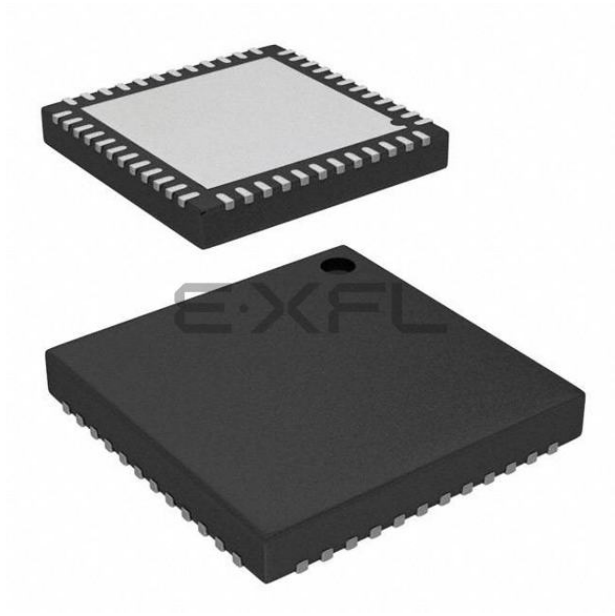


Welcome to E-XFL.COM

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.

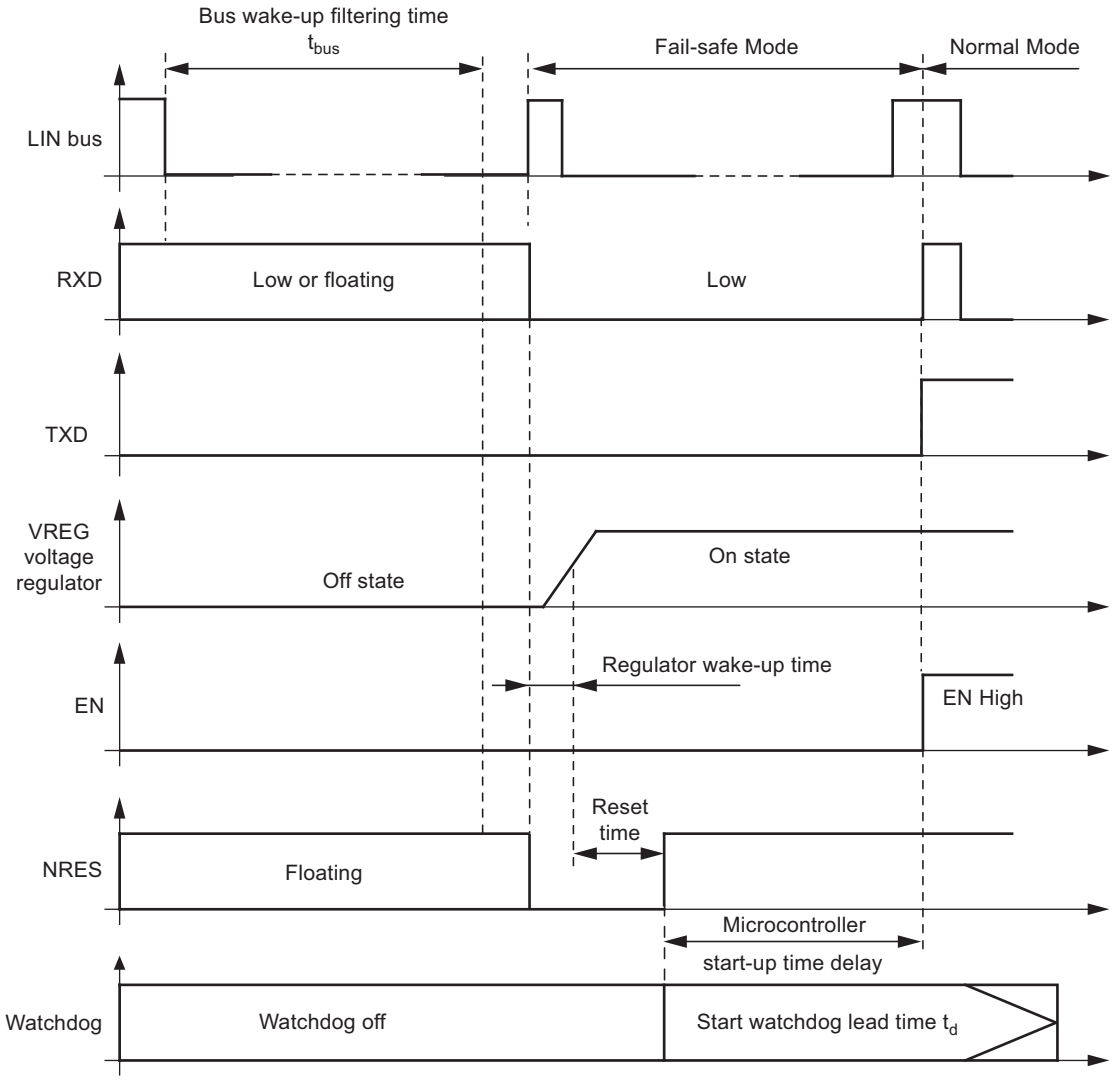
Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	AVR
Core Size	8-Bit
Speed	15MHz
Connectivity	LINbus, SPI, UART/USART, LINbus-SBC
Peripherals	Brown-out Detect/Reset, POR, WDT
Number of I/O	10
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 1x17b Sigma Delta, 1x18b Sigma Delta
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	48-VFQFN Exposed Pad
Supplier Device Package	48-VQFN (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atmega32hve2-plqw

A falling edge at the LIN pin followed by a dominant bus level maintained for a certain time period (t_{bus}) and a rising edge at pin LIN result in a remote wake-up request. The device switches from Sleep Mode to Fail-safe Mode.The VREG regulator is activated, and the internal LIN slave termination resistor is switched on. The remote wake-up request is indicated by a low level at the RXD pin to interrupt the microcontroller (see Figure 7-5).

EN high can be used to switch directly from Sleep/Silent to Fail-safe Mode. If EN is still high after VREG ramp up and undervoltage reset time, the IC switches to the Normal Mode.

Figure 7-5. LIN Wake Up from Sleep Mode

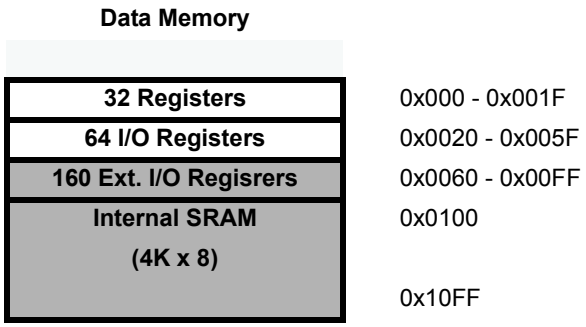


When using register indirect addressing modes with automatic pre-decrement and post-increment, the address registers X, Y, and Z are decremented or incremented.

The 32 general purpose working registers, 64 I/O Registers, 160 Extended I/O Registers, and the 4K bytes of internal data SRAM in the Atmel AVR MCU are all accessible through all these addressing modes. The Register File is described in Section 13.4 “General Purpose Register File” on page 37.

SRAM data will be unaffected by all other resets than Power-On reset. Note however that if a reset occurs while writing data types larger than 8-bit to the SRAM, the write might only be partially completed. This can leave, e.g., one byte updated in SRAM while the rest of the data word (int, long etc) was not written since the reset canceled the ongoing write operation.

