

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](#)"

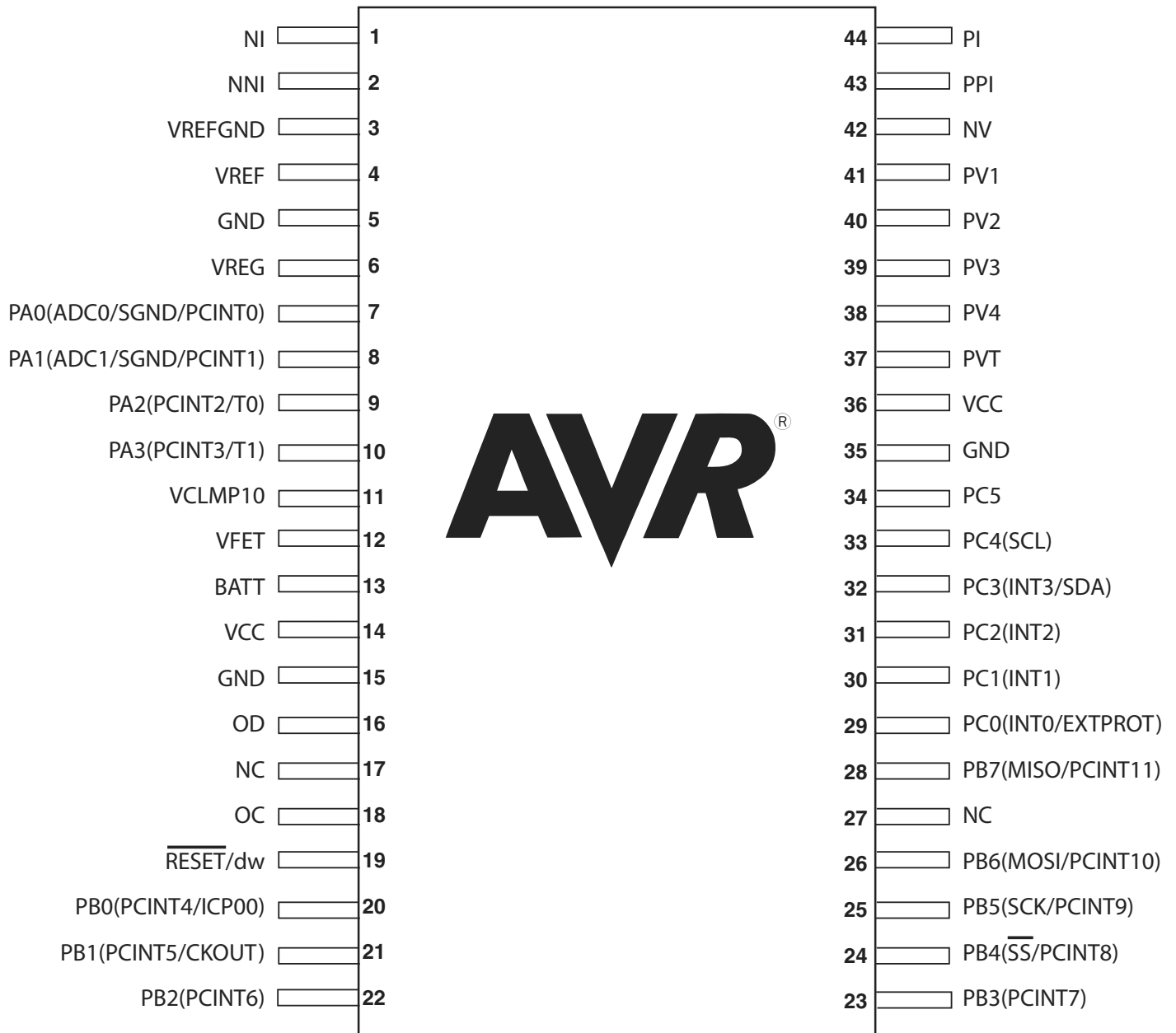
Details

Product Status	Obsolete
Core Processor	AVR
Core Size	8-Bit
Speed	8MHz
Connectivity	I ² C, SPI
Peripherals	POR, WDT
Number of I/O	17
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)	4V ~ 25V
Data Converters	A/D 7x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-TFSOP (0.173", 4.40mm Width)
Supplier Device Package	44-TSSOP
Purchase URL	https://www.e-xfl.com/product-detail/atmel/atmega16hvb-8x3

1. Pin configurations

1.1 TSSOP

Figure 1-1. TSSOP - pinout the Atmel ATmega16HVB/32HVB.



1.2 Pin descriptions

1.2.1 VFET

High voltage supply pin. This pin is used as supply for the internal voltage regulator, described in "Voltage regulator" on page 129.

1.2.2 VCLMP10

Internal 10V clamping of VFET voltage for external decoupling.

1.2.3 VCC

Digital supply voltage. Normally connected to VREG.

1.2.4 VREG

Output from the internal voltage regulator. Used for external decoupling to ensure stable regulator operation. For details, see "Voltage regulator" on page 129.

1.2.5 VREF

Internal voltage reference for external decoupling. For details, see "Voltage reference and temperature sensor" on page 122.

1.2.6 VREFGND

Ground for decoupling of internal voltage reference. For details, see "Voltage reference and temperature sensor" on page 122. Do not connect to GND or SGND on PCB.

1.2.7 GND

Ground.

1.2.8 Port A (PA3..PA0)

Port A serves as a low-voltage 4-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). As inputs, Port A pins that are externally pulled low will source current if the pull-up resistors are activated. The Port A pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port A also serves the functions of various special features of the Atmel ATmega16HVB/32HVB as listed in "Alternate functions of Port A" on page 74.

1.2.9 Port B (PB7..PB0)

Port B is a low-voltage 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port B also serves the functions of various special features of the ATmega16HVB/32HVB as listed in "Alternate functions of Port B" on page 75.

1.2.10 Port C (PC5)

Port C (PC5) is a high voltage Open Drain output port.

