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.

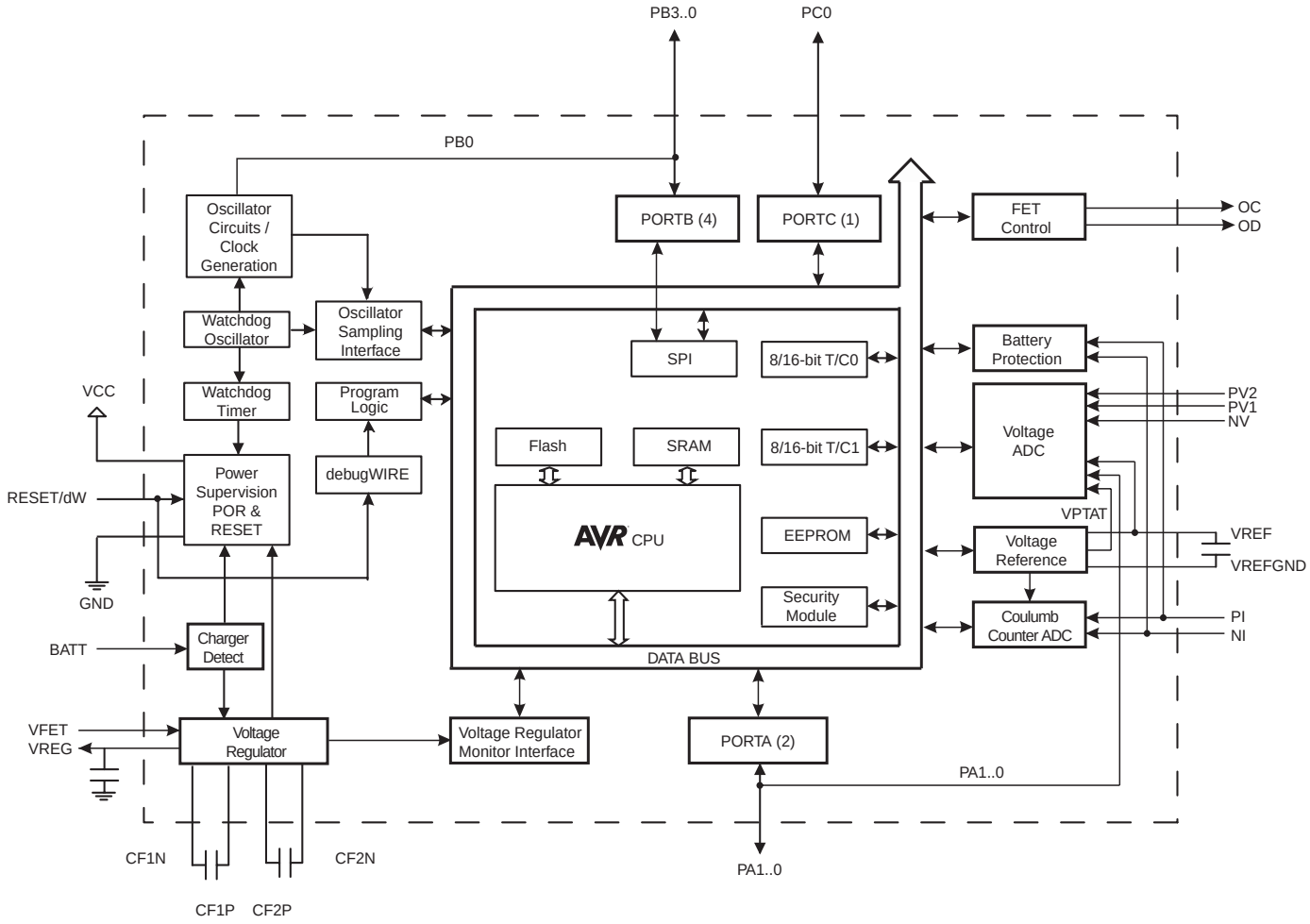
Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	AVR
Core Size	8-Bit
Speed	4MHz
Connectivity	SPI
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	7
Program Memory Size	8KB (4K x 16)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 9V
Data Converters	A/D 5x12b
Oscillator Type	Internal
Operating Temperature	-20°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	36-WFLGA
Supplier Device Package	36-LGA (6.5x3.5)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atmega8hva-4ckur

2. Overview

The ATmega8HVA/16HVA is a monitoring and protection circuit for 1-cell and 2-cell Li-ion applications with focus on high security/authentication, accurate monitoring, low cost and high utilization of the cell energy. The device contains secure authentication features as well as autonomous battery protection during charging and discharging. The chip allows very accurate accumulated current measurements using an 18-bit ADC with a resolution of 0.84 µV. The feature set makes the ATmega8HVA/16HVA a key component in any system focusing on high security, battery protection, accurate monitoring, high system utilization and low cost.

Figure 2-1. Block Diagram



A combined step-up and linear voltage regulator ensures that the chip can operate with supply voltages as low as 1.8V for 1-cell applications. The regulator automatically switches to linear mode when the input voltage is sufficiently high, thereby ensuring a minimum power consumption at all times. For 2-cell applications, only linear regulation is enabled. The regulator capabilities, combined with an extremely low power consumption in the power saving modes, greatly enhances the cell energy utilization compared to existing solutions.

The chip utilizes Atmel's patented Deep Under-voltage Recovery (DUVR) mode that supports pre-charging of deeply discharged battery cells without using a separate Pre-charge FET.

- **Bits 7:6 – Res: Reserved Bits**

These bits are reserved bits in the ATmega8HVA/16HVA and will always read as zero.

- **Bits 5, 4 – EEPM1 and EEPM0: EEPROM Programming Mode Bits**

The EEPROM Programming mode bit setting defines which programming action that will be triggered when writing EEPE. It is possible to program data in one atomic operation (erase the old value and program the new value) or to split the Erase and Write operations in two different operations. The Programming times for the different modes are shown in Table 8-1. While EEPE is set, any write to EEPMn will be ignored. During reset, the EEPMn bits will be reset to 0b00 unless the EEPROM is busy programming.