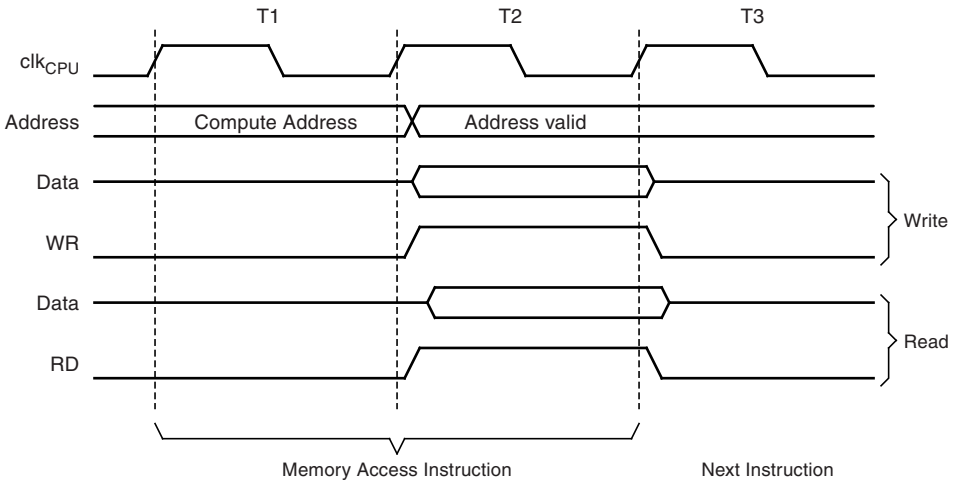
Figure 14-2. Data Memory Map



14.3.1 Data Memory Access Times

This section describes the general access timing concepts for internal memory access. The internal data SRAM access is performed in two clk_{CPU} cycles as described in Figure 14-3.

Figure 14-3. On-chip Data SRAM Access Cycles



16.7.2 PRR0 – Power Reduction Register 0

Bit	7	6	5	4	3	2	1	0	
(0x64)	–	–	–	–	PRLIN	PRSPI	PRTIM1	PRTIM0	PRR0
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7:4 – Reserved**

  These bits are reserved for future use. For compatibility with future devices, these bits must be written to zero when PRR0 is written.

- **Bit 3 – PRLIN: Power Reduction LIN UART Interface**

  Writing logic one to this bit shuts down the LIN UART Interface by stopping the clock to the module. When waking up the LIN UART again, the LIN UART should be reinitialized to ensure proper operation.

- **Bit 4 – PRSPI: Power Reduction Serial Peripheral Interface**

  Writing logic one to this bit shuts down the Serial Peripheral Interface by stopping the clock to the module. When waking up the SPI again, the SPI should be reinitialized to ensure proper operation.

- **Bit 1 – PRTIM1: Power Reduction Timer/Counter1**

  Writing a logic one to this bit shuts down the Timer/Counter1 module. When the Timer/Counter1 is enabled, operation will continue like before the shutdown.

- **Bit 0 – PRTIM0: Power Reduction Timer/Counter0**

  Writing a logic one to this bit shuts down the Timer/Counter0 module. When the Timer/Counter0 is enabled, operation will continue like before the shutdown.

The following code example shows one assembly and one C function for changing the time-out value of the Watchdog Timer.

Assembly Code Example[1]

```
WDT_Prescaler_Change:
        ; Turn off global interrupt
        in    r17, SREG ; store SREG value
        cli ; disable interrupts during timed sequence
        ; Reset Watchdog Timer
        wdr
        ; Start timed sequence
        in    r16, WDTCSR
        ori   r16, (1<<WDCE) | (1<<WDE)
        out   WDTCSR, r16
        ; --  Got four cycles to set the new values from here -
        ; Set new prescaler(time-out) value = 64Kcycles (~0.5 s)
        ldi   r16, (1<<WDE) | (1<<WDP2) | (1<<WDP0)
        out   WDTCSR, r16
        ; --  Finished setting new values, used 2 cycles -
        ; Restore global interrupt enable setting
        out   SREG, r17 ; restore SREG value (I-bit)
        ret
```

C Code Example[1]

```
void WDT_Prescaler_Change(void)
{
        char SREG;
        cSREG = SREG; /* store SREG value */
        __disable_interrupt();
        __watchdog_reset();
        /* Start timed sequence */
        WDTCSR |= (1<<WDCE) | (1<<WDE);
        /* Set new prescaler(time-out) value = 64Kcycles (~0.5 s) */
        WDTCSR  = (1<<WDE) | (1<<WDP2) | (1<<WDP0);
        SREG = cSREG; /* restore SREG value (I-bit) */
}
```

Notes:
1. See Section 12. “About Code Examples” on page 34
2. The Watchdog Timer should be reset before any change of the WDP bits, since a change in the WDP bits can result in a time-out when switching to a shorter time-out period.

As a further safe-guard against software run-away, software has the option to lock the Watchdog configuration from further modification after the initial configuration. The Watchdog configuration will then be locked until the next system reset. To lock the Watchdog configuration, the following algorithm must be followed:

1. In the same operation, write a logic one to WDCLE and WDCL.
2. Within the next four clock cycles, in the same operation, write a logic zero to WDCLE and a logic one to WDCL.

20. External Interrupts

20.1 Overview

The External Interrupts are triggered by the INT0 pin or any of the PCINT pins. Observe that, if enabled, the interrupts will trigger even if the INT0 or PCINT pins are configured as outputs. This feature provides a way of generating a software interrupt.

The Pin change interrupt PCI1 will trigger if any enabled PCINT9:2 pin toggles and a Pin change interrupt PCI0 will trigger if any enabled PCINT1:0 pin toggles. PCMSK1 and PCMSK0 Registers control which pins contribute to the pin change interrupts. Pin change interrupts on PCINT pins are detected asynchronously. This implies that these interrupts can be used for waking the part also from sleep modes other than Idle mode.

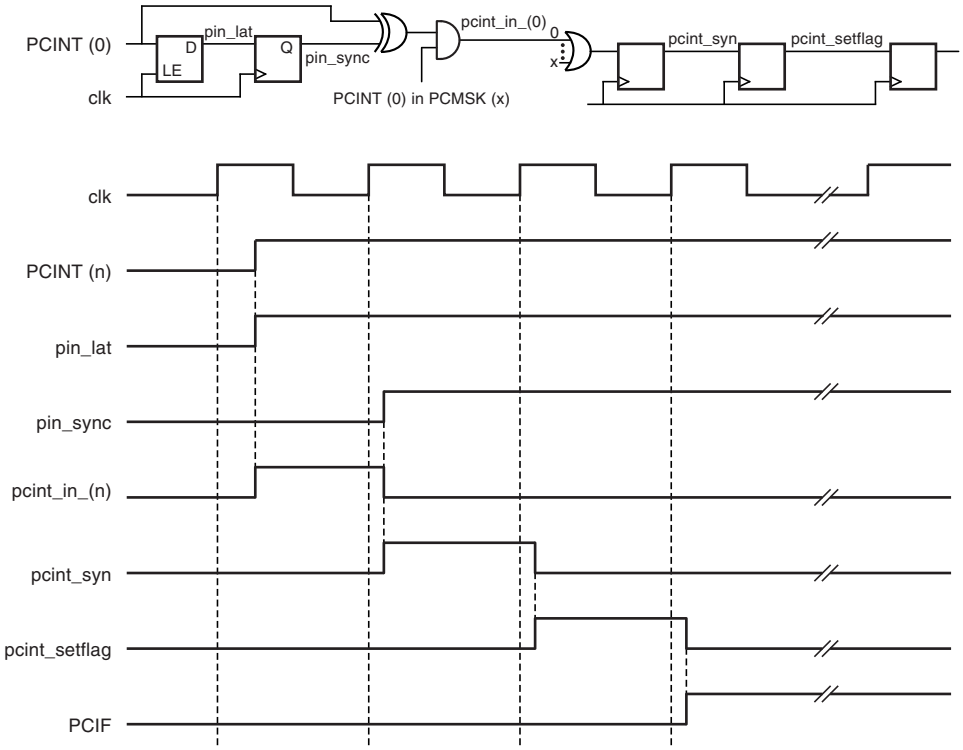
The INT0 interrupts can be triggered by a falling or rising edge or a low level. This is set up as indicated in the specification for the External Interrupt Control Register A – EICRA. When the INT0 interrupt is enabled and is configured as level triggered, the interrupt will trigger as long as the pin is held low. Note that recognition of falling or rising edge interrupts on INT0 requires the presence of an I/O clock, described in Section 15.1 “Clock Systems and their Distribution” on page 48. Low level interrupt on INT0 is detected asynchronously. This implies that this interrupt can be used for waking the part also from sleep modes other than Idle mode. The I/O clock is halted in all sleep modes except Idle mode.

