## Features

- High Performance, Low Power AVR® 8-Bit Microcontroller
- Advanced RISC Architecture
  - 135 Powerful Instructions – Most Single Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 16 MIPS Throughput at 16 MHz
  - On-Chip 2-cycle Multiplier
- Non-volatile Program and Data Memories
  - 16/32K Bytes of In-System Self-Programmable Flash
    - Endurance: 100,000 Write/Erase Cycles
  - Optional Boot Code Section with Independent Lock Bits
    - In-System Programming by On-chip Boot Program hardware activated after reset
    - True Read-While-Write Operation
    - All supplied parts are preprogramed with a default USB bootloader
  - 1.25/2.5K Bytes Internal SRAM
  - 512Bytes/1K Bytes Internal EEPROM
    - Endurance: 100,000 Write/Erase Cycles
  - Programming Lock for Software Security
- JTAG (IEEE std. 1149.1 compliant) Interface
  - Boundary-scan Capabilities According to the JTAG Standard
  - Extensive On-chip Debug Support
  - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- USB 2.0 Full-speed/Low Speed Device Module with Interrupt on Transfer Completion
  - Complies fully with Universal Serial Bus Specification Rev 2.0
  - Supports data transfer rates up to 12 Mbit/s and 1.5 Mbit/s
  - Endpoint 0 for Control Transfers: up to 64-bytes
  - 6 Programmable Endpoints with IN or Out Directions and with Bulk, Interrupt or Isochronous Transfers
  - Configurable Endpoints size up to 256 bytes in double bank mode
  - Fully independent 832 bytes USB DPRAM for endpoint memory allocation
  - Suspend/Resume Interrupts
  - CPU Reset possible on USB Bus Reset detection
  - 48 MHz from PLL for Full-speed Bus Operation
  - USB Bus Connection/Disconnection on Microcontroller Request
- Peripheral Features
  - On-chip PLL for USB and High Speed Timer: 32 up to 96 MHz operation
  - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
  - Two 16-bit Timer/Counter with Separate Prescaler, Compare- and Capture Mode
  - One 10-bit High-Speed Timer/Counter with PLL (64 MHz) and Compare Mode
  - Four 8-bit PWM Channels
  - Four PWM Channels with Programmable Resolution from 2 to 16 Bits
  - Six PWM Channels for High Speed Operation, with Programmable Resolution from 2 to 11 Bits
  - Output Compare Modulator
  - 12-channels, 10-bit ADC (features Differential Channels with Programmable Gain)
  - Programmable Serial USART with Hardware Flow Control
  - Master/Slave SPI Serial Interface