1.2.11 Port C (PC4..PC0)

Port C is a 5-bit high voltage Open Drain bi-directional I/O port.

1.2.12 OC/OD

High voltage output to drive Charge/Discharge FET. For details, see "FET driver" on page 145.

1.2.13 PI/NI

Filtered positive/negative input from external current sense resistor, used to by the Coulomb Counter ADC to measure charge/discharge currents flowing in the battery pack. For details, see "Coulomb counter – Dedicated fuel gauging Sigma-Delta ADC" on page 108.

1.2.14 PPI/NNI

Unfiltered positive/negative input from external current sense resistor, used by the battery protection circuit, for over-current and short-circuit detection. For details, see "Battery protection" on page 132.

1.2.15 NV/PV1/PV2/PV3/PV4

NV, PV1, PV2, PV3, and PV4 are the inputs for battery cells one, two, three and four, used by the Voltage ADC to measure each cell voltage. For details, see "Voltage ADC – 7-channel general purpose 12-bit Sigma-Delta ADC" on page 116.

1.2.16 PVT

Defines the source voltage level for the Charge FET driver. For details, see "FET driver" on page 145.

1.2.17 BATT

Input for detecting when a charger is connected. Defines the source voltage level for the Discharge FET driver. For details, see "FET driver" on page 145.

1.2.18 RESET/dw

Reset input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in Table 32-3 on page 227. Shorter pulses are not guaranteed to generate a reset. This pin is also used as debugWIRE communication pin.

nominally 3.3 volts for the integrated logic and analog functions. 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 the Atmel patented Deep Under-voltage Recovery (DUVR) mode that supports pre-charging of deeply discharged battery cells without using a separate Pre-charge FET. DUVR mode cannot be used in 2-cell applications. Optionally, Pre-charge FETs are supported for integration into many existing battery charging schemes.

The battery protection monitors the charge and discharge current to detect illegal conditions and protect the battery from these when required. A 12-bit Voltage ADC allows software to monitor each cell voltage individually with high accuracy. The ADC also provides one internal input channel to measure on-chip temperature and two input channels intended for external thermistors. An 18-bit ADC optimized for Coulomb Counting accumulates charge and discharge currents and reports accumulated current with high resolution and accuracy. It can also be used to provide instantaneous current measurements with 13-bit resolution. Integrated Cell Balancing FETs allow cell balancing algorithms to be implemented in software.

The MCU provides the following features: 16K/32Kbytes of In-System Programmable Flash with Read-While-Write capabilities, 512/1Kbytes EEPROM, 1K/2Kbytes SRAM. 32 general purpose working registers, 12 general purpose I/O lines, five general purpose high voltage open drain I/O lines, one general purpose super high voltage open drain output, debugWIRE for on-chip debugging and SPI for In-system Programming, a SM-Bus compliant TWI module, two flexible Timer/Counters with Input Capture and compare modes.

Internal and external interrupts, a 12-bit Sigma Delta ADC for voltage and temperature measurements, a high resolution Sigma Delta ADC for Coulomb Counting and instantaneous current measurements, integrated cell balancing FETs, Additional Secure Authentication Features, an autonomous Battery Protection module, a programmable Watchdog Timer with internal Oscillator, and software selectable power saving modes.

The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The device is manufactured using the Atmel high voltage high density non-volatile memory technology. The On-chip ISP Flash allows the program memory to be reprogrammed In-System, through an SPI serial interface, by a conventional non-volatile memory programmer or by an On-chip Boot program running on the AVR core. The Boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash section will continue to run while the Application Flash section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash and highly accurate analog front-end in a monolithic chip.

The Atmel ATmega16HVB/32HVB is a powerful microcontroller that provides a highly flexible and cost effective solution. It is part of the AVR Battery Management family that provides secure authentication, highly accurate monitoring and autonomous protection for Lithium-ion battery cells.

The ATmega16HVB/32HVB AVR is supported with a full suite of program and system development tools including: C Compilers, Macro Assemblers, Program Debugger/Simulators, and On-chip Debugger.

2.1 Comparison between the Atmel ATmega16HVB and the Atmel ATmega32HVB

The ATmega16HVB and the ATmega32HVB differ only in memory size for flash, EEPROM and internal SRAM. Table 2-1 summarizes the different configuration for the two devices.

Table 2-1. Configuration summary.

Device	Flash	EEPROM	SRAM
ATmega16HVB	16K	512	1K
ATmega32HVB	32K	1K	2K

3. Disclaimer

All parameters contained in this datasheet are preliminary and based on characterization of the Atmel ATmega16/32HVB.

4. Resources

A comprehensive set of development tools, application notes and datasheets are available for download on <http://www.atmel.com/avr>.

5. About code examples

This documentation contains simple code examples that briefly show how to use various parts of the device. These code examples assume that the part specific header file is included before compilation. Be aware that not all C compiler vendors include bit definitions in the header files and interrupt handling in C is compiler dependent. Please confirm with the C compiler documentation for more details.

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”.

6. Data retention

Reliability Qualification results show that the projected data retention failure rate is much less than one PPM over 20 years at 85°C or 100 years at 25°C.