Note that 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 Fuses as described in Section 15. “System Clock and Clock Options” on page 48.

20.2 Pin Change Interrupt Timing

An example of timing of a pin change interrupt is shown in Figure 20-1.

Figure 20-1. Pin Change Interrupt



21.4.7 PINB – Port B Input Pins Address

Bit	7	6	5	4	3	2	1	0	
0x03 (0x23)	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	PINB
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	

21.4.8 PBOV – Port B Override

Bit	7	6	5	4	3	2	1	0	
(0xDC)	PBOVCE	–	–	–	PBOE3	–	–	PBOE0	PBOV
Read/Write	R/W	R	R	R	R/W	R	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – PBOVCE: Port B Override Change Enable**

  The PBOE0 bit can only be changed by a timed sequence:

  1. In the same operation, write one to the PBOVCE bit and zero to all other bits in the PBOV register.
  2. Within the next four clock cycles, write zero to the PBOVCE bit and the desired values to the PBOE0 bit. This must be done in one operation.

- **Bits 6:4 - Reserved**

  These bits are reserved bits and will always read as zero.

- **Bit 3 - PBOE3: Port B Override Enable 3**

  This bit overrides normal driving capabilities for the LIN transmit signal. If this bit is set, the LIN transmit signal may either be using the internal pull-up transistor or tristating the port for the LINTX high value, depending on the PORTB3 setting. In both cases the pin will be driven actively low for the LINTX low value.

  When the PBOE3 bit is cleared, normal LINTX driving capabilities will be restored and the pin will be driven actively high for the LINTX high value and and driven actively low for the LINTX low value. This is regardless of the PORTB3 setting.

  Note that when the LIN module is disabled, the PBOE3 bit has no effect on the port functionality.

  The LINTX driving capabilities is shown in Table 21-8.

Table 21-8. LINTX Driving Capabilities

PBOE3	PORTB3	LINTX low	LINTX high
0	x	Active low	Active high
1	0	Active low	Tristated
1	1	Active low	Internal pull-up

- **Bits 2:1 - Reserved**

  These bits are reserved bits and will always read as zero.

- **Bit 0 - PBOE0: Port B Override Enable 0**

  When this bit is set, PB0 is set to one regardless of settings in the PORTB, DDRB and PINB registers. When this bit is cleared, PB0 the overriding is disabled.

The Output Compare Register OCRnA is a dual-purpose register that is also used as an 8-bit Input Capture Register ICRn. In 16-bit Input Capture mode the Output Compare Register OCRnB serves as the high byte of the Input Capture Register ICRn. In 8-bit Input Capture mode the Output Compare Register OCRnB is free to be used as a normal Output Compare Register, but in 16-bit Input Capture mode the Output Compare Unit cannot be used as there are no free Output Compare Register(s). Even though the Input Capture register is called ICRn in this section, it is referring to the Output Compare Register(s). For more information on how to access the 16-bit registers refer to Section 23.9 “Accessing Registers in 16-bit Mode” on page 102.

When a change of the logic level (an event) occurs on the Input Capture pin (ICPx), and this change confirms to the setting of the edge detector, a capture will be triggered. When a capture is triggered, the value of the counter (TCNTn) is written to the /nput Capture Register (ICRn). The Input Capture Flag (ICFn) is set at the same system clock as the TCNTn value is copied into Input Capture Register. If enabled (TICIEn=1), the Input Capture Flag generates an Input Capture interrupt. The ICFn flag is automatically cleared when the interrupt is executed. Alternatively the ICFn flag can be cleared by software by writing a logical one to its I/O bit location.

23.6.1 Input Capture Trigger Source

The default trigger source for the Input Capture unit is the completion of a CADC Instantaneous Conversion in Timer/Counter0 and the I/O port PB7 in Timer/Counter1. Alternatively can the completion of CADC Accumulated Conversion event be used as trigger source for Timer/Counter0, and the LIN RX and TX Complete events be used as trigger sources for Timer/Counter1. The Input Capture Trigger sources are selected as trigger sources by setting the Input Capture Select bits. Be aware that changing trigger source can trigger a capture. The Input Capture Flag must therefore be cleared after the change.

Both Input Capture inputs are sampled using the same technique. The edge detector is also identical. However, when the noise canceler is enabled, additional logic is inserted before the edge detector, which increases the delay by four system clock cycles. An Input Capture on Timer/Counter1 can also be triggered by software by controlling the port of the PB7 pin.

23.6.2 Noise Canceler

The noise canceler improves noise immunity by using a simple digital filtering scheme. The noise canceler input is monitored over four samples, and all four must be equal for changing the output that in turn is used by the edge detector.

The noise canceler is enabled by setting the Input Capture Noise Canceler (ICNCn) bit in Section 23.10.1 “TCCRnA – Timer/Counter n Control Register A” on page 104. When enabled the noise canceler introduces additional four system clock cycles of delay from a change applied to the input, to the update of the ICRn Register. The noise canceler uses the system clock and is therefore not affected by the prescaler.

23.6.3 Using the Input Capture Unit

The main challenge when using the Input Capture unit is to assign enough processor capacity for handling the incoming events. The time between two events is critical. If the processor has not read the captured value in the ICRn Register before the next event occurs, the ICRn will be overwritten with a new value. In this case the result of the capture will be incorrect.

When using the Input Capture interrupt, the ICRn Register should be read as early in the interrupt handler routine as possible. The maximum interrupt response time is dependent on the maximum number of clock cycles it takes to handle any of the other interrupt requests.

Measurement of an external signal duty cycle requires that the trigger edge is changed after each capture. Changing the edge sensing must be done as early as possible after the ICRn Register has been read. After a change of the edge, the Input Capture Flag (ICFn) must be cleared by software (writing a logical one to the I/O bit location). For measuring frequency only, the trigger edge change is not required.

Table 23-3. Timer/Counter0 Input Capture Source (ICS)

ICS[1:0]	Source
00[1][2]	ICP00: CADC Instantaneous Conversion Complete Interrupt
01[1][2]	ICP01: CADC Accumulated Conversion Complete Interrupt
10	Reserved for future use
11	Reserved for future use

Notes:
1. The noise canceler may filter out this source and it is therefore not recommended to use noise canceler with this source.
2. If this interrupt is chosen as the Input Capture source, an Input Capture event will generate both the chosen interrupt and the Input Capture interrupt. If both interrupts are enabled, the sequence in which the interrupts are handled depends on a number of factors. The application software must therefore allow for both the Input Capture interrupt being handled before the chosen interrupt trigger, and after the chosen interrupt trigger.