Table 8-1. EEPROM Mode Bits

EEPM1	EEPM0	Typ Programming Time, f_{osc} = 4.0 MHz	Operation
0	0	3.4 ms	Erase and Write in one operation (Atomic Operation)
0	1	1.8 ms	Erase Only
1	0	1.8 ms	Write Only
1	1	–	Reserved for future use

- **Bit 3 – EERIE: EEPROM Ready Interrupt Enable**

Writing EERIE to one enables the EEPROM Ready Interrupt if the I bit in SREG is set. Writing EERIE to zero disables the interrupt. The EEPROM Ready interrupt generates a constant interrupt when EEPE is cleared.

- **Bit 2 – EEMPE: EEPROM Master Write Enable**

The EEMPE bit determines whether setting EEPE to one causes the EEPROM to be written. When EEMPE is set, setting EEPE within four clock cycles will write data to the EEPROM at the selected address If EEMPE is zero, setting EEPE will have no effect. When EEMPE has been written to one by software, hardware clears the bit to zero after four clock cycles. See the description of the EEPE bit for an EEPROM write procedure.

- **Bit 1 – EEPE: EEPROM Write Enable**

The EEPROM Write Enable Signal EEPE is the write strobe to the EEPROM. When address and data are correctly set up, the EEPE bit must be written to one to write the value into the EEPROM. The EEMPE bit must be written to one before a logical one is written to EEPE, otherwise no EEPROM write takes place. The following procedure should be followed when writing the EEPROM (the order of steps 2 and 3 is not essential):

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

Caution:

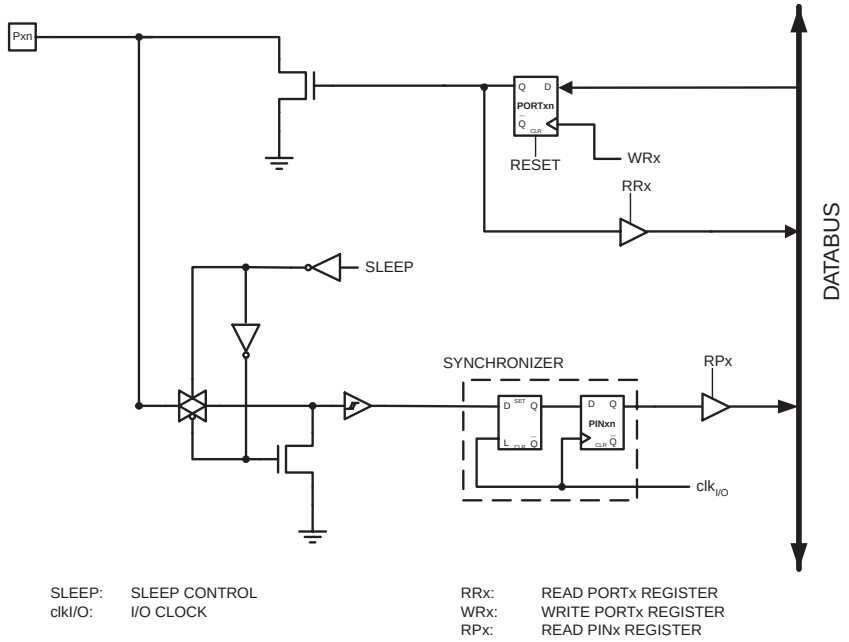
An interrupt between step 4 and step 5 will make the write cycle fail, since the EEPROM Master Write Enable will time-out. If an interrupt routine accessing the EEPROM is interrupting another EEPROM access, the EEAR or EEDR Register will be modified, causing the interrupted

14.2 High Voltage Ports as General Digital I/O

14.3 Overview

The high voltage ports are high voltage tolerant open collector output ports. In addition they can be used as general digital inputs. Figure 14-2 shows a functional description of one output port pin, here generically called Pxn.

Figure 14-2. General High Voltage Digital I/O(1)



Note: 1. WRx, RRx and RPx are common to all pins within the same port. $clk_{I/O}$ and SLEEP are common to all ports.

14.3.1 Configuring the Pin

Each port pin consist of two register bits: PORTxn and PINxn. As shown in ”Register Description” on page 62, the PORTxn bits are accesed at the PORTx I/O address, and the PINxn bits at the PINx I/O address.

If PORTxn is written logic one, the port pin is driven low (zero). If PORTxn is written logic zero, the port pin is tri-stated. The port pins are tri-stated when a reset condition becomes active, even if no clocks are running.

14.3.2 Reading the Pin

The port pin can be read through the PINxn Register bit. As shown in Figure 14-2, the PINxn Register bit and the preceding latch constitute a synchronizer. This is needed to avoid metastability if the physical pin changes value near the edge of the internal clock, but it also introduces a delay.

17.10 Register Description

17.10.1 TCCRnA – Timer/Counter n Control Register A

Bit	7	6	5	4	3	2	1	0	
	TCWn	ICENn	ICNCn	ICESn	ICSn	–	–	WGMn0	TCCRnA
Read/Write	R/W	R/W	R/W	R/W	R/W	R	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7– TCWn: Timer/Counter Width**

When this bit is written to one 16-bit mode is selected. The Timer/Counter width is set to 16-bits and the Output Compare Registers OCRnA and OCRnB are combined to form one 16-bit Output Compare Register. Because the 16-bit registers TCNTnH/L and OCRnB/A are accessed by the AVR CPU via the 8-bit data bus, special procedures must be followed. These procedures are described in section ”Accessing Registers in 16-bit Mode” on page 86.

- **Bit 6– ICENn: Input Capture Mode Enable**

The Input Capture Mode is enabled when this bit is written to one.

- **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.