# 8-bit AVR® Microcontroller with 16/32K Bytes of ISP Flash and USB Controller

## ATmega16U4
## ATmega32U4

## Preliminary

## Summary

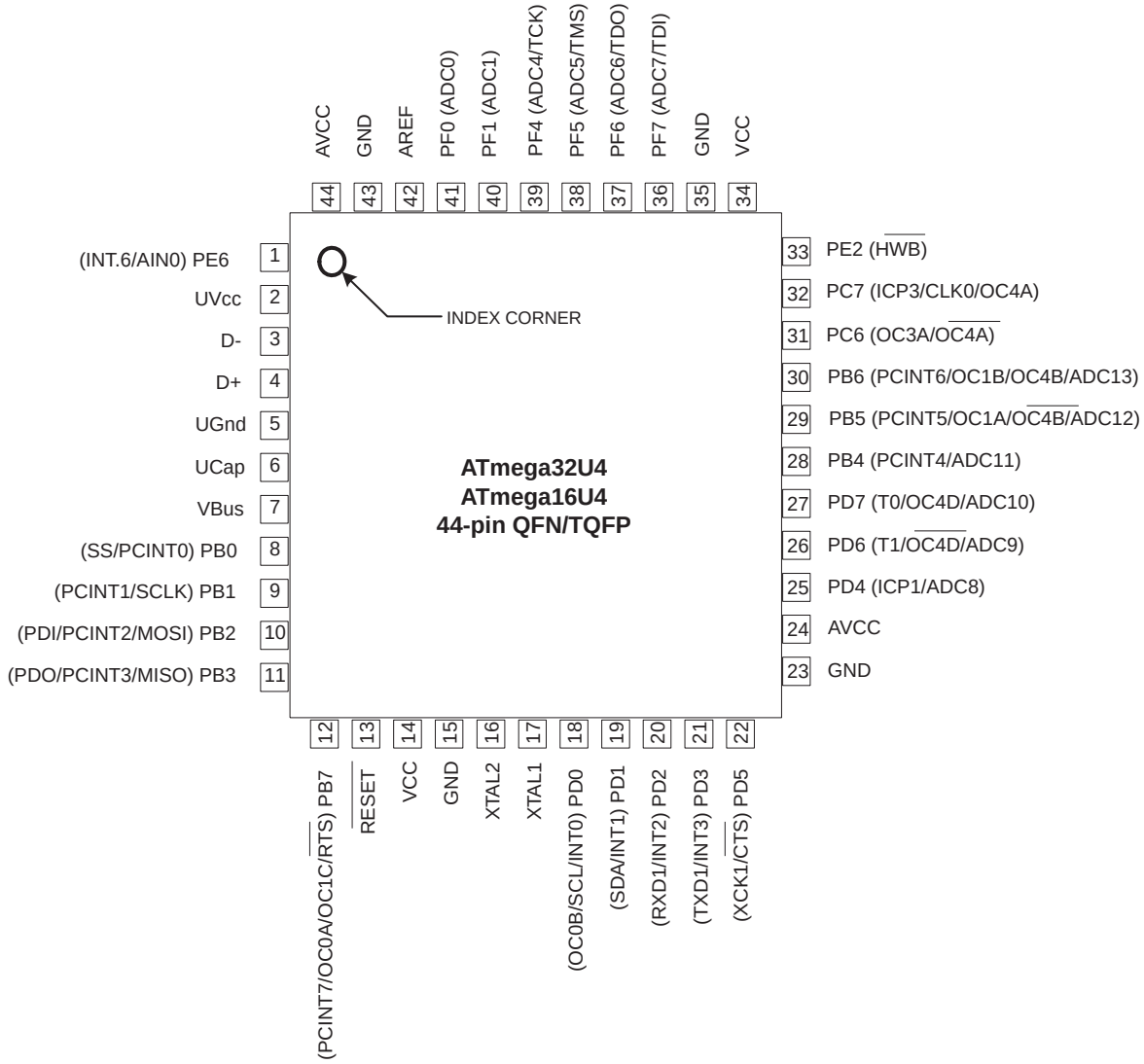
7766CS–AVR–11/08

- Byte Oriented 2-wire Serial Interface
- Programmable Watchdog Timer with Separate On-chip Oscillator
- On-chip Analog Comparator
- Interrupt and Wake-up on Pin Change (8xPCINT + 5xINT sources)
- On-chip Temperature Sensor (see A/D Converter section)

- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal 8 MHz Calibrated Oscillator
  - Internal clock prescaler & On-the-fly Clock Switching (Int RC / Ext Osc)
  - External and Internal Interrupt Sources
  - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
  - All I/O combine CMOS outputs and LVTTL inputs
  - 26 Programmable I/O Lines
  - 44-lead TQFP Package, 10x10mm
  - 44-lead QFN Package, 7x7mm
- Operating Voltages
  - 2.7 - 5.5V
- Operating temperature
  - Industrial (-40°C to +85°C)
- Maximum Frequency
  - 8 MHz at 2.7V - Industrial range
  - 16 MHz at 4.5V - Industrial range