Table 23-4. Timer/Counter1 Input Capture Source (ICS)

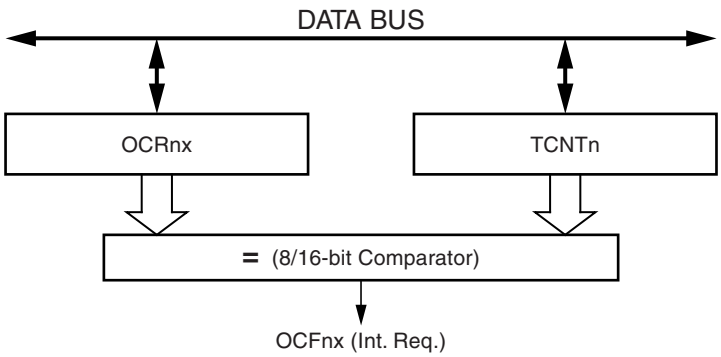
ICS[1:0]	Source
00	ICP10: Port PB7
01[1][2]	ICP11: LIN RX Complete Interrupt
10[1][2]	ICP12: LIN TX Complete Interrupt
11	Reserved for future use

Notes:
1. The noise canceler may filter out this source and it is therefore not recommended to use noise canceler with this source.
2. If this interrupt is chosen as the Input Capture source, an Input Capture event will generate both the chosen interrupt and the Input Capture interrupt. If both interrupts are enabled, the sequence in which the interrupts are handled depends on a number of factors. The application software must therefore allow for both the Input Capture interrupt being handled before the chosen interrupt trigger, and after the chosen interrupt trigger.

23.7 Output Compare Unit

The comparator continuously compares the Timer/Counter (TCNTn) with the Output Compare Registers (OCRnA and OCRnB), and whenever the Timer/Counter equals to the Output Compare Regisers, the comparator signals a match. A match will set the Output Compare Flag at the next timer clock cycle. In 8-bit mode the match can set either the Output Compare Flag OCFnA or OCFnB, but in 16-bit mode the match can set only the Output Compare Flag OCFnA as there is only one Output Compare Unit. If the corresponding interrupt is enabled, the Output Compare Flag generates an Output Compare interrupt. The Output Compare Flag is automatically cleared when the interrupt is executed. Alternatively, the flag can be cleared by software by writing a logical one to its I/O bit location. Figure 23-5 on page 99 shows a block diagram of the Output Compare unit.

Figure 23-5. Output Compare Unit, Block Diagram



- **Bit 5 – ICNCn: Input Capture Noise Canceler**

  Setting this bit activates the Input Capture Noise Canceler. When the noise canceler is activated, the input from the Input Capture Source is filtered. The filter function requires four successive equal valued samples of the Input Capture Source for changing its output. The Input Capture is therefore delayed by four System Clock cycles when the noise canceler is enabled.

- **Bit 4 – ICESn: Input Capture Edge Select**

  This bit selects which edge on the Input Capture Source 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. 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.

- **Bits 3 – Reserved**

  This bit is reserved in the Atmel® AVR MCU and should always be written to zero.

- **Bits 2:1 – Reserved**

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

- **Bit 0 – WGMn0: Waveform Generation Mode**

  This bit controls the counting sequence of the counter, the source for maximum (TOP) counter value, see Figure 23-6 on page 100. Modes of operation supported by the Timer/Counter unit are: Normal mode (counter) and Clear Timer on Compare Match (CTC) mode (see Section 23.8 “Timer/Counter Timing Diagrams” on page 100).

23.10.2 TCCRnC – Timer/Counter n Control Register C

Bit	7	6	5	4	3	2	1	0	
	–	–	–	–	–	–	ICn1	ICn0	TCCRnC
Read/Write	R	R	R	R	R	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7:2 – Reserved**

  These bits are reserved bits in the Atmel® AVR MCU and will always read as zero.

- **Bit 1:0 – ICS[1:0]: Input Capture Select 1:0**

  These bits control which Input Capture source that should trigger the Timer/Counter Input Capture functionality. To also trigger the Timer/Counter n Input Capture interrupt, the TICIEn bit in the Timer Interrupt Mask Register TIMSK) must be set.

  See Table 23-3 on page 99 and Table 23-4 on page 99 for Input Capture sources.

23.10.3 TCNTnL – Timer/Counter n Register Low Byte

Bit	7	6	5	4	3	2	1	0	
	TCNTnL[7:0]								TCNTnL
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 TCNTnL gives direct access, both for read and write operations, to the Timer/Counter unit 8-bit counter. Writing to the TCNTnL Register blocks (disables) the Compare Match on the following timer clock. Modifying the counter (TCNTnL) while the counter is running, introduces a risk of missing a Compare Match between TCNTnL and the OCRnx Registers. In 16-bit mode the TCNTnL register contains the lower part of the 16-bit Timer/Counter n Register.

23.10.4 TCNTnH – Timer/Counter n Register High Byte

Bit	7	6	5	4	3	2	1	0	
	TCNTnH[7:0]								TCNTnH
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	

When 16-bit mode is selected (the TCWn bit is set to one) the Timer/Counter Register TCNTnH combined to the Timer/Counter Register TCNTnL gives direct access, both for read and write operations, to the Timer/Counter unit 16-bit counter. To ensure that both the high and low bytes are read and written simultaneously when the CPU accesses these registers, the access is performed using an 8-bit temporary high byte register (TEMP). This temporary register is shared by all the other 16-bit registers. See Section 23.9 “Accessing Registers in 16-bit Mode” on page 102.

23.10.5 OCRnA – Timer/Counter n Output Compare Register A

Bit	7	6	5	4	3	2	1	0	
	OCRnA[7:0]								OCRnA
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 A contains an 8-bit value that is continuously compared with the counter value (TCNTnL). A match can be used to generate an Output Compare interrupt.

In 16-bit mode the OCRnA register contains the low byte of the 16-bit Output Compare Register. To ensure that both the high and the low bytes are written simultaneously when the CPU writes to these 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. See Section 23.9 “Accessing Registers in 16-bit Mode” on page 102.

23.10.6 OCRnB – Timer/Counter n Output Compare Register B

Bit	7	6	5	4	3	2	1	0	
	OCRnB[7:0]								OCRnB
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 B contains an 8-bit value that is continuously compared with the counter value (TCNTnL in 8-bit mode and TCNTnH in 16-bit mode). A match can be used to generate an Output Compare interrupt.

In 16-bit mode the OCRnB register contains the high byte of the 16-bit Output Compare Register. To ensure that both the high and the low bytes are written simultaneously when the CPU writes to these 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. See Section 23.9 “Accessing Registers in 16-bit Mode” on page 102.

23.10.7 TIMSKn – Timer/Counter n Interrupt Mask Register

Bit	7	6	5	4	3	2	1	0	
	–	–	–	–	ICIEn	OCIEnB	OCIEnA	TOIEn	TIMSKn
Read/Write	R	R	R	R	R/W	R/W	R/W	R	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 3 – ICIEn: Timer/Counter n Input Capture Interrupt Enable**

  When this bit is written to one, and the I-flag in the Status Register is set (interrupts globally enabled), the Timer/Counter n Input Capture interrupt is enabled. The corresponding Interrupt Vector (see Section 19. “Interrupts” on page 70) is executed when the ICFn flag, located in TIFRn, is set.