- **Bit 3 - ICSn: Input Capture Select**

When written to a logic one, this bit selects the alternate Input Capture Source as trigger for the Timer/Counter input capture function. To trigger the Timer/Counter Input Capture interrupt, the ICIEn bit in the Timer Interrupt Mask Register (TIMSK) must be set. See Table 17-3 on page 84 and Table 17-4 on page 84.

- **Bits 2:0 – Res: Reserved Bits**

These bits are reserved bits in the ATmega8HVA/16HVA 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 17-6 on page 85. Modes of operation supported by the Timer/Counter unit are: Normal mode (counter) and Clear Timer on Compare Match (CTC) mode (see ”Timer/Counter Timing Diagrams” on page 85).

Assembly Code Example[1]

```
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

SPI_MasterTransmit:
  ; Start transmission of data (r16)
  out  SPDR,r16
Wait_Transmit:
  ; Wait for transmission complete
  sbis SPSR,SPIF
  rjmp Wait_Transmit
  ret
```

C Code Example[1]

```
void SPI_MasterInit(void)
{
  /* Set MOSI and SCK output, all others input */
  DDR_SPI = (1<<DD_MOSI)|(1<<DD_SCK);
  /* Enable SPI, Master, set clock rate fck/16 */
  SPCR = (1<<SPE)|(1<<MSTR)|(1<<SPR0);
}

void SPI_MasterTransmit(char cData)
{
  /* Start transmission */
  SPDR = cData;
  /* Wait for transmission complete */
  while(!(SPSR & (1<<SPIF)))
    ;
}
```

Note: 1. See “About Code Examples” on page 7.

The following code examples show how to initialize the SPI as a Slave and how to perform a simple reception.

Assembly Code Example[1]

```
SPI_SlaveInit:
  ; Set MISO output, all others input
  ldi   r17,(1<<DD_MISO)
  out   DDR_SPI,r17
  ; Enable SPI
  ldi   r17,(1<<SPE)
  out   SPCR,r17
  ret

SPI_SlaveReceive:
  ; Wait for reception complete
  sbis SPSR,SPIF
  rjmp SPI_SlaveReceive
  ; Read received data and return
  in    r16,SPDR
  ret
```

C Code Example[1]

```
void SPI_SlaveInit(void)
{
  /* Set MISO output, all others input */
  DDR_SPI = (1<<DD_MISO);
  /* Enable SPI */
  SPCR = (1<<SPE);
}

char SPI_SlaveReceive(void)
{
  /* Wait for reception complete */
  while(!(SPSR & (1<<SPIF)))
    ;
  /* Return Data Register */
  return SPDR;
}
```

Note: 1. See “About Code Examples” on page 7.

18.5 Register Description

18.5.1 SPCR – SPI Control Register

Bit	7	6	5	4	3	2	1	0	
0x2C (0x4C)	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	SPCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – SPIE: SPI Interrupt Enable**

This bit causes the SPI interrupt to be executed if SPIF bit in the SPSR Register is set and the if the Global Interrupt Enable bit in SREG is set.

- **Bit 6 – SPE: SPI Enable**

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

- **Bit 5 – DORD: Data Order**

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

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

- **Bit 4 – MSTR: Master/Slave Select**

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

- **Bit 3 – CPOL: Clock Polarity**

When this bit is written to one, SCK is high when idle. When CPOL is written to zero, SCK is low when idle. Refer to Figure 18-3 and Figure 18-4 for an example. The CPOL functionality is summarized below:

Table 18-3. CPOL Functionality

CPOL	Leading Edge	Trailing Edge
0	Rising	Falling
1	Falling	Rising

- **Bit 2 – CPHA: Clock Phase**

The settings of the Clock Phase bit (CPHA) determine if data is sampled on the leading (first) or trailing (last) edge of SCK. Refer to Figure 18-3 and Figure 18-4 for an example. The CPOL functionality is summarized below:

Table 18-4. CPHA Functionality

CPHA	Leading Edge	Trailing Edge
0	Sample	Setup
1	Setup	Sample

20. Voltage ADC – 5-channel General Purpose 12-bit Sigma-Delta ADC

20.1 Features

- **12-bit Resolution**
- **519µs Conversion Time @ 1 MHz clk_{VADC}**
- **Two Differential Input Channels for Cell Voltage Measurements**
- **Three Single Ended Input Channels**
- **0.2x Pre-scaling of Cell Voltages**
- **Interrupt on V-ADC Conversion Complete**

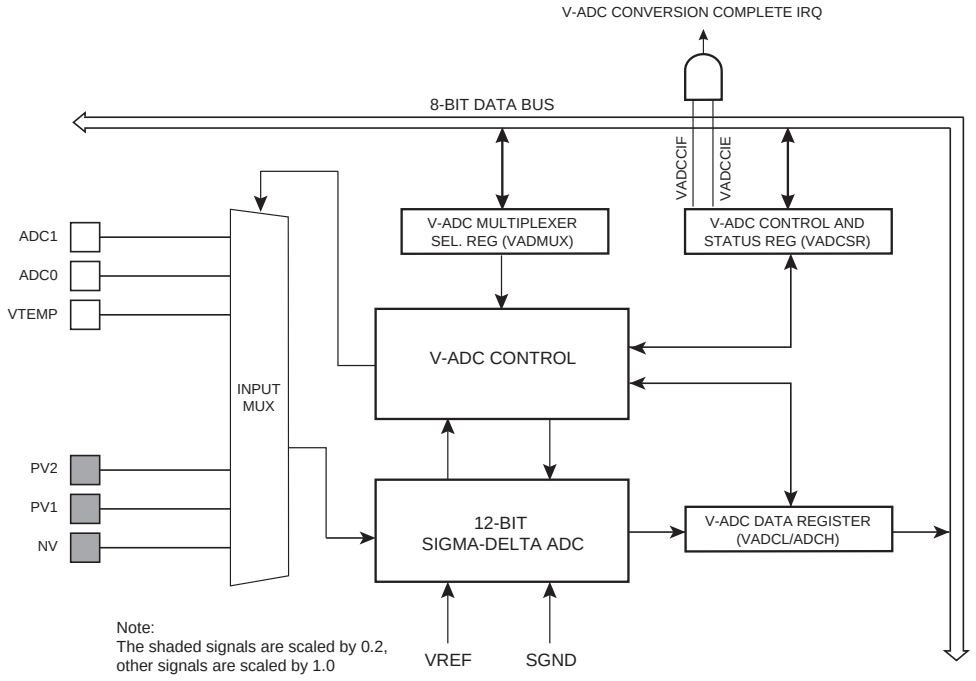
20.2 Overview

The ATmega8HVA/16HVA features a 12-bit Sigma-Delta ADC.