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
(0xBF)	Reserved	–	–	–	–	–	–	–	–	
(0xBE)	TWBCSR	TWBCIF	TWBCIE	–	–	–	TWBDT1	TWBDT0	TWBCIP	184
(0xBD)	TWAMR	TWAM[6:0]							–	184
(0xBC)	TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE	181
(0xBB)	TWDR	2-wire Serial Interface Data Register								183
(0xBA)	TWAR	TWA[6:0]							TWGCE	183
(0xB9)	TWSR	TWS[7:3]					–	TWPS1	TWPS0	182
(0xB8)	TWBR	2-wire Serial Interface Bit Rate Register								181
(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	–	–	–	–	–	–	–	–	
(0x99)	Reserved	–	–	–	–	–	–	–	–	
(0x98)	Reserved	–	–	–	–	–	–	–	–	
(0x97)	Reserved	–	–	–	–	–	–	–	–	
(0x96)	Reserved	–	–	–	–	–	–	–	–	
(0x95)	Reserved	–	–	–	–	–	–	–	–	
(0x94)	Reserved	–	–	–	–	–	–	–	–	
(0x93)	Reserved	–	–	–	–	–	–	–	–	
(0x92)	Reserved	–	–	–	–	–	–	–	–	
(0x91)	Reserved	–	–	–	–	–	–	–	–	
(0x90)	Reserved	–	–	–	–	–	–	–	–	
(0x8F)	Reserved	–	–	–	–	–	–	–	–	
(0x8E)	Reserved	–	–	–	–	–	–	–	–	
(0x8D)	Reserved	–	–	–	–	–	–	–	–	
(0x8C)	Reserved	–	–	–	–	–	–	–	–	
(0x8B)	Reserved	–	–	–	–	–	–	–	–	
(0x8A)	Reserved	–	–	–	–	–	–	–	–	
(0x89)	OCR1B	Timer/Counter1 – Output Compare Register B								95
(0x88)	OCR1A	Timer/Counter1 – Output Compare Register A								95
(0x87)	Reserved	–	–	–	–	–	–	–	–	
(0x86)	Reserved	–	–	–	–	–	–	–	–	
(0x85)	TCNT1H	Timer/Counter1 (8 Bit) High Byte								95
(0x84)	TCNT1L	Timer/Counter1 (8 Bit) Low Byte								95
(0x83)	Reserved	–	–	–	–	–	–	–	–	
(0x82)	Reserved	–	–	–	–	–	–	–	–	
(0x81)	TCCR1B	–	–	–	–	–	CS12	CS11	CS10	81
(0x80)	TCCR1A	TCW1	ICEN1	ICNC1	ICES1	ICS1	–	–	WGM10	94
(0x7F)	Reserved	–	–	–	–	–	–	–	–	
(0x7E)	DIDR0	–	–	–	–	–	–	PA1DID	PA0DID	121

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
(0x7D)	Reserved	–	–	–	–	–	–	–	–	
(0x7C)	VADMUX	–	–	–	–	VADMUX[3:0]				119
(0x7B)	Reserved	–	–	–	–	–	–	–	–	
(0x7A)	VADCSR	–	–	–	–	VADEN	VADSC	VADCCIF	VADCCIE	119
(0x79)	VADCH	–	–	–	–	VADC Data Register High byte				120
(0x78)	VADCL	VADC Data Register Low byte								120
(0x77)	Reserved	–	–	–	–	–	–	–	–	
(0x76)	Reserved	–	–	–	–	–	–	–	–	
(0x75)	Reserved	–	–	–	–	–	–	–	–	
(0x74)	Reserved	–	–	–	–	–	–	–	–	
(0x73)	Reserved	–	–	–	–	–	–	–	–	
(0x72)	Reserved	–	–	–	–	–	–	–	–	
(0x71)	Reserved	–	–	–	–	–	–	–	–	
(0x70)	Reserved	–	–	–	–	–	–	–	–	
(0x6F)	TIMSK1	–	–	–	–	ICIE1	OCIE1B	OCIE1A	TOIE1	96
(0x6E)	TIMSK0	–	–	–	–	ICIE0	OCIE0B	OCIE0A	TOIE0	96
(0x6D)	Reserved	–	–	–	–	–	–	–	–	
(0x6C)	PCMSK1	PCINT[15:8]								60
(0x6B)	PCMSK0	–	–	–	–	PCINT[3:0]				61
(0x6A)	Reserved	–	–	–	–	–	–	–	–	
(0x69)	EICRA	ISC31	ISC30	ISC21	ISC20	ISC11	ISC10	ISC01	ISC00	58
(0x68)	PCICR	–	–	–	–	–	–	PCIE1	PCIE0	60
(0x67)	Reserved	–	–	–	–	–	–	–	–	
(0x66)	FOSCCAL	Fast Oscillator Calibration Register								32
(0x65)	Reserved	–	–	–	–	–	–	–	–	
(0x64)	PRR0	–	PRTWI	PRVRM	–	PRSPI	PRTIM1	PRTIM0	PRVADC	40
(0x63)	Reserved	–	–	–	–	–	–	–	–	
(0x62)	Reserved	–	–	–	–	–	–	–	–	
(0x61)	CLKPR	CLKPCE	–	–	–	–	–	CLKPS1	CLKPS0	32
(0x60)	WDTCR	WDIF	WDIE	WDP3	WDCE	WDE	WDP2	WDP1	WDP0	49
0x3F (0x5F)	SREG	I	T	H	S	V	N	Z	C	10
0x3E (0x5E)	SPH	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP8	13
0x3D (0x5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	13
0x3C (0x5C)	Reserved	–	–	–	–	–	–	–	–	
0x3B (0x5B)	Reserved	–	–	–	–	–	–	–	–	
0x3A (0x5A)	Reserved	–	–	–	–	–	–	–	–	
0x39 (0x59)	Reserved	–	–	–	–	–	–	–	–	
0x38 (0x58)	Reserved	–	–	–	–	–	–	–	–	
0x37 (0x57)	SPMCSR	SPMIE	RWWSB	SIGRD	CTPB	RFLB	PGWRT	PGERS	SPMEN	202
0x36 (0x56)	Reserved	–	–	–	–	–	–	–	–	
0x35 (0x55)	MCUCR	–	–	CKOE	PUD	–	–	IVSEL	IVCE	78/32
0x34 (0x54)	MCUSR	–	–	–	OCDRF	WDRF	BODRF	EXTRF	PORF	49
0x33 (0x53)	SMCR	–	–	–	–	SM[2:0]			SE	39
0x32 (0x52)	Reserved	–	–	–	–	–	–	–	–	
0x31 (0x51)	DWDR	debugWIRE Data Register								187
0x30 (0x50)	Reserved	–	–	–	–	–	–	–	–	
0x2F (0x4F)	Reserved	–	–	–	–	–	–	–	–	
0x2E (0x4E)	SPDR	SPI Data Register								107
0x2D (0x4D)	SPSR	SPIF	WCOL	–	–	–	–	–	SPI2X	106
0x2C (0x4C)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	105
0x2B (0x4B)	GPIOR2	General Purpose I/O Register 2								24
0x2A (0x4A)	GPIOR1	General Purpose I/O Register 1								24
0x29 (0x49)	OCR0B	Timer/Counter0 Output Compare Register B								95
0x28 (0x48)	OCR0A	Timer/Counter0 Output Compare Register A								95
0x27 (0x47)	TCNT0H	Timer/Counter0 (8 Bit) High Byte								95
0x26 (0x46)	TCNT0L	Timer/Counter0 (8 Bit) Low Byte								95
0x25 (0x45)	TCCR0B	–	–	–	–	–	CS02	CS01	CS00	81
0x24 (0x44)	TCCR0A	TCW0	ICEN0	ICNC0	ICES0	ICS0	–	–	WGM00	94
0x23 (0x43)	GTCCR	TSM	–	–	–	–	–	–	PSRSYNC	
0x22 (0x42)	EEARH	–	–	–	–	–	–	EEPROM High byte		20
0x21 (0x41)	EEARL	EEPROM Address Register Low Byte								20
0x20 (0x40)	EEDR	EEPROM Data Register								20
0x1F (0x3F)	EECR	–	–	EEP1	EEP0	EERIE	EEMPE	EEPE	EERE	21
0x1E (0x3E)	GPIOR0	General Purpose I/O Register 0								24
0x1D (0x3D)	EIMSK	–	–	–	–	INT3	INT2	INT1	INT0	59
0x1C (0x3C)	EIFR	–	–	–	–	INTF3	INTF2	INTF1	INTF0	59