- **Bit 2 – OCIEnB: Timer/Counter n Output Compare Match B Interrupt Enable**

  When the OCIEnB bit is written to one, and the I-bit in the Status Register is set, the Timer/Counter Compare Match B interrupt is enabled. The corresponding interrupt is executed if a Compare Match in Timer/Counter occurs, i.e., when the OCFnB bit is set in the Timer/Counter Interrupt Flag Register – TIFRn.

25.8 LIN / UART Register Description

25.8.1 LINCR – LIN Control Register

Bit	7	6	5	4	3	2	1	0	
(0xC0)	LSWRES	LIN13	LCONF1	LCONF0	LENA	LCMD2	LCMD1	LCMD0	LINCR
Read/Write	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – LSWRES: Software Reset**
  - 0 = No action,
  - 1 = Software reset (this bit is self-reset at the end of the reset procedure).
- **Bit 6 – LIN13: LIN 1.3 mode**
  - 0 = LIN 2.1 (default),
  - 1 = LIN 1.3.
- **Bit 5:4 – LCONF[1:0]: Configuration**
  1. LIN mode (default = 00):
     - 00 = LIN Standard configuration (listen mode “off”, CRC “on” and Frame_Time_Out “on”,
     - 01 = No CRC, no Time out (listen mode “off”),
     - 10 = No Frame_Time_Out (listen mode “off” and CRC “on”),
     - 11 = Listening mode (CRC “on” and Frame_Time_Out “on”).
  2. UART mode (default = 00):
     - 00 = 8-bit, no parity (listen mode “off”),
     - 01 = 8-bit, even parity (listen mode “off”),
     - 10 = 8-bit, odd parity (listen mode “off”),
     - 11 = Listening mode, 8-bit, no parity.
- **Bit 3 – LENA: Enable**
  - 0 = Disable (both LIN and UART modes),
  - 1 = Enable (both LIN and UART modes).
- **Bit 2:0 – LCMD[2:0]: Command and mode**

  The command is only available if LENA is set.
  - 000 = LIN Rx Header - LIN abort,
  - 001 = LIN Tx Header,
  - 010 = LIN Rx Response,
  - 011 = LIN Tx Response,
  - 100 = UART Rx and Tx Byte disable,
  - 11x = UART Rx Byte enable,
  - 1x1 = UART Tx Byte enable.

25.8.9 LINSEL – LIN Data Buffer Selection Register

Bit	7	6	5	4	3	2	1	0	
(0xC9)	–	–	–	–	LAINC	LINDX2	LINDX1	LINDX0	LINSEL
Read/Write	–	–	–	–	R/W	R/W	R/W	R/W	
Initial Value	–	–	–	–	0	0	0	0	

- **Bits 7:4 – Reserved Bits**

  These bits are reserved for future use. For compatibility with future devices, they must be written to zero when LINSEL is written.

- **Bit 3 – $\overline{\text{LAINC}}$: $\overline{\text{Auto Increment}}$ of Data Buffer Index**

  In LIN mode:

  - 0 = Auto incrementation of FIFO data buffer index (default),
  - 1 = No auto incrementation.

  In UART mode this field is unused.

- **Bits 2:0 – LINDX 2:0: FIFO LIN Data Buffer Index**

  In LIN mode: location (index) of the LIN response data byte into the FIFO data buffer. The FIFO data buffer is accessed through LINDAT.

  In UART mode this field is unused.

25.8.10 LINDAT – LIN Data Register

Bit	7	6	5	4	3	2	1	0	
(0xCA)	LDATA7	LDATA6	LDATA5	LDATA4	LDATA3	LDATA2	LDATA1	LDATA0	LINDAT
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 – LDATA[7:0]: LIN Data In / Data out**

  In LIN mode: FIFO data buffer port.

  In UART mode: data register (no data buffer - no FIFO).

  - In Write access, data out.
  - In Read access, data in.

26.6.15 DIDR0 – Digital Input Disable Register 0

Bit	7	6	5	4	3	2	1	0	
(0x7E)	–	–	–	–	–	–	PA1DID	PA0DID	DIDR0
Read/Write	R	R	R	R	R	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

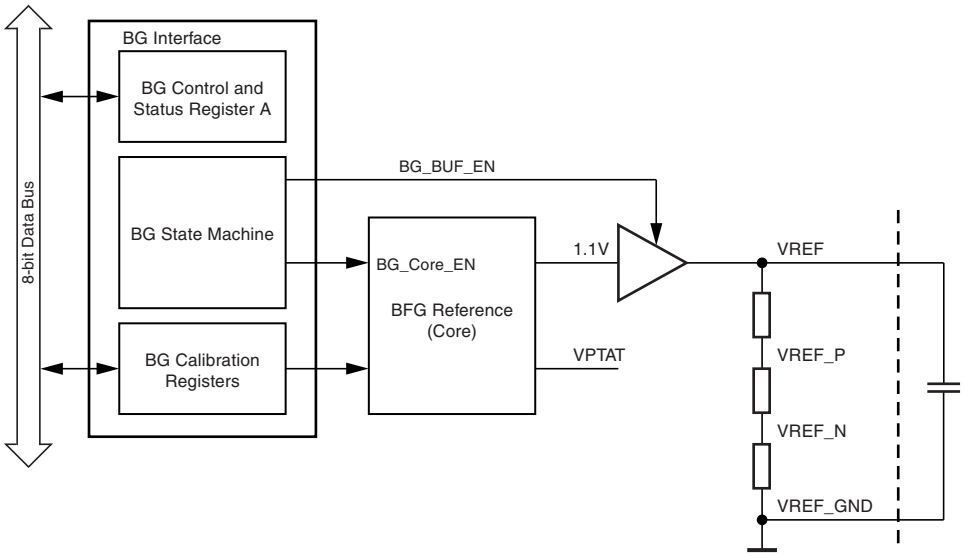
- **Bit 7:2 - Reserved**

  These bits are reserved and is always read as zero.

- **Bit 1:0 - PA1DID:PA0DID**

  When this bit is written logic one, the digital input buffer on the corresponding ADC pin is disabled. The corresponding PIN Register bit will always read as zero when this bit is set. When an analog signal is applied to the PA1DID:PA0DID pin and the digital input from this pin is not needed, this bit should be written logic one to reduce power consumption in the digital input buffer.

Figure 27-1. Reference Circuitry

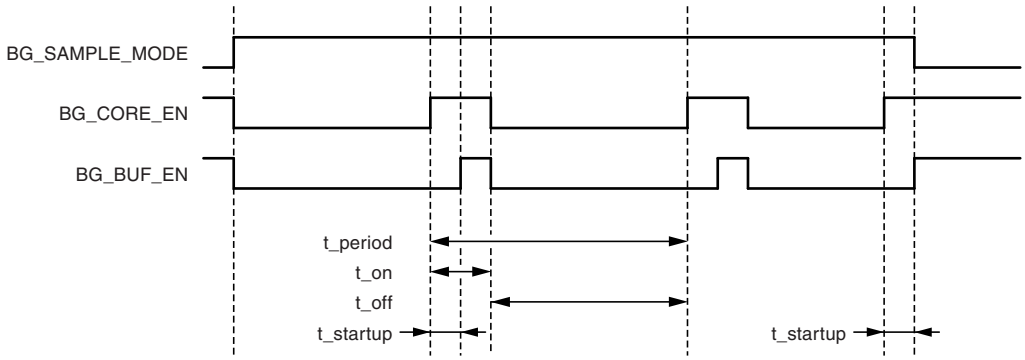


27.4 Band Gap Sample Mode

The Band Gap module has two operation modes, continuous mode and sampled mode. When the Band Gap module is run in sampled mode, the Band Gap core and output buffer are switched on and off at regular time intervals. When the Band Gap core and output buffer are off, the reference voltage is stored on the external capacitor. In continuous mode, the core and output buffer are switched on all the time. The sampled mode is enabled automatically in Power Down sleep mode. In all other operating modes, the continuous mode is used.