The Voltage ADC (V-ADC) is connected to five different sources through the Input Multiplexer. There are two differential channels for Cell Voltage measurements. These channels are scaled 0.2x to comply with the Full Scale range of the V-ADC. In addition there are three single ended channels referenced to SGND. One channel is for measuring the internal temperature sensor VPTAT and two channels for measuring the voltage at ADC0 and ADC1.

When the V-ADC is not used, power consumption can be minimized by writing the PRVADC bit in PRR0 to one. See "PRR0 – Power Reduction Register 0" on page 39 for details on how to use the PRVADC bit.

Figure 20-1. Voltage ADC Block Schematic



20.3 Operation

To enable V-ADC conversions, the V-ADC Enable bit, VADEN, in V-ADC Control and Status Register – VADCSR must be set. If this bit is cleared, the V-ADC will be switched off, and any ongoing conversions will be terminated. The V-ADC is automatically disabled in Power-save and

VADSC will read as one as long as the conversion is not finished. When the conversion is complete, it returns to zero. Writing zero to this bit has no effect. VADSC will automatically be cleared when the VADEN bit is written to zero.

- **Bit 1 – VADCCIF: V-ADC Conversion Complete Interrupt Flag**

This bit is set when a V-ADC conversion completes and the data registers are updated. The V-ADC Conversion Complete Interrupt is executed if the VADCCIE bit and the I-bit in SREG are set. VADCCIF is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, VADCCIF is cleared by writing a logical one to the flag. Beware that if doing a Read-Modify-Write on VADCSR, a pending interrupt can be lost.

- **Bit 0 – VADCCIE: V-ADC Conversion Complete Interrupt Enable**

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

20.4.3 VADCL and VADCH – V-ADC Data Register

Bit	15	14	13	12	11	10	9	8	
(0x79)	–	–	–	–	VADC[11:8]				VADCH
(0x78)	VADC[7:0]								VADCL
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R	R	R	R	
	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

When a V-ADC conversion is complete, the result is found in these two registers. To ensure that correct data is read, both high and low byte data registers should be read before starting a new conversion.

- **VADC11:0: V-ADC Conversion Result**

These bits represent the result from the conversion.

To obtain the best absolute accuracy for the cell voltage measurements, gain and offset compensation is required. Factory calibration values are stored in the device signature row, refer to section "Reading the Signature Row from Software" on page 144 for details. The cell voltage in mV is given by:

$$Cell_n[\text{mV}] = \frac{(\text{VADCH/L} - \text{VADC Cell}_n\text{Offset}) \cdot \text{VADC Cell}_n\text{ Gain Calibration Word}}{16384}$$

The voltage on the ADCn is given by:

$$\text{ADCn[mV]} = \frac{1}{10} \cdot \frac{(\text{VADCH/L} - \text{VADC ADCn Offset}) \cdot \text{VADC ADCn Gain Calibration Word}}{16384}$$

When performing a VTEMP conversion, the result must be adjusted by the factory calibration value stored in the signature row, refer to section "Reading the Signature Row from Software" on page 144 for details. The absolute temperature in Kelvin is given by:

21. Voltage Reference and Temperature Sensor

21.1 Features

- **Accurate Voltage Reference of 1.100V**
- **Internal Temperature Sensor**
- **Possibility for Runtime Compensation of Temperature Drift in Both Voltage Reference and On-chip Oscillators**
- **External Decoupling for Optimum Noise Performance**
- **Low Power Consumption**

21.2 Overview

A low power band-gap reference provides ATmega8HVA/16HVA with an accurate On-chip voltage reference V_{REF} of 1.100V. This reference voltage is used as reference for the On-chip Voltage Regulator, the V-ADC and the CC-ADC. The reference to the ADCs uses a buffer with external decoupling capacitor to enable excellent noise performance with minimum power consumption. The reference voltage V_{REF_P}/V_{REF_N} to the CC-ADC is scaled to match the full scale requirement at the current sense input pins. This configuration also enables concurrent operation of both V-ADC and CC-ADC.

To guarantee ultra low temperature drift after factory calibration, ATmega8HVA/16HVA features a two-step calibration algorithm. The first step is performed at T_{HOT}°C and the second at room temperature. By default, Atmel factory calibration is performed at T_{HOT}°C, and the result is stored in the signature row. The value of T_{HOT} can also be found in the signature row. See ”Reading the Signature Row from Software” on page 144 for details. The customer can easily implement the second calibration step in their test flow. This requires an accurate input voltage and a stable room temperature. Temperature drift after this calibration is guaranteed by design and characterization to be less than 90 ppm/°C from -10°C to 70°C. The BG Calibration C Register can also be altered runtime to implement temperature compensation in software. Very high accuracy for any temperature inside the temperature range can thus be achieved at the cost of extra calibration steps.

ATmega8HVA/16HVA has an On-chip temperature sensor for monitoring the die temperature. A voltage Proportional-To-Absolute-Temperature, V_{PTAT}, is generated in the voltage reference circuit and connected to the multiplexer at the V-ADC input. This temperature sensor can be used for runtime compensation of temperature drift in both the voltage reference and the On-chip Oscillator. To get the absolute temperature in degrees Kelvin, the measured V_{PTAT} voltage must be scaled with the VPTAT factory calibration value stored in the signature row. See Section “26.2.5” on page 144. for details.