# 1. Pin Configurations

**Figure 1-1.** Pinout ATmega16U4/ATmega32U4




## 1.1 Disclaimer

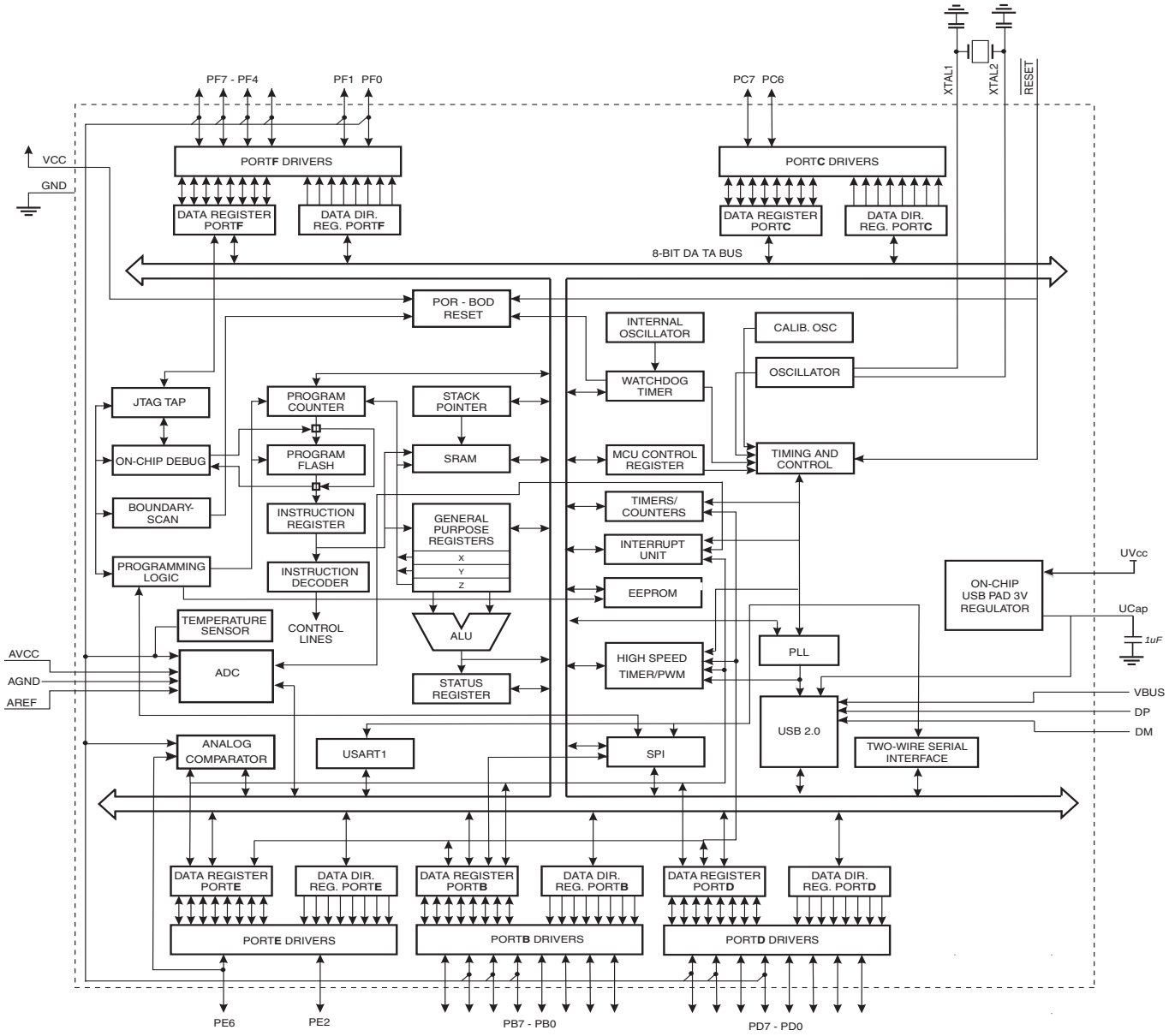
Typical values contained in this datasheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Min and Max values will be available after the device is characterized.