8. Instruction set summary

Mnemonics	Operands	Description	Operation	Flags	#Clocks
ARITHMETIC AND LOGIC INSTRUCTIONS					
ADD	Rd, Rr	Add two Registers	$Rd \leftarrow Rd + Rr$	Z, C, N, V, H	1
ADC	Rd, Rr	Add with Carry two Registers	$Rd \leftarrow Rd + Rr + C$	Z, C, N, V, H	1
ADIW	Rdl, K	Add Immediate to Word	$Rdh:Rdl \leftarrow Rdh:Rdl + K$	Z, C, N, V, S	2
SUB	Rd, Rr	Subtract two Registers	$Rd \leftarrow Rd - Rr$	Z, C, N, V, H	1
SUBI	Rd, K	Subtract Constant from Register	$Rd \leftarrow Rd - K$	Z, C, N, V, H	1
SBC	Rd, Rr	Subtract with Carry two Registers	$Rd \leftarrow Rd - Rr - C$	Z, C, N, V, H	1
SBCI	Rd, K	Subtract with Carry Constant from Reg.	$Rd \leftarrow Rd - K - C$	Z, C, N, V, H	1
SBIW	Rdl, K	Subtract Immediate from Word	$Rdh:Rdl \leftarrow Rdh:Rdl - K$	Z, C, N, V, S	2
AND	Rd, Rr	Logical AND Registers	$Rd \leftarrow Rd \cdot Rr$	Z, N, V	1
ANDI	Rd, K	Logical AND Register and Constant	$Rd \leftarrow Rd \cdot K$	Z, N, V	1
OR	Rd, Rr	Logical OR Registers	$Rd \leftarrow Rd \vee Rr$	Z, N, V	1
ORI	Rd, K	Logical OR Register and Constant	$Rd \leftarrow Rd \vee K$	Z, N, V	1
EOR	Rd, Rr	Exclusive OR Registers	$Rd \leftarrow Rd \oplus Rr$	Z, N, V	1
COM	Rd	One's Complement	$Rd \leftarrow 0xFF - Rd$	Z, C, N, V	1
NEG	Rd	Two's Complement	$Rd \leftarrow 0x00 - Rd$	Z, C, N, V, H	1
SBR	Rd, K	Set Bit(s) in Register	$Rd \leftarrow Rd \vee K$	Z, N, V	1
CBR	Rd, K	Clear Bit(s) in Register	$Rd \leftarrow Rd \cdot (0xFF - K)$	Z, N, V	1
INC	Rd	Increment	$Rd \leftarrow Rd + 1$	Z, N, V	1
DEC	Rd	Decrement	$Rd \leftarrow Rd - 1$	Z, N, V	1
TST	Rd	Test for Zero or Minus	$Rd \leftarrow Rd \cdot Rd$	Z, N, V	1
CLR	Rd	Clear Register	$Rd \leftarrow Rd \oplus Rd$	Z, N, V	1
SER	Rd	Set Register	$Rd \leftarrow 0xFF$	None	1
MUL	Rd, Rr	Multiply Unsigned	$R1:R0 \leftarrow Rd \times Rr$	Z, C	2
MULS	Rd, Rr	Multiply Signed	$R1:R0 \leftarrow Rd \times Rr$	Z, C	2
MULSU	Rd, Rr	Multiply Signed with Unsigned	$R1:R0 \leftarrow Rd \times Rr$	Z, C	2
FMUL	Rd, Rr	Fractional Multiply Unsigned	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$	Z, C	2
FMULS	Rd, Rr	Fractional Multiply Signed	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$	Z, C	2
FMULSU	Rd, Rr	Fractional Multiply Signed with Unsigned	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$	Z, C	2
BRANCH INSTRUCTIONS					
RJMP	k	Relative Jump	$PC \leftarrow PC + k + 1$	None	2
IJMP		Indirect Jump to (Z)	$PC \leftarrow Z$	None	2
JMP	k	Direct Jump	$PC \leftarrow k$	None	3
RCALL	k	Relative Subroutine Call	$PC \leftarrow PC + k + 1$	None	3
ICALL		Indirect Call to (Z)	$PC \leftarrow Z$	None	3
CALL	k	Direct Subroutine Call	$PC \leftarrow k$	None	4
RET		Subroutine Return	$PC \leftarrow \text{STACK}$	None	4
RETI		Interrupt Return	$PC \leftarrow \text{STACK}$	I	4
CPSE	Rd, Rr	Compare, Skip if Equal	if $(Rd = Rr)$ $PC \leftarrow PC + 2$ or 3	None	1/2/3
CP	Rd, Rr	Compare	$Rd - Rr$	Z, N, V, C, H	1
CPC	Rd, Rr	Compare with Carry	$Rd - Rr - C$	Z, N, V, C, H	1
CPI	Rd, K	Compare Register with Immediate	$Rd - K$	Z, N, V, C, H	1
SBRC	Rr, b	Skip if Bit in Register Cleared	if $(Rr(b) = 0)$ $PC \leftarrow PC + 2$ or 3	None	1/2/3
SBRS	Rr, b	Skip if Bit in Register is Set	if $(Rr(b) = 1)$ $PC \leftarrow PC + 2$ or 3	None	1/2/3
SBIC	P, b	Skip if Bit in I/O Register Cleared	if $(P(b) = 0)$ $PC \leftarrow PC + 2$ or 3	None	1/2/3
SBIS	P, b	Skip if Bit in I/O Register is Set	if $(P(b) = 1)$ $PC \leftarrow PC + 2$ or 3	None	1/2/3
BRBS	s, k	Branch if Status Flag Set	if $(SREG(s) = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRBC	s, k	Branch if Status Flag Cleared	if $(SREG(s) = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2
BREQ	k	Branch if Equal	if $(Z = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRNE	k	Branch if Not Equal	if $(Z = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRCS	k	Branch if Carry Set	if $(C = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRCC	k	Branch if Carry Cleared	if $(C = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRSH	k	Branch if Same or Higher	if $(C = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRLO	k	Branch if Lower	if $(C = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRMI	k	Branch if Minus	if $(N = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRPL	k	Branch if Plus	if $(N = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRGE	k	Branch if Greater or Equal, Signed	if $(N \oplus V = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRLT	k	Branch if Less Than Zero, Signed	if $(N \oplus V = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRHS	k	Branch if Half Carry Flag Set	if $(H = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRHC	k	Branch if Half Carry Flag Cleared	if $(H = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRTS	k	Branch if T Flag Set	if $(T = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRTC	k	Branch if T Flag Cleared	if $(T = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRVS	k	Branch if Overflow Flag is Set	if $(V = 1)$ then $PC \leftarrow PC + k + 1$	None	1/2
BRVC	k	Branch if Overflow Flag is Cleared	if $(V = 0)$ then $PC \leftarrow PC + k + 1$	None	1/2