22. Voltage Regulator

22.1 Features

- **3.3V fixed output voltage**
- **Automatic selection of Step-up or Linear Regulation depending on VFET voltage.**
- **Fixed Linear Regulation mode can be selected for 2-cell applications**
- **Battery Pack Short mode allowing large voltage drop at VFET without pulling VREG low.**

22.2 Overview

The Voltage Regulator is a Combined Step-up and Linear Voltage Regulator. This allows the same Voltage Regulator module to be used efficiently for a large range of input voltages.

A built in Charge-pump with external capacitors is combined with a linear regulator to keep a constant output voltage for input voltages in the range 1.8 - 9.0V.

Figure 22-1 on page 121 shows the Voltage Regulator block diagram with external components for combined Step-up and Linear mode. Figure 22-2 on page 121 shows the regulated voltage VREG as a function of the input voltage VFET for 1-cell operation. When the VFET is sufficiently high, the regulator switches automatically to linear operation. When VFET drops below a certain level the regulator automatically switch back to step-up regulation. The different reset sources during initialisation and shut down is also shown.

Figure 22-3 on page 122 shows the Voltage Regulator block diagram with external components for Linear mode only, intended for 2-cell applications. In Linear mode only, the input voltage range is 3.6 - 9.0V. In this case, no external fly capacitors are needed, and CF1N should be grounded. Figure 22-4 on page 122 illustrates this operation.

In case of battery pack shortening, the voltage at the input of the regulator will drop quickly. If it drops below minimum operating voltage, the voltage regulator can no longer supply internal or external circuitry. However, the output voltage will not be pulled down by this incident, and the external CREG capacitor can supply the circuitry for a time given by the size of the capacitor and the total current consumption during the same period. VREG must stay above the Brown-Out Threshold to avoid BOD reset. If a battery pack short occurs when VREG is equal to 3.3V and the BOD level is 2.9V, the chip can continue operation for a time given by:

$$t = \frac{c\Delta v}{I_{AVG}} = \frac{CREG \cdot 0.4V}{I_{AVG}}$$

where I_{AVG} represents the average current drawn from CREG. For CREG = 2.2 µF and I_{AVG} = 100 µA, this time equals 8.8 ms. The Voltage Regulator Monitor will detect if a short-circuit has occured, allowing SW to minimize I_{AVG}.

When charging deeply over-discharged cells, the FET Driver will be operated in Deep Under-Voltage Recovery (DUVR) mode. See “FET Driver” on page 136. In this mode a suitable voltage drop is developed across the Charge FET to ensure proper operating voltage at the VFET pin. This will ensure normal operation of the chip during 0-volt charging without setting the charger in quick-charge mode before the cell has reached a safe cell voltage.

applies when enabling the Discharge FET. For Charge Over-Current protection, this applies when enabling the Charge FET. With nominal ULP frequency this delay is maximum 0.1 ms.

Note: Due to synchronization of parameters between clock domains, a guard time of 3 ULP oscillator cycles + 3 CPU clock cycles is required between each time the BPOCTR register is written. Any writing to the BPOCTR register during this period will be ignored.

23.9.5 BPHCTR – Battery Protection High-current Timing Register

Bit	7	6	5	4	3	2	1	0	
(0xFC)	–	–	HCPT[5:0]						BPHCTR
Read/Write	R	R	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	1	

- **Bit 7:6 – Res: Reserved Bits**

These bits are reserved and will always read as zero.

- **Bit 5:0 – HCPT5:0: High-current Protection Timing**

These bits control the delay of the High-circuit Protection. The High-current Timing can be set with a step size of 2 ms as shown in Table 23-4 on page 130.

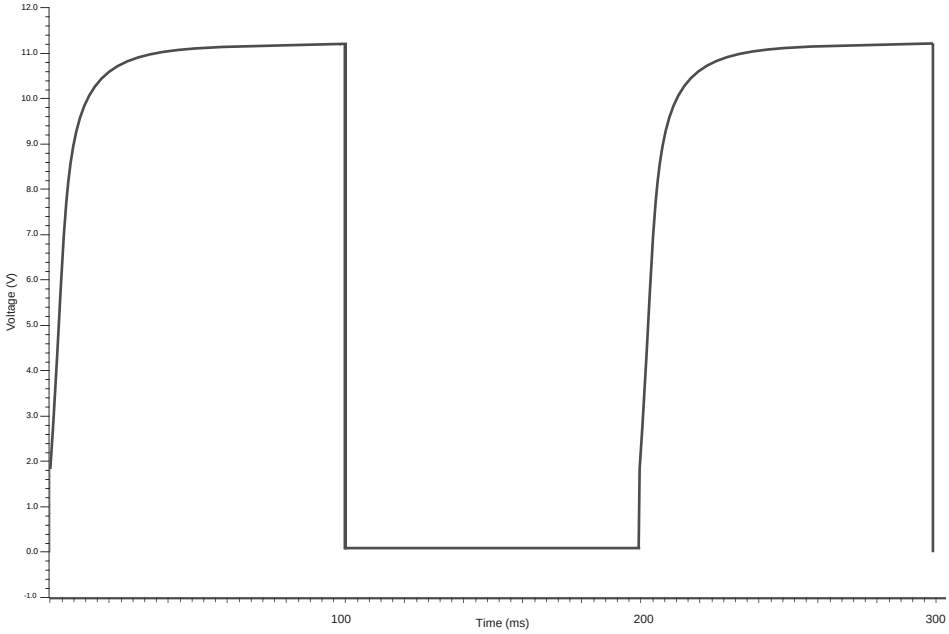
Table 23-4. High-current Protection Reaction Time. HCPT[5:0] with corresponding High-current Delay Time.