The timing of the signals in sampled mode is shown in Figure 27-2. The BGSC bits in the BGCSRA register control the timing of t_off. For the timing of BG_CORE_EN and BG_BUF_EN also see Section 31. “Electrical Characteristics AVR MCU” on page 192ff.

Figure 27-2. Band Gap Timing

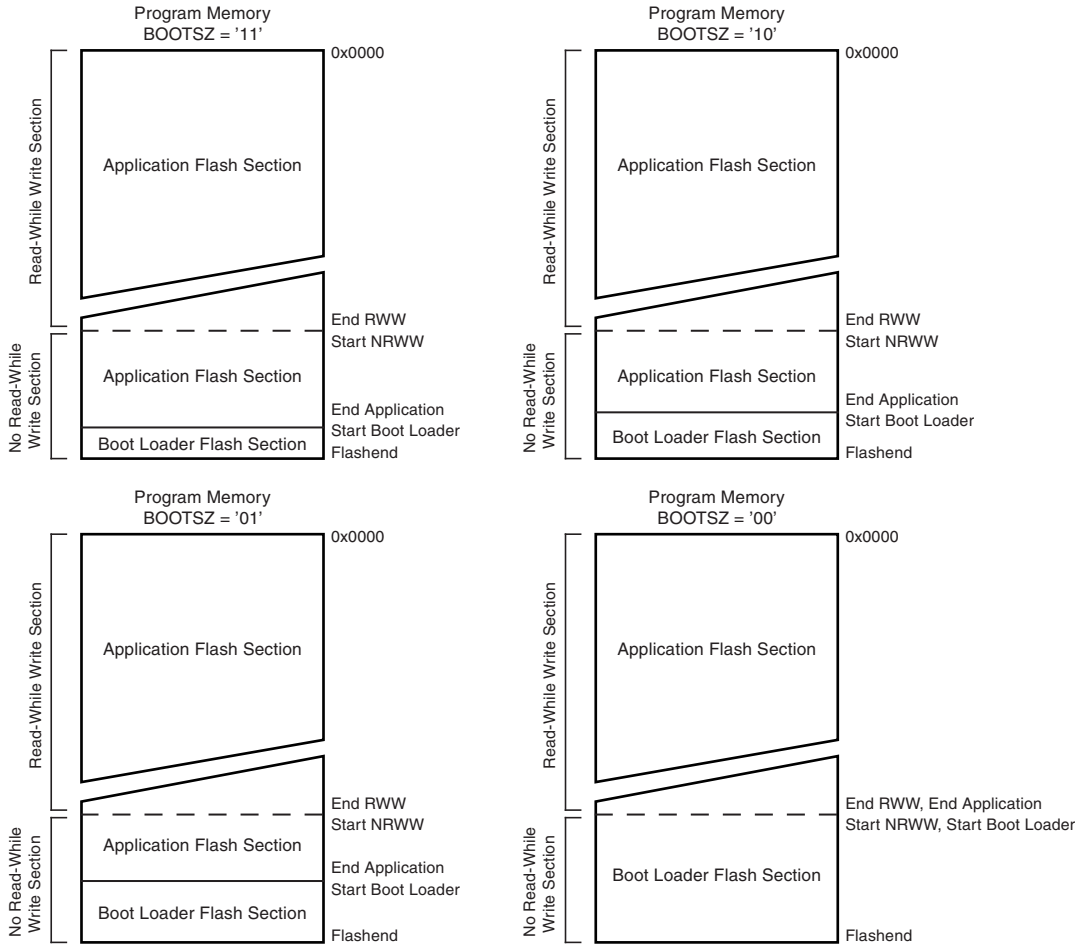


The timing of the sampled mode is dependent on the leakage current from the external capacitor storing the VREF voltage during the off period. For instance a ceramic capacitor of 1 µF may have an insulation resistance of ~500M , while a tantalum capacitor of 1 µF may have an insulation resistance of ~1M . The maximum off period is given by:

$$t\_off < \frac{CREF \times \Delta VREF \times R_{ins}}{VREF}$$

where CREF and R_{ins} are the capacitance and the insulation resistance of the external decoupling capacitor, VREF is the bandgap output voltage and $\Delta VREF$ is the maximum allowed variation in VREF. As the Brown-out Detector is the only module using the Voltage Reference in Power Down the maximum acceptable degradation of the BOD level in Power-down is the deciding factor when setting VREF. Setting ΔVREF to 15.5mV gives t_off < 7s for a ceramic capacitor and t_off < 14 ms for a tantalum capacitor. Timeout settings longer than t_off will violate the VREF requirement used in this example.

Figure 29-2. Memory Sections



Note: The parameters in the figure above are given in Section 29.8.13 "Atmel ATmega32HVE Boot Loader Parameters" on page 176.

29.5 Boot Loader Lock Bits

If no Boot Loader capability is needed, the entire Flash is available for application code. The Boot Loader has two separate sets of Boot Lock bits which can be set independently. This gives the user a unique flexibility to select different levels of protection.

The user can select:

- To protect the entire Flash from a software update by the MCU.
- To protect only the Boot Loader Flash section from a software update by the MCU.
- To protect only the Application Flash section from a software update by the MCU.
- Allow software update in the entire Flash.

See Table 30-2 on page 180 for further details. The Boot Lock bits can be set in software and in Serial or Parallel Programming mode, but they can be cleared by a Chip Erase command only. The general Write Lock (Lock Bit mode 2) does not control the programming of the Flash memory by SPM instruction. Similarly, the general Read/Write Lock (Lock Bit mode 1) does not control reading nor writing by LPM/SPM, if it is attempted.

30.2.2 Low Byte

Table 30-4. Fuse Low Byte

Bit No	Fuse Low Byte	Description	Default Value
7	WDTON	Watchdog Timer always on	1 (unprogrammed)(1)
6	EESAVE	EEPROM memory is preserved through the Chip Erase	1 (unprogrammed, EEPROM not preserved)
5	SPIEN	Enable Serial Programmable Data Downloading	0 (programmed, SPI programming enabled)
4	BODEN	BOD Enable	1 (unprogrammed, BOD disabled)(5)
3	CKDIV8(3)	Divide clock by 8	0 (programmed)
2:1	SUT1:0	Select start-up time	11 (unprogrammed)(2)
0	OSCSEL0	Oscillator Select	1 (unprogrammed)(4)

Notes:
1. The Watchdog is enabled/disabling by writing to the Watchdog Timer Control and Status Register (WDTCSR). But as a fail-safe, the WDTON fuse can be used to force the Watchdog to run in System Reset mode.
2. The SUTx fuse bits are used to configure the startup time from sleep or reset. By default the longest startup time is selected.
3. See Section 15.4 “System Clock Prescaler” on page 50 for details.
4. When unprogrammed, PLL is used as system clock source. Programming this fuse is for test purpose only, and should not be used in application.
5. Disabling BOD assumed that safe VCC operation is guaranteed by other parts of the application.

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.

30.2.3 Latching of Fuses

The fuse values are latched when the device enters programming mode and changes of the fuse values will have no effect until the part leaves Programming mode. This does not apply to the EESAVE Fuse which will take effect once it is programmed. The fuses are also latched on Power-up in Normal mode.

30.3 Signature Bytes

All Atmel microcontrollers have a three-byte signature code which identifies the device. This code can be read in both Programming mode, also when the device is locked. The three bytes reside in a separate address space. The signature bytes of the Atmel® AVR MCU are given in Table 30-5.