8. Instruction set summary (Continued)

Mnemonics	Operands	Description	Operation	Flags	#Clocks
BRIE	k	Branch if Interrupt Enabled	if (I = 1) then PC ← PC + k + 1	None	1/2
BRID	k	Branch if Interrupt Disabled	if (I = 0) then PC ← PC + k + 1	None	1/2
BIT AND BIT-TEST INSTRUCTIONS					
SBI	P, b	Set Bit in I/O Register	I/O(P,b) ← 1	None	2
CBI	P, b	Clear Bit in I/O Register	I/O(P,b) ← 0	None	2
LSL	Rd	Logical Shift Left	Rd(n+1) ← Rd(n), Rd(0) ← 0	Z, C, N, V	1
LSR	Rd	Logical Shift Right	Rd(n) ← Rd(n+1), Rd(7) ← 0	Z, C, N, V	1
ROL	Rd	Rotate Left Through Carry	Rd(0) ← C, Rd(n+1) ← Rd(n), C ← Rd(7)	Z, C, N, V	1
ROR	Rd	Rotate Right Through Carry	Rd(7) ← C, Rd(n) ← Rd(n+1), C ← Rd(0)	Z, C, N, V	1
ASR	Rd	Arithmetic Shift Right	Rd(n) ← Rd(n+1), n=0..6	Z, C, N, V	1
SWAP	Rd	Swap Nibbles	Rd(3..0) ← Rd(7..4), Rd(7..4) ← Rd(3..0)	None	1
BSET	s	Flag Set	SREG(s) ← 1	SREG(s)	1
BCLR	s	Flag Clear	SREG(s) ← 0	SREG(s)	1
BST	Rr, b	Bit Store from Register to T	T ← Rr(b)	T	1
BLD	Rd, b	Bit load from T to Register	Rd(b) ← T	None	1
SEC		Set Carry	C ← 1	C	1
CLC		Clear Carry	C ← 0	C	1
SEN		Set Negative Flag	N ← 1	N	1
CLN		Clear Negative Flag	N ← 0	N	1
SEZ		Set Zero Flag	Z ← 1	Z	1
CLZ		Clear Zero Flag	Z ← 0	Z	1
SEI		Global Interrupt Enable	I ← 1	I	1
CLI		Global Interrupt Disable	I ← 0	I	1
SES		Set Signed Test Flag	S ← 1	S	1
CLS		Clear Signed Test Flag	S ← 0	S	1
SEV		Set Twos Complement Overflow	V ← 1	V	1
CLV		Clear Twos Complement Overflow	V ← 0	V	1
SET		Set T in SREG	T ← 1	T	1
CLT		Clear T in SREG	T ← 0	T	1
SEH		Set Half Carry Flag in SREG	H ← 1	H	1
CLH		Clear Half Carry Flag in SREG	H ← 0	H	1
DATA TRANSFER INSTRUCTIONS					
MOV	Rd, Rr	Move Between Registers	Rd ← Rr	None	1
MOVW	Rd, Rr	Copy Register Word	Rd+1:Rd ← Rr+1:Rr	None	1
LDI	Rd, K	Load Immediate	Rd ← K	None	1
LD	Rd, X	Load Indirect	Rd ← (X)	None	2
LD	Rd, X+	Load Indirect and Post-Inc.	Rd ← (X), X ← X + 1	None	2
LD	Rd, -X	Load Indirect and Pre-Dec.	X ← X - 1, Rd ← (X)	None	2
LD	Rd, Y	Load Indirect	Rd ← (Y)	None	2
LD	Rd, Y+	Load Indirect and Post-Inc.	Rd ← (Y), Y ← Y + 1	None	2
LD	Rd, -Y	Load Indirect and Pre-Dec.	Y ← Y - 1, Rd ← (Y)	None	2
LDD	Rd, Y+q	Load Indirect with Displacement	Rd ← (Y + q)	None	2
LD	Rd, Z	Load Indirect	Rd ← (Z)	None	2
LD	Rd, Z+	Load Indirect and Post-Inc.	Rd ← (Z), Z ← Z + 1	None	2
LD	Rd, -Z	Load Indirect and Pre-Dec.	Z ← Z - 1, Rd ← (Z)	None	2
LDD	Rd, Z+q	Load Indirect with Displacement	Rd ← (Z + q)	None	2
LDS	Rd, k	Load Direct from SRAM	Rd ← (k)	None	2
ST	X, Rr	Store Indirect	(X) ← Rr	None	2
ST	X+, Rr	Store Indirect and Post-Inc.	(X) ← Rr, X ← X + 1	None	2
ST	-X, Rr	Store Indirect and Pre-Dec.	X ← X - 1, (X) ← Rr	None	2
ST	Y, Rr	Store Indirect	(Y) ← Rr	None	2
ST	Y+, Rr	Store Indirect and Post-Inc.	(Y) ← Rr, Y ← Y + 1	None	2
ST	-Y, Rr	Store Indirect and Pre-Dec.	Y ← Y - 1, (Y) ← Rr	None	2
STD	Y+q, Rr	Store Indirect with Displacement	(Y + q) ← Rr	None	2
ST	Z, Rr	Store Indirect	(Z) ← Rr	None	2
ST	Z+, Rr	Store Indirect and Post-Inc.	(Z) ← Rr, Z ← Z + 1	None	2
ST	-Z, Rr	Store Indirect and Pre-Dec.	Z ← Z - 1, (Z) ← Rr	None	2
STD	Z+q, Rr	Store Indirect with Displacement	(Z + q) ← Rr	None	2
STS	k, Rr	Store Direct to SRAM	(k) ← Rr	None	2
LPM		Load Program Memory	R0 ← (Z)	None	3
LPM	Rd, Z	Load Program Memory	Rd ← (Z)	None	3
LPM	Rd, Z+	Load Program Memory and Post-Inc.	Rd ← (Z), Z ← Z + 1	None	3
SPM		Store Program Memory	(Z) ← R1:R0	None	-
IN	Rd, P	In Port	Rd ← P	None	1