High-current Protection Reaction Time(1)	
HCPT[5:0]	**Typ**
0x00	(0 - 2 ms) + T_d(3)
0x01(2)	(0 - 2 ms) + T_d(3)
0x02	(2 - 4 ms) + T_d(3)
0x03	(4 - 6 ms) + T_d(3)
...	...
0x3E	(122 - 124 ms) + T_d(3)
0x3F	(124 - 126 ms) + T_d(3)

Notes:
1. The actual value depends on the actual frequency of the "Ultra Low Power RC Oscillator" on page 26. See "Electrical Characteristics" on page 165.
2. Initial value.
3. An additional delay T_d can be expected after enabling the corresponding FET. This is related to the initialization of the protection circuitry. For the Discharge High-Current protection, this applies when enabling the Discharge FET. For Charge High-Current protection, this applies when enabling the Charge FET. With nominal ULP frequency this delay is maximum 0.2 ms.

Note: Due to synchronization of parameters between clock domains, a guard time of 3 ULP oscillator cycles + 3 CPU clock cycles is required between each time the BPHCTR register is written. Any writing to the BPHCTR register during this period will be ignored.

Figure 24-3. Switching NFET on and off during NORMAL operation



24.3 DUVR – Deep Under-Voltage Recovery Mode operation

The purpose of DUVR mode is to control the Charge FET so that the VFET voltage is above the minimum operating voltage while charging cells below minimum operating voltage. This is useful when the cell has been discharged below the minimum operating voltage of the chip. In DUVR mode the Charge FET is switched partly on to provide a suitable voltage drop between the cell voltage and the VFET terminal. As the cell voltage increases, the voltage drop across the Charge FET will gradually decrease until the Charge FET is switched completely on. This means that for high cell voltages, DUVR mode operation is equivalent to normal enabling of the Charge FET (CFE=1).

ATmega8HVA/16HVA should operate in DUVR mode until software detects that the cell has recovered from Deep Under-Voltage condition. When the cell has recovered from Deep Under-Voltage condition, software should first set CFE=1. This is safe now since the cell voltage is above minimum operating voltage. After that software should disable DUVR mode by setting DUVRD = 1.

If both DUVRD and CFE bit is set before the cell voltage is above minimum operating voltage, the VFET voltage will drop and the chip will enter BOD reset and switch off both the Charge- and Discharge FET. Switching off the FET's will cause the VFET voltage to rise again so that the chip restarts from BOD reset, with DUVRD = 0 and CFE = 0 (default values). To avoid this, software must always check the cell voltage by V-ADC measurements before setting CFE=1.

DUVR mode is default enabled after reset. However, while the chip is in reset state, DUVR mode is disabled. This is a safety feature that ensures that the Charge FET will not be switched on until the Charge Over-current Protection is operating. This implies that the DUVR mode will be disabled from the time that a charger is connected until the selected start-up time expired. During this period, the VFET voltage will be higher than the normal VFET Level in DUVR mode.

For more details about DUVR mode, refer to application note AVR354.

26. Self-Programming the Flash

26.1 Overview

The device provides a Self-Programming mechanism for downloading and uploading program code by the MCU itself. The Self-Programming can use any available data interface and associated protocol to read code and write (program) that code into the Program memory.

The Program memory is updated in a page by page fashion. Before programming a page with the data stored in the temporary page buffer, the page must be erased. The temporary page buffer is filled one word at a time using SPM and the buffer can be filled either before the Page Erase command or between a Page Erase and a Page Write operation:

Alternative 1, fill the buffer before a Page Erase

- Fill temporary page buffer
- Perform a Page Erase
- Perform a Page Write

Alternative 2, fill the buffer after Page Erase

- Perform a Page Erase
- Fill temporary page buffer
- Perform a Page Write

If only a part of the page needs to be changed, the rest of the page must be stored (for example in the temporary page buffer) before the erase, and then be re-written. When using alternative 1, the Boot Loader provides an effective Read-Modify-Write feature which allows the user software to first read the page, do the necessary changes, and then write back the modified data. If alternative 2 is used, it is not possible to read the old data while loading since the page is already erased. The temporary page buffer can be accessed in a random sequence. It is essential that the page address used in both the Page Erase and Page Write operation is addressing the same page.

26.1.1 Performing Page Erase by SPM

To execute Page Erase, set up the address in the Z-pointer, write “00000011” to SPMCSR and execute SPM within four clock cycles after writing SPMCSR. The data in R1 and R0 is ignored. The page address must be written to PCPAGE in the Z-register. Other bits in the Z-pointer will be ignored during this operation.

- The CPU is halted during the Page Erase operation.

26.1.2 Filling the Temporary Buffer (Page Loading)

To write an instruction word, set up the address in the Z-pointer and data in R1:R0, write “00000001” to SPMCSR and execute SPM within four clock cycles after writing SPMCSR. The content of PCWORD in the Z-register is used to address the data in the temporary buffer. The temporary buffer will auto-erase after a Page Write operation or by writing the CTPB bit in SPMCSR. It is also erased after a system reset. Note that it is not possible to write more than one time to each address without erasing the temporary buffer.

If the EEPROM is written in the middle of an SPM Page Load operation, all data loaded will be lost.