# 2. Overview

The ATmega16U4/ATmega32U4 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega16U4/ATmega32U4 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

## 2.1 Block Diagram

**Figure 2-1.** Block Diagram




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 ATmega16U4/ATmega32U4 provides the following features: 16/32K bytes of In-System Programmable Flash with Read-While-Write capabilities, 512Bytes/1K bytes EEPROM, 1.25/2.5K bytes SRAM, 26 general purpose I/O lines (CMOS outputs and LVTTL inputs), 32 general purpose working registers, four flexible Timer/Counters with compare modes and PWM, one more high-speed Timer/Counter with compare modes and PLL adjustable source, one USART (including CTS/RTS flow control signals), a byte oriented 2-wire Serial Interface, a 12-

channels 10-bit ADC with optional differential input stage with programmable gain, an on-chip calibrated temperature sensor, a programmable Watchdog Timer with Internal Oscillator, an SPI serial port, IEEE std. 1149.1 compliant JTAG test interface, also used for accessing the On-chip Debug system and programming and six software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next interrupt or Hardware Reset. The ADC Noise Reduction mode stops the CPU and all I/O modules except ADC, to minimize switching noise during ADC conversions. In Standby mode, the Crystal/Resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low power consumption.

The device is manufactured using ATMEL's high-density nonvolatile memory technology. The On-chip ISP Flash allows the program memory to be reprogrammed in-system through an SPI serial interface, by a conventional nonvolatile 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 on a monolithic chip, the ATMEL ATmega16U4/ATmega32U4 is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications.

The ATmega16U4/ATmega32U4 AVR is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, in-circuit emulators, and evaluation kits.

## 2.2 Pin Descriptions

### 2.2.1 VCC

Digital supply voltage.

### 2.2.2 GND

Ground.

### 2.2.3 Port B (PB7..PB0)

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. 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 has better driving capabilities than the other ports.

Port B also serves the functions of various special features of the ATmega16U4/ATmega32U4 as listed on page 70.

### 2.2.4 Port C (PC7,PC6)

Port C is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Only bits 6 and 7 are present on the product pinout.

Port C also serves the functions of special features of the ATmega16U4/ATmega32U4 as listed on page 73.

### 2.2.5 Port D (PD7..PD0)

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port D also serves the functions of various special features of the ATmega16U4/ATmega32U4 as listed on page 75.

### 2.2.6 Port E (PE6,PE2)

Port E is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port E output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port E pins that are externally pulled low will source current if the pull-up resistors are activated. The Port E pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Only bits 2 and 6 are present on the product pinout.

Port E also serves the functions of various special features of the ATmega16U4/ATmega32U4 as listed on page 78.

### 2.2.7 Port F (PF7..PF4, PF1,PF0)

Port F serves as analog inputs to the A/D Converter.

Port F also serves as an 8-bit bi-directional I/O port, if the A/D Converter channels are not used. Port pins can provide internal pull-up resistors (selected for each bit). The Port F output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port F pins that are externally pulled low will source current if the pull-up resistors are activated. The Port F pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Bits 2 and 3 are not present on the product pinout.

Port F also serves the functions of the JTAG interface. If the JTAG interface is enabled, the pull-up resistors on pins PF7(TDI), PF5(TMS), and PF4(TCK) will be activated even if a reset occurs.

### 2.2.8 D-

USB Full speed / Low Speed Negative Data Upstream Port. Should be connected to the USB D- connector pin with a serial 22 Ohms resistor.

### 2.2.9 D+

USB Full speed / Low Speed Positive Data Upstream Port. Should be connected to the USB D+ connector pin with a serial 22 Ohms resistor.

### 2.2.10 UGND

USB Pads Ground.

#### 2.2.11 UVCC

USB Pads Internal Regulator Input supply voltage.

#### 2.2.12 UCAP

USB Pads Internal Regulator Output supply voltage. Should be connected to an external capacitor (1µF).

#### 2.2.13 VBUS

USB VBUS monitor input.

#### 2.2.14 $\overline{\text{RESET}}$

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 8-1 on page 48. Shorter pulses are not guaranteed to generate a reset.

#### 2.2.15 XTAL1

Input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

#### 2.2.16 XTAL2

Output from the inverting Oscillator amplifier.

#### 2.2.17 AVCC

AVCC is the supply voltage pin (input) for all the A/D Converter channels. If the