8. Instruction set summary (Continued)

Mnemonics	Operands	Description	Operation	Flags	#Clocks
OUT	P, Rr	Out Port	$P \leftarrow Rr$	None	1
PUSH	Rr	Push Register on Stack	$STACK \leftarrow Rr$	None	2
POP	Rd	Pop Register from Stack	$Rd \leftarrow STACK$	None	2
MCU CONTROL INSTRUCTIONS					
NOP		No Operation		None	1
SLEEP		Sleep	(see specific descr. for Sleep function)	None	1
WDR		Watchdog Reset	(see specific descr. for WDR/timer)	None	1
BREAK		Break	For On-chip Debug Only	None	N/A

9. Ordering information

9.1 The Atmel ATmega16HVB

Speed (MHz)	Power supply	Ordering code	Package	Operation range
1MHz - 8MHz	4V - 18V	ATMEGA16HVB-8X3	44X1	-40°C to 85°C

Package type	
44X1	44-lead, 4.4mm body width, plastic thin shrink small outline package (TSSOP)

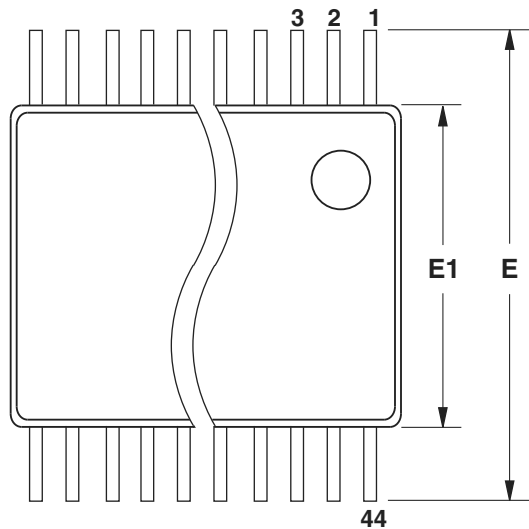
9.2 The Atmel ATmega32HVB

Speed (MHz)	Power supply	Ordering code	Package	Operation range
1MHz - 8MHz	4V - 18V	ATMEGA32HVB-8X3	44X1	-40°C to 85°C

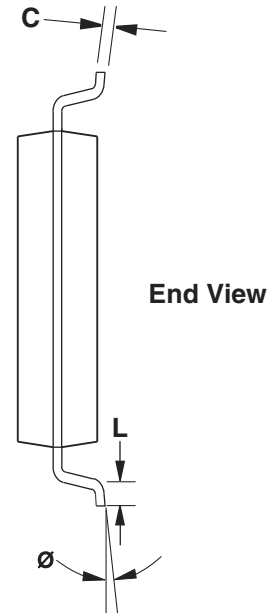
Package type	
44X1	44-lead, 4.4mm body width, plastic thin shrink small outline package (TSSOP)

10. Packaging information

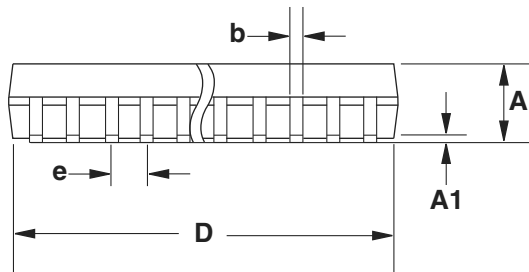
10.1 44X1



Top View



End View



Side View

COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	—	—	1.20	
A1	0.05	—	0.15	
b	0.17	—	0.27	
C	0.09	—	0.20	
D	10.90	11.00	11.10	
E1	4.30	4.40	4.50	
E	6.20	6.40	6.60	
e	0.50 TYP			
L	0.50	0.60	0.70	
Ø	0°	—	8°	

Note: These drawings are for general information only. Refer to JEDEC Drawing MO-153BE.