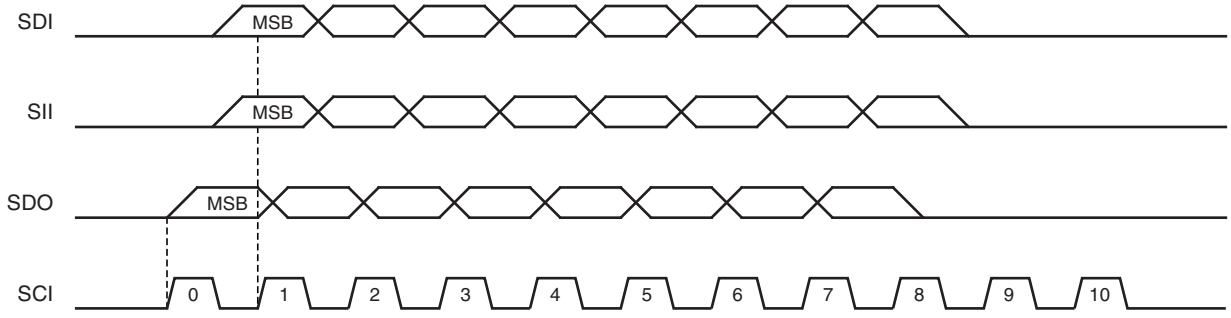
Table 30-5. Device ID

Part	Signature Bytes Address		
	0x000	0x001	0x002
Atmel ATmega32HVE	0x1E	0x95	0x13
Atmel ATmega64HVE	0x1E	0x96	0x10

30.4 Calibration Bytes

The Atmel® AVR MCU has calibration bytes for the RC Oscillators, internal voltage reference, internal temperature reference and TBD. These bytes reside in the signature address space. See Section 29.8.9 “Reading the Signature Row from Software” on page 173 for details.

Figure 30-5. High-voltage Serial Programming Waveforms



30.8.5 Programming the EEPROM

The EEPROM is organized in pages, see Section 31.8.2 “High-voltage Serial Programming” on page 202. When programming the EEPROM, the data is latched into a page buffer. This allows one page of data to be programmed simultaneously. The programming algorithm for the EEPROM Data memory is as follows (refer to Table 30-14 on page 190):

1. Load Command “Write EEPROM”.
2. Load EEPROM Page Buffer.
3. Program EEPROM Page. Wait after Instr. 2 until SDO goes high for the “Page Programming” cycle to finish.
4. Repeat 2 through 3 until the entire EEPROM is programmed or until all data has been programmed.
5. End Page Programming by Loading Command “No Operation”.

30.8.6 Reading the Flash

The algorithm for reading the Flash memory is as follows (refer to Table 30-14 on page 190):

1. Load Command "Read Flash".
2. Read Flash Low and High Bytes. The contents at the selected address are available at serial output SDO.

30.8.7 Reading the EEPROM

The algorithm for reading the EEPROM memory is as follows (refer to Table 30-14 on page 190):

1. Load Command “Read EEPROM”.
2. Read EEPROM Byte. The contents at the selected address are available at serial output SDO.

30.8.8 Programming and Reading the Fuse and Lock Bits

The algorithms for programming and reading the Fuse Low/High bits and Lock bits are shown in Table 30-14 on page 190.

30.8.9 Reading the Signature Bytes and Calibration Byte

The algorithms for reading the Signature bytes and Calibration byte are shown in Table 30-14 on page 190.

31.4 Oscillator Characteristics

T_A= -40 to 125°C, V_{CC} = 3.3V unless otherwise noted.

No.	Parameters		Test Conditions	Pin	Symbol	Min	Typ	Max	Unit	Type*
4.1	Slow RC Oscillator	Frequency	Reference frequency: Slow RCOSC			–4%	128	+4%	kHz	A
4.2		Temperature drift	Centered at 25°C			–1		+1	%	A
4.3	PLL	Frequency multiplication factor					112			D
4.4		Startup time from Pdown/Psave	Latency from module enabled until first clock edge (ref = 128kHz)				25	40	µs	C
4.5		Frequency settling time	Setting to 99% of target				180	250	µs	C
4.6	Ultra Low Power RC Oscillator	Frequency, initial accuracy				–20%	131	+20%	kHz	A
4.7		Temperature drift	Centered at 25°C. VCC = 3.3V			–5		+5	%	A

*) Type means: A = 100% tested, B = 100% correlation tested, C = Characterized on samples, D = Design parameter

31.5 External Interrupt Characteristics

Asynchronous External Interrupt Characteristics

No.	Parameters	Test Conditions	Pin	Symbol	Min	Typ	Max	Unit	Type*
5.1	Pulse width for asynchronous external interrupt			t_{INT}		500	2000	ns	C

*) Type means: A = 100% tested, B = 100% correlation tested, C = Characterized on samples, D = Design parameter

32. Register Summary

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
(0xFF)	Reserved	–	–	–	–	–	–	–	–	
(0xFE)	Reserved	–	–	–	–	–	–	–	–	
(0xFD)	Reserved	–	–	–	–	–	–	–	–	
(0xFC)	Reserved	–	–	–	–	–	–	–	–	
(0xFB)	Reserved	–	–	–	–	–	–	–	–	
(0xFA)	Reserved	–	–	–	–	–	–	–	–	
(0xF9)	Reserved	–	–	–	–	–	–	–	–	
(0xF8)	Reserved	–	–	–	–	–	–	–	–	
(0xF7)	Reserved	–	–	–	–	–	–	–	–	
(0xF6)	VADAC3	VADAC3[31:24]								158
(0xF5)	VADAC2	VADAC2[23:16]								158
(0xF4)	VADAC1	VADAC1[15:8]								158
(0xF3)	VADAC0	VADAC0[7:0]								158
(0xF2)	VADICH	VADICH[15:8]								158
(0xF1)	VADICL	VADICL[7:0]								158
(0xF0)	CADAC3	CADAC3[31:24]								159
(0xEF)	CADAC2	CADAC2[23:16]								159
(0xEE)	CADAC1	CADAC1[15:8]								159
(0xED)	CADAC0	CADAC0[7:0]								159
(0xEC)	CADICH	CADICH[15:8]								159
(0xEB)	CADICL	CADICL[7:0]								159
(0xEA)	CADRCLH	CADRCLH[15:8]								157
(0xE9)	CADRCLL	CADRCLL[7:0]								157
(0xE8)	ADIMR	–	–	VADACIE	VADICIE	–	CADRCIE	CADACIE	CADICIE	157
(0xE7)	ADIFR	–	–	VADACIF	VADICIF	–	CADRCIF	CADACIF	CADICIF	156
(0xE6)	ADCRE	VADEN	–	VADREFS	VADPDM1	VADPDM0	VAMUX2	VAMUX1	VAMUX0	155
(0xE5)	ADCRD	–	–	CADG2	CADG1	CADG0	CADPDM1	CADPDM0	CADDSEL	154
(0xE4)	ADCRC	CADEN	–	CADRCM1	CADRCM0	CADRCT3	CADRCT2	CADRCT1	CADRCT0	153
(0xE3)	ADCRB	–	–	–	ADIDES1	ADIDES0	ADADES2	ADADES1	ADADES0	152
(0xE2)	ADCRA	–	–	–	–	ADPSEL	ADCMS1	ADCMS0	CKSEL	151
(0xE1)	ADSCSRB	–	VADICPS	VADACRB	VADICRB	–	CADICPS	CADACRB	CADICRB	150
(0xE0)	ADSCSRA	–	–	–	–	–	SBSY	SCMD1	SCMD0	149
(0xDF)	Reserved	–	–	–	–	–	–	–	–	
(0xDE)	Reserved	–	–	–	–	–	–	–	–	
(0xDD)	Reserved	–	–	–	–	–	–	–	–	
(0xDC)	PBOV	PBOVCE	–	–	–	PBOE3	–	–	PBOE0	89
(0xDB)	Reserved	–	–	–	–	–	–	–	–	
(0xDA)	Reserved	–	–	–	–	–	–	–	–	
(0xD9)	Reserved	–	–	–	–	–	–	–	–	
(0xD8)	PLLCSR	–	–	SWEN	LOCK	–	–	PLLCIF	PLLCIE	51
(0xD7)	Reserved	–	–	–	–	–	–	–	–	
(0xD6)	Reserved	–	–	–	–	–	–	–	–	
(0xD5)	Reserved	–	–	–	–	–	–	–	–	
(0xD4)	BGLR	–	–	–	–	–	–	BGPLE	BPGL	164
(0xD3)	BGCRA	BGCN[7:0]								163
(0xD2)	BGCRB	BGCL[7:0]								163
(0xD1)	BGCSRA	–	–	–	–	–		BGSC[2:0]		163
(0xD0)	Reserved	–	–	–	–	–	–	–	–	
(0xCF)	Reserved	–	–	–	–	–	–	–	–	
(0xCE)	Reserved	–	–	–	–	–	–	–	–	
(0xCD)	Reserved	–	–	–	–	–	–	–	–	