Table 27-14. High-voltage Serial Programming Instruction Set for ATmega8HVA/16HVA (Continued)

Instruction		Instruction Format				Operation Remarks
		Instr.1/5	Instr.2/6	Instr.3	Instr.4	
Read EEPROM Byte	SDI SII SDO	0_**bbbb_bbbb**_00 0_0000_1100_00 x_xxxx_xxxx_xx	0_**aaaa_aaaa**_00 0_0001_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1000_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1100_00 **q_qqqq_qqq**0_00	
Write Fuse High Byte	SDI SII SDO	0_0100_0000_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_**hhhh_hhhh**_00 0_0010_1100_11 x_xxxx_xxxx_xx	0_0000_0000_00 0_0111_0100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0111_1100_00 x_xxxx_xxxx_xx	Wait after Instr. 4 until SDO goes high. Write “0” to program the Fuse Bits.
Write Fuse Low Byte	SDI SII SDO	0_0100_0000_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_**llll_llll**_00 0_0010_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_0100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1100_00 x_xxxx_xxxx_xx	Wait after Instr. 4 until SDO goes high. Write “0” to program the Fuse bit.
Write Lock Bit Byte	SDI SII SDO	0_0010_0000_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_cccc_cccc_00 0_0010_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_0100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1100_00 x_xxxx_xxxx_xx	Wait after Instr. 4 until SDO goes high. Write “0” to program the Lock Bit.
Read Fuse High Byte	SDI SII SDO	0_0000_0100_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0111_1000_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0111_1100_0 h_hhhh_hhhx_xx		Reading “0” means the Fuse bit is programmed.
Read Fuse Low Byte	SDI SII SDO	0_0000_0100_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1000_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1100_00 **l_llll_lll**x_xx		Reading “0” means the Fuse bit is programmed.
Read Lock Bit Byte	SDI SII SDO	0_0000_0100_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0111_1000_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0111_1100_00 c_cccc_cccx_xx		Reading “0” means the Lock bit is programmed.
Read Signature Row Low Byte	SDI SII SDO	0_0000_1000_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_**bbbb_bbbb**_00 0_0000_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1000_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0110_1100_00 **q_qqqq_qqq**x_xx	Repeats Instr 2 4 for each signature low byte address.
Read Signature Row High Byte	SDI SII SDO	0_0000_1000_00 0_0100_1100_00 x_xxxx_xxxx_xx	0_**aaaa_aaaa**_00 0_0001_1100_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0111_1000_00 x_xxxx_xxxx_xx	0_0000_0000_00 0_0111_1100_00 **p_pppp_ppp**x_xx	Repeats Instr 2 4 for each signature high byte address.
Load “No Operation” Command	SDI SII SDO	0_0000_0000_00 0_0100_1100_00 x_xxxx_xxxx_xx				

Note: 1. **a** = address high bits, **b** = address low bits, **d** = data in high bits, **e** = data in low bits, **p** = data out high bits, **q** = data out low bits, x = don’t care, **c** = Lock Bit Byte, **l** = fuse low byte, **h** = fuse high byte.

Notes: 1. For page sizes less than 256 words, parts of the address (bbbb_bbbb) will be parts of the page address.
2. For page sizes less than 256 bytes, parts of the address (bbbb_bbbb) will be parts of the page address.

29. Electrical Characteristics

29.1 Absolute Maximum Ratings*

Operating Temperature	-20°C to +85°C
Storage Temperature	-65°C to +150°C
Voltage on PA0 - PA1, PI, and NI with respect to Ground	-0.5V to V_{REG} +0.5V
Voltage on PB0 - PB3 with respect to Ground	-0.5V to VCC +0.5V
Voltage on PC0 with respect to Ground	-0.5V to + 6.0V
Voltage on VFET with respect to Ground	-0.5V to + 18V
Voltage on OD, OC, BATT, and $\overline{\text{RESET}}$ with respect to Ground	-0.5V to + 13V
Voltage on NV, PV1, and PV2 with respect to Ground	-0.5V to VFET + 1.0V
Maximum Operating Voltage on VREG and VCC	4.5V
Maximum Operating Voltage on VFET	9V

*NOTICE: Stresses beyond those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

29.4 General I/O Lines characteristics

Table 29-3. [1] T_A = -10°C to 70°C, V_{CC} = 3.3V

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
V_{IL}	Input Low Voltage, Except RESET pin		-0.5		$0.3V_{CC}$ [2]	V
V_{IL1}	Input Low Voltage, RESET pin				$0.3V_{CC}$ [2]	
V_{IH}	Input High Voltage, Except RESET pin		$0.6V_{CC}$ [3]		V_{CC} + 0.5	V
V_{IH1}	Input High Voltage, RESET pin		$0.9V_{CC}$ [3]		V_{CC} + 0.5	V
V_{OL}	Output Low Voltage	I_{OL} = 5mA			0.5	V
V_{OH}	Output High Voltage	I_{OH} = 2 mA	2.3			V
I_{IL}	Input Leakage Current I/O Pin	Pin low (absolute value)			1	µA
I_{IH}	Input Leakage Current I/O Pin	Pin high (absolute value)			1	µA
R_{RST}	Reset Pull-up Resistor		30		60	kΩ
R_{PU}	I/O Pin Pull-up Resistor		20		50	kΩ

Notes:
1. Applicable for all except PC0.
2. “Max” means the highest value where the pin is guaranteed to be read as low
3. “Min” means the lowest value where the pin is guaranteed to be read as high
4. Although each I/O port can sink more than the test conditions (5 mA at V_{CC} = 3.3V) under steady state conditions (non-transient), the following must be observed:
   - The sum of all IOL should not exceed 20 mA.
   If IOL exceeds the test condition, VOL may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test condition.
5. Although each I/O port can source more than the test conditions (2 mA at V_{CC} = 3.3V) under steady state conditions (non-transient), the following must be observed:
   - The sum of all IOH should not exceed 2 mA.