5/16/07



2325 Orchard Parkway
San Jose, CA 95131

TITLE
44X1, 44-lead, 4.4mm Body Width, Plastic Thin Shrink
Small Outline Package (TSSOP)

DRAWING NO.
44X1

REV.
A

11. Errata

11.1 The Atmel ATmega16HVB

11.1.1 Rev. E

TWI bus can get stuck if TWI STOP condition bit is set in slave mode

If the TWSTO bit in TWCR is set while the TWI starts to receive data in slave mode, it can result in pulling the SCL pin low and then the TWI bus will get stuck. To release the SCL pin and get out of this situation the TWI module needs to be disabled and then re-enabled.

Problem fix/workaround

While in slave mode the TWSTO bit should be written only to recover from an error condition and then cleared before a data transfer starts.

11.1.2 Rev. D

Not sampled.

11.1.3 Rev. C

TWI bus can get stuck if TWI STOP condition bit is set in slave mode

If the TWSTO bit in TWCR is set while the TWI starts to receive data in slave mode, it can result in pulling the SCL pin low and then the TWI bus will get stuck. To release the SCL pin and get out of this situation the TWI module needs to be disabled and then re-enabled.

Problem fix/workaround

While in slave mode the TWSTO bit should be written only to recover from an error condition and then cleared before a data transfer starts.

11.1.4 Rev. B

Stack pointer initial value

The stack pointer in ATmega16HVB is incorrectly initialized to 0x08ff instead of 0x04ff.

Problem fix/workaround

Initialize the stack pointer in software before the stack is used. Most C-compilers does initialize the stack pointer without manual intervention.

Assembly Code Example:

Idi r16,high(RAMEND); Main program start out SPH,r16 ; Set Stack Pointer to top of RAM
Idi r16,low(RAMEND) out SPL,r16 C Code Example (if required): SP = RAMEND;

TWI bus can get stuck if TWI STOP condition bit is set in slave mode

If the TWSTO bit in TWCR is set while the TWI starts to receive data in slave mode, it can result in pulling the SCL pin low and then the TWI bus will get stuck. To release the SCL pin and get out of this situation the TWI module needs to be disabled and then re-enabled.

Problem fix/workaround

While in slave mode the TWSTO bit should be written only to recover from an error condition and then cleared before a data transfer starts.

11.1.5 Rev. A

Not sampled.

11.2 The Atmel ATmega32HVB

11.2.1 Rev. E

TWI bus can get stuck if TWI STOP condition bit is set in slave mode

If the TWSTO bit in TWCR is set while the TWI starts to receive data in slave mode, it can result in pulling the SCL pin low and then the TWI bus will get stuck. To release the SCL pin and get out of this situation the TWI module needs to be disabled and then re-enabled.

Problem fix/workaround

While in slave mode the TWSTO bit should be written only to recover from an error condition and then cleared before a data transfer starts.

11.2.2 Rev. D

Not sampled.

11.2.3 Rev. C

TWI bus can get stuck if TWI STOP condition bit is set in slave mode

If the TWSTO bit in TWCR is set while the TWI starts to receive data in slave mode, it can result in pulling the SCL pin low and then the TWI bus will get stuck. To release the SCL pin and get out of this situation the TWI module needs to be disabled and then re-enabled.

Problem fix/workaround

While in slave mode the TWSTO bit should be written only to recover from an error condition and then cleared before a data transfer starts.

11.2.4 Rev. B

TWI bus can get stuck if TWI STOP condition bit is set in slave mode

If the TWSTO bit in TWCR is set while the TWI starts to receive data in slave mode, it can result in pulling the SCL pin low and then the TWI bus will get stuck. To release the SCL pin and get out of this situation the TWI module needs to be disabled and then re-enabled.

Problem fix/workaround

While in slave mode the TWSTO bit should be written only to recover from an error condition and then cleared before a data transfer starts.

11.2.5 Rev. A

Not sampled.

12. Revision history

Please note that the referring page numbers in this section are referring to this document. The referring revision in this section are referring to the document revision.

12.1 Rev. 8042E-09/2013

1. Updated "Errata" on page 19:
ATmega16HVB: Added errata sections for "Rev. C" , "Rev. D" and "Rev. E" .
ATmega32HVB: Added errata sections for "Rev. B" , "Rev. C" , "Rev. D" and "Rev. E" .

12.2 Rev. 8042D-10/2011

1. Operating voltage has been changed from 4V - 25V to 4V - 18V
2. The methods for determining the actual clock period of the ULP Oscillator i Section 9.2.3 on page 27 have been changed
3. In "Bit 1:0 – CLKPS[1:0]: Clock Prescaler select Bit[1:0]" on page 33 new text has been inserted in and the text "If CKDIV8 is programmed" has been corrected to "If CKDIV8 is programmed"
4. Note 2 in "Bit 0 – OSIEN: Oscillator sampling interface enable" on page 34 has been deleted
5. Figure 11-1 on page 43 has been corrected
6. New Note 2 has been added below Table 11-2 on page 51
7. The last sentence in Section 21.5 on page 123 has been corrected
8. The text in Section 25.3.1 on page 146 below Figure 25-2 has been corrected several places
9. V_{CC} in Figure 28-1 on page 186 has been corrected
10. Bit no 4 in Table 30-3 on page 205 has been corrected
11. Note 1 below Table 30-3 on page 205 has been corrected
12. The text in point 4 and 5 in Section 30.6.1 on page 208 has been corrected
13. The V_{FET} value in Figure 30-3 on page 212 has been corrected
14. The table in Section 32.1 on page 225 have been updated with several new values
15. I_{LOAD} in Table 32-2 on page 226 has been added
16. Note 1 below Table 32-2 on page 226 has been added
17. The maximum value for V_{BOT} in Table 32-3 on page 227 has been added
18. In Table 32-4 on page 227 the maximum value for V_{RSC_L} has been corrected and the maximum value for V_{REG} pin has been added
19. In Table 32-7 on page 229 the typical and maximum values for INL has been corrected
20. In Table 32-8 on page 229 the typical value for frequency prediction error (slow RC oscillator) has been corrected
21. In Table 32-10 on page 230 the text below "Parameter" has been corrected
22. In Table 32-12 on page 231 Note 5 has been added