Notes:
1. For compatibility with future devices, reserved bits should be written to zero if accessed. Reserved I/O memory addresses should never be written.
2. I/O registers within the address range $00 - $1F are directly bit-accessible using the SBI and CBI instructions. In these registers, the value of single bits can be checked by using the SBIS and SBIC instructions.
3. Some of the status flags are cleared by writing a logical one to them. Note that the CBI and SBI instructions will operate on all bits in the I/O register, writing a one back into any flag read as set, thus clearing the flag. The CBI and SBI instructions work with registers 0x00 to 0x1F only.
4. When using the I/O specific commands IN and OUT, the I/O addresses $00 - $3F must be used. When addressing I/O registers as data space using LD and ST instructions, $20 must be added to these addresses. The Atmel AVR MCU is a complex microcontroller with more peripheral units than can be supported within the 64 location reserved in Opcode for the IN and OUT instructions. For the Extended I/O space from $60 - $FF in SRAM, only the ST/STS/STD and LD/LDS/LDD instructions can be used.

32. Register Summary (Continued)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
(0xCC)	Reserved	–	–	–	–	–	–	–	–	
(0xCB)	Reserved	–	–	–	–	–	–	–	–	
(0xCA)	LINDAT	LDATA[7:0]								137
(0xC9)	LINSEL	–	–	–	–	$\overline{\text{LAINC}}$		LINDX[2:0]		137
(0xC8)	LINIDR	LP1	LP0	LID5/LIDL1	LID4/LIDL0		LID[3:0]			136
(0xC7)	LINLDR	LTXDL[7:0]								136
(0xC6)	LINBRH	–	–	–	–		LDIV[11:8]			135
(0xC5)	LINBRL	LDIV[7:0]								135
(0xC4)	LINBTR	LDISR	–	LBT[5:0]						135
(0xC3)	LINERR	LABORT	LTOERR	LOVERR	LFERR	LSERR	LPERR	LCERR	LBERR	134
(0xC2)	LINENIR	–	–	–	–	LENERR	LENIDOK	LENTXOK	LENRXOK	134
(0xC1)	LINSIR		LIDST[2:0]		LBUSY	LERR	LIDOK	LTXOK	LRXOK	133
(0xC0)	LINCR	LSWRES	LIN13	LCONF[1:0]		LENA		LCMD[2:0]		132
(0xBF)	Reserved	–	–	–	–	–	–	–	–	
(0xBE)	Reserved	–	–	–	–	–	–	–	–	
(0xBD)	Reserved	–	–	–	–	–	–	–	–	
(0xBC)	Reserved	–	–	–	–	–	–	–	–	
(0xBB)	Reserved	–	–	–	–	–	–	–	–	
(0xBA)	Reserved	–	–	–	–	–	–	–	–	
(0xB9)	Reserved	–	–	–	–	–	–	–	–	
(0xB8)	Reserved	–	–	–	–	–	–	–	–	
(0xB7)	Reserved	–		–	–	–	–	–	–	
(0xB6)	Reserved	–	–	–	–	–	–	–	–	
(0xB5)	Reserved	–	–	–	–	–	–	–	–	
(0xB4)	Reserved	–	–	–	–	–	–	–	–	
(0xB3)	Reserved	–	–	–	–	–	–	–	–	
(0xB2)	Reserved	–	–	–	–	–	–	–	–	
(0xB1)	Reserved	–	–	–	–	–	–	–	–	
(0xB0)	Reserved	–	–	–	–	–	–	–	–	
(0xAF)	Reserved	–	–	–	–	–	–	–	–	
(0xAE)	Reserved	–	–	–	–	–	–	–	–	
(0xAD)	Reserved	–	–	–	–	–	–	–	–	
(0xAC)	Reserved	–	–	–	–	–	–	–	–	
(0xAB)	Reserved	–	–	–	–	–	–	–	–	
(0xAA)	Reserved	–	–	–	–	–	–	–	–	
(0xA9)	Reserved	–	–	–	–	–	–	–	–	
(0xA8)	Reserved	–	–	–	–	–	–	–	–	
(0xA7)	Reserved	–	–	–	–	–	–	–	–	
(0xA6)	Reserved	–	–	–	–	–	–	–	–	
(0xA5)	Reserved	–	–	–	–	–	–	–	–	
(0xA4)	Reserved	–	–	–	–	–	–	–	–	
(0xA3)	Reserved	–	–	–	–	–	–	–	–	
(0xA2)	Reserved	–	–	–	–	–	–	–	–	
(0xA1)	Reserved	–	–	–	–	–	–	–	–	
(0xA0)	Reserved	–	–	–	–	–	–	–	–	
(0x9F)	Reserved	–	–	–	–	–	–	–	–	
(0x9E)	Reserved	–	–	–	–	–	–	–	–	
(0x9D)	Reserved	–	–	–	–	–	–	–	–	
(0x9C)	Reserved	–	–	–	–	–	–	–	–	
(0x9B)	Reserved	–	–	–	–	–	–	–	–	
(0x9A)	Reserved	–	–	–	–	–	–	–	–	

Notes:
1. For compatibility with future devices, reserved bits should be written to zero if accessed. Reserved I/O memory addresses should never be written.
2. I/O registers within the address range $00 - $1F are directly bit-accessible using the SBI and CBI instructions. In these registers, the value of single bits can be checked by using the SBIS and SBIC instructions.
3. Some of the status flags are cleared by writing a logical one to them. Note that the CBI and SBI instructions will operate on all bits in the I/O register, writing a one back into any flag read as set, thus clearing the flag. The CBI and SBI instructions work with registers 0x00 to 0x1F only.
4. When using the I/O specific commands IN and OUT, the I/O addresses $00 - $3F must be used. When addressing I/O registers as data space using LD and ST instructions, $20 must be added to these addresses. The Atmel AVR MCU is a complex microcontroller with more peripheral units than can be supported within the 64 location reserved in Opcode for the IN and OUT instructions. For the Extended I/O space from $60 - $FF in SRAM, only the ST/STS/STD and LD/LDS/LDD instructions can be used.