at 1.5 LSB below maximum). Ideal value: 0 LSB.

Figure 28-11. Gain Error



- Integral Non-linearity (INL): After adjusting for offset and gain error, the INL is the maximum deviation of an actual transition compared to an ideal transition for any code. Ideal value: 0 LSB.

MUX[3:0]	Single Ended Input
0110	ADC6
0111	ADC7
1000	Temperature sensor
1001	Reserved
1010	Reserved
1011	Reserved
1100	Reserved
1101	Reserved
1110	1.1V (V_{BG})
1111	0V (GND)

28.9.4. ADC Data Register High (ADLAR=0)

Name: ADCH
Offset: 0x79
Reset: 0x00
Property: ADLAR = 0

Bit	7	6	5	4	3	2	1	0
							ADC9	ADC8
Access							R	R
Reset							0	0

Bit 1 – ADC9: ADC Conversion Result
Refer to [ADCL](#).

Bit 0 – ADC8: ADC Conversion Result

Please refer to [Addressing the Flash During Self-Programming](#) for details about the use of Z-pointer during Self- Programming.

31.8.15. ATmega168/V Boot Loader Parameters

The following tables are the parameters used in the description of the self programming are given.

Table 31-10. Boot Size Configuration, ATmega168/V

BOOTSZ1	BOOTSZ0	Boot Size	Pages	Application Flash Section	Boot Loader Flash Section	End Application Section	Boot Reset Address (Start Boot Loader Section)
1	1	128 words	2	0x0000 - 0x1F7F	0x1F80 - 0x1FFF	0x1F7F	0x1F80
1	0	256 words	4	0x0000 - 0x1EFF	0x1F00 - 0x1FFF	0x1EFF	0x1F00
0	1	512 words	8	0x0000 - 0x1DFF	0x1E00 - 0x1FFF	0x1DFF	0x1E00
0	0	1024 words	16	0x0000 - 0x1BFF	0x1C00 - 0x1FFF	0x1BFF	0x1C00

Note: The different BOOTSZ Fuse configurations are shown in [Figure 31-2](#).

Table 31-11. Read-While-Write Limit, ATmega168/V

Section	Pages	Address
Read-While-Write section (RWW)	112	0x0000 - 0x1BFF
No Read-While-Write section (NRWW)	16	0x1C00 - 0x1FFF

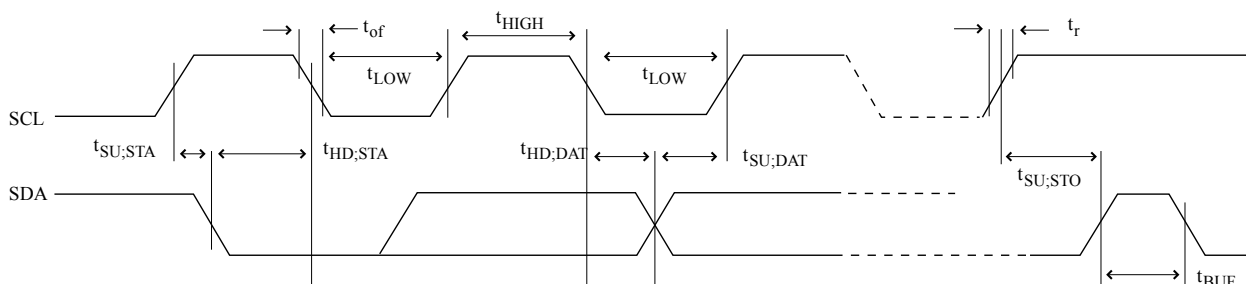
For details about these two section, please refer to [NRWW – No Read-While-Write Section](#) and [RWW – Read-While-Write Section](#).

Table 31-12. Explanation of Different Variables used in [Figure 31-3](#), ATmega168/V

Variable		Corresponding Z-value ⁽¹⁾	Description
PCMSB	12		Most significant bit in the Program Counter. (The Program Counter is 13 bits PC[12:0])
PAGEMSB	5		Most significant bit which is used to address the words within one page (64 words in a page requires 6 bits PC [5:0])
ZPCMSB		Z13	Bit in Z-register that is mapped to PCMSB. Because Z0 is not used, the ZPCMSB equals PCMSB + 1.
ZPAGEMSB		Z6	Bit in Z-register that is mapped to PAGEMSB. Because Z0 is not used, the ZPAGEMSB equals PAGEMSB + 1.

4. f_{CK} = CPU clock frequency.
5. This requirement applies to all 2-wire Serial Interface operation. Other devices connected to the 2-wire Serial Bus need only obey the general f_{SCL} requirement.

Figure 33-6. Two-wire Serial Bus Timing



33.8. ADC Characteristics

Table 33-10. ADC Characteristics

Symbol	Parameter	Condition	Min.	Typ	Max	Units
	Resolution		-	10	-	Bits
	Absolute accuracy (Including INL, DNL, quantization error, gain and offset error)	$V_{REF} = 4V, V_{CC} = 4V,$ ADC clock = 200kHz	-	2	-	LSB
		$V_{REF} = 4V, V_{CC} = 4V,$ ADC clock = 1MHz	-	4.5	-	LSB
		$V_{REF} = 4V, V_{CC} = 4V,$ ADC clock = 200kHz Noise Reduction Mode	-	2	-	LSB
		$V_{REF} = 4V, V_{CC} = 4V,$ ADC clock = 1MHz Noise Reduction Mode	-	4.5	-	LSB
	Integral Non-Linearity (INL)	$V_{REF} = 4V, V_{CC} = 4V,$ ADC clock = 200kHz	-	0.5	-	LSB
	Differential Non-Linearity (DNL)	$V_{REF} = 4V, V_{CC} = 4V,$ ADC clock = 200kHz	-	0.25	-	LSB
	Gain Error	$V_{REF} = 4V, V_{CC} = 4V,$ ADC clock = 200kHz	-	2	-	LSB
	Offset Error	$V_{REF} = 4V, V_{CC} = 4V,$ ADC clock = 200kHz	-	2	-	LSB
	Conversion Time	Free Running Conversion	13	-	260	μs
	Clock Frequency		50	-	1000	kHz
$AV_{CC}^{(1)}$	Analog Supply Voltage		$V_{CC} - 0.3$	-	$V_{CC} + 0.3$	V
V_{REF}	Reference Voltage		1.0	-	AV_{CC}	V
V_{IN}	Input Voltage		GND	-	V_{REF}	V