Table 29-4. PC0 Characteristics (T_A = -10°C to 70°C unless otherwise specified)

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low-voltage		-0.5	0.8 [1]	V
V_{IH}	Input High-voltage		2.1 [2]	5.5	V
V_{OL}	Output Low-voltage	350 µA sink current	0	0.4	V
t_r [3]	Rise Time			300	ns
t_{of} [3]	Output Fall Time from V_{IHmin} to V_{ILmax}	C_b < 400 pF [4]		250	ns
t_{SP} [3]	Spikes Suppressed by Input Filter		0	50	ns
I_i [3]	Input Current	$0.1V_{BUS} < V_i < 0.9V_{BUS}$	-5	5	µA
C_i [3]	Capacitance			10	pF

Notes:
1. “Max” means the highest value where the pin is guaranteed to be read as low
2. “Min” means the lowest value where the pin is guaranteed to be read as high
3. This value is not tested in production.
4. C_b = capacitance of one bus line in pF

29.7 SPI Timing Characteristics

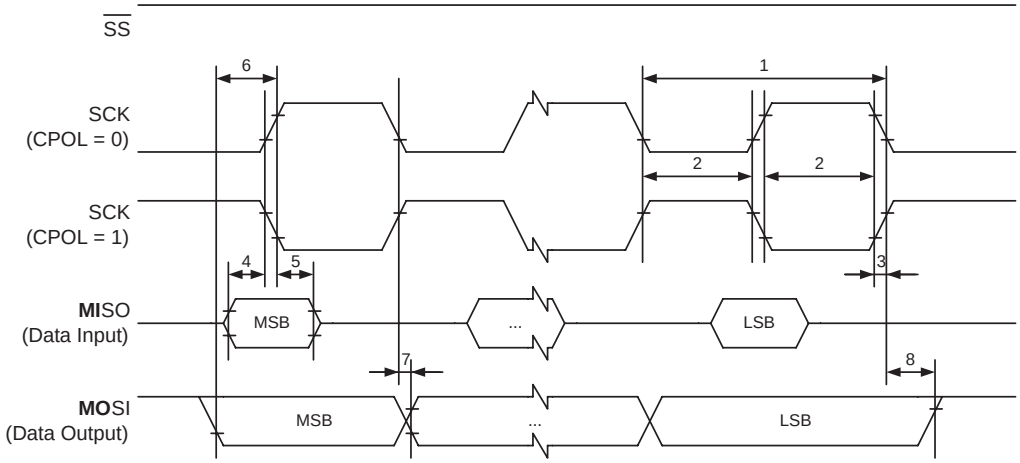
See Figure 29-1 on page 171 and Figure on page 172 for details.

Table 29-7. SPI Timing Parameters

	Description	Mode	Min	Typ	Max	Units
1	SCK period	Master		See Figure		ns
2	SCK high/low	Master		50% duty		
3	Rise/Fall time	Master		3.6		
4	Setup	Master		10		
5	Hold	Master		10		
6	Out to SCK	Master		0.5 • t_{sck}		
7	SCK to out	Master		10		
8	SCK to out high	Master		10		
9	$\overline{SS}$ low to out	Slave		15		
10	SCK period	Slave	4 • t_{ck} + 40 ns			
11	SCK high/low(1)	Slave	2 • t_{ck} + 20 ns			
12	Rise/Fall time	Slave		1.6		µs
13	Setup	Slave	10			ns
14	Hold	Slave	t_{ck}			
15	SCK to out	Slave		15		
16	SCK to $\overline{SS}$ high	Slave	20			
17	$\overline{SS}$ high to tri-state	Slave		10		
18	$\overline{SS}$ low to SCK	Slave	20			

Note: 1. Refer to "Serial Programming" on page 151 for serial programming requirements.

Figure 29-1. SPI Interface Timing Requirements (Master Mode)



Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
0x1B (0x3B)	Reserved	–	–	–	–	–	–	–	–	
0x1A (0x3A)	Reserved	–	–	–	–	–	–	–	–	
0x19 (0x39)	Reserved	–	–	–	–	–	–	–	–	
0x18 (0x38)	Reserved	–	–	–	–	–	–	–	–	
0x17 (0x37)	OSICSR	–	–	–	OSISEL0	–	–	OSIST	OSIEN	32
0x16 (0x36)	TIFR1	–	–	–	–	ICF1	OCF1B	OCF1A	TOV1	93
0x15 (0x35)	TIFR0	–	–	–	–	ICF0	OCF0B	OCF0A	TOV0	93
0x14 (0x34)	Reserved	–	–	–	–	–	–	–	–	
0x13 (0x33)	Reserved	–	–	–	–	–	–	–	–	
0x12 (0x32)	Reserved	–	–	–	–	–	–	–	–	
0x11 (0x31)	Reserved	–	–	–	–	–	–	–	–	
0x10 (0x30)	Reserved	–	–	–	–	–	–	–	–	
0x0F (0x2F)	Reserved	–	–	–	–	–	–	–	–	
0x0E (0x2E)	Reserved	–	–	–	–	–	–	–	–	
0x0D (0x2D)	Reserved	–	–	–	–	–	–	–	–	
0x0C (0x2C)	Reserved	–	–	–	–	–	–	–	–	
0x0B (0x2B)	Reserved	–	–	–	–	–	–	–	–	
0x0A (0x2A)	Reserved	–	–	–	–	–	–	–	–	
0x09 (0x29)	Reserved	–	–	–	–	–	–	–	–	
0x08 (0x28)	PORTC	–	–	–	–	–	–	–	PORTC0	62
0x07 (0x27)	Reserved	–	–	–	–	–	–	–	–	
0x06 (0x26)	PINC	–	–	–	–	–	–	–	PINC0	62
0x05 (0x25)	PORTB	–	–	–	–	PORTB3	PORTB2	PORTB1	PORTB0	73
0x04 (0x24)	DDRB	–	–	–	–	DDB3	DDB2	DDB1	DDB0	73
0x03 (0x23)	PINB	–	–	–	–	PINB3	PINB2	PINB1	PINB0	73
0x02 (0x22)	PORTA	–	–	–	–	–	–	PORTA1	PORTA0	73
0x01 (0x21)	DDRA	–	–	–	–	–	–	DDA1	DDA0	73
0x00 (0x20)	PINA	–	–	–	–	–	–	PINA1	PINA0	73

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 ATmega8HVA/16HVA 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.