23. In Table 32-18 on page 236 the maximum value for t_{WLRH_CE} has been corrected
24. The former figure "Active supply current vs. V_{VFET} , WDT, V_{REF} , CBP, OC/OD and CC-ADC enabled" on page 238 has been removed
25. In Table 33-1 on page 243 the text "CC-OD" has been changed to "OC-OD" and below "Typical current consumption" the value "55 μ A" has been changed to "85 μ A"
26. New text is added below the two notes for Table 33-1 on page 243
27. New Figure 33-11 on page 245 "Power-save supply current vs. V_{VFET} , WDT, V_{REF} , CBP, OC/OD, and CC-ADC enabled" is added
28. The plot in Figure 33-13 on page 246 has been updated
29. The plot in Figure 33-14 on page 246 has been updated
30. New Figure 33-15 on page 247 has been added
31. New Figure 33-21 on page 250 has been added
32. Heading in Figure 33-27 on page 253 has been corrected
33. The power supply voltage in the table in Section 9.1 on page 16 has been corrected
34. The power supply voltage in the table in Section 9.2 on page 17 has been corrected
35. The Section 11. on page 19 has been corrected by adding an errata for "all revisions"
36. The text "...clock period of the Slow RC Oscillator..." in point 2 in Section 9.2.3 on page 27 has been corrected to "...clock period of the ULP RC Oscillator..."
37. Note 1 below Table 19-1 on page 112 has been corrected
38. Note 1 below Table 19-2 on page 112 has been corrected
39. Figure 31-1 on page 220 has been updated
40. Figure 31-2 on page 221 has been updated
41. Figure 31-3 on page 222 has been updated
42. Table 31-1 on page 223 has been updated according to the changes in Figure 31-1 on page 220, Figure 31-2 on page 221, and Figure 31-3 on page 222

12.3 Rev. 8042C-06/2011

1. The columns "Minimum" and "Maximum" in Table 24-5 on page 142 are deleted
2. A new row ("Device lot ID and position") in Table 29-3 on page 196 is added
3. A new note ("Note 16") in Table 29-3 on page 196 is added
4. In "Absolute maximum ratings*" on page 225 the following values have been changed: "Voltage on OC and OD with respect to ground", "Voltage on PC5, BATT, PVT, VFET, PV4, PV3, and PV2 with respect to ground", and "Maximum operating voltage on VFET"
5. In Table 32-1 on page 225 the values for "Typical" and "Maximum" in the row "VFET = 16V, WDT, CC-ADC, OC, OD, and battery protection enabled, DUVR mode disabled" are added
6. "Frequency drift" for "Slow RC oscillator" in Table 32-8 on page 229 is deleted
7. A new note ("Note 4") in Table 32-8 on page 229 is added
8. Table 32-10 on page 230 is updated and corrected

9. The text " $C_{EQ} = 4.7nF$, $V_{FET} = 16V$ " is added to "Condition" for $t_{f,OC}$ and $t_{f,OD}$ in Table 32-2 on page 226
10. New Figure 33-1 on page 238 is added
11. Corrected formula in Table 32-15 on page 232
12. Corrected and added some short-cuts in addition to general update and some minor corrections in text

12.4 Rev. 8042B-06/2010

1. Removed direction arrow in Figure 17-1 on page 82.
2. Updated "Configuring PA1 and PA0 for V-ADC operation" on page 117.
3. Updated "Operating circuit" on page 220, with correct naming convention for thermistors RT32 and RT33.

12.5 Rev. 8042A-08/2009

1. Initial revision



Enabling Unlimited Possibilities®

Atmel Corporation

1600 Technology Drive
San Jose, CA 95110
USA

Tel: (+1) (408) 441-0311

Fax: (+1) (408) 487-2600

www.atmel.com

Atmel Asia Limited

Unit 01-5 & 16, 19F
BEA Tower, Millennium City 5
418 Kwun Tong Roa

Kwun Tong, Kowloon

HONG KONG

Tel: (+852) 2245-6100

Fax: (+852) 2722-1369

Atmel Munich GmbH

Business Campus
Parkring 4
D-85748 Garching b. Munich
GERMANY

Tel: (+49) 89-31970-0

Fax: (+49) 89-3194621

Atmel Japan G.K.

16F Shin-Osaki Kangyo Bldg
1-6-4 Osaki, Shinagawa-ku
Tokyo 141-0032
JAPAN

Tel: (+81) (3) 6417-0300

Fax: (+81) (3) 6417-0370

© 2013 Atmel Corporation. All rights reserved. / Rev.: 8042ES-AVR-09/2013

Atmel®, Atmel logo and combinations thereof, Enabling Unlimited Possibilities®, AVR® and others are registered trademarks or trademarks of Atmel Corporation or its subsidiaries. Other terms and product names may be trademarks of others.

Disclaimer: The information in this document is provided in connection with Atmel products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Atmel products. EXCEPT AS SET FORTH IN THE ATMEL TERMS AND CONDITIONS OF SALES LOCATED ON THE ATMEL WEBSITE, ATMEL ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL ATMEL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS AND PROFITS, BUSINESS INTERRUPTION, OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF ATMEL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Atmel makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and products descriptions at any time without notice. Atmel does not make any commitment to update the information contained herein. Unless specifically provided otherwise, Atmel products are not suitable for, and shall not be used in, automotive applications. Atmel products are not intended, authorized, or warranted for use as components in applications intended to support or sustain life.

8042ES-AVR-09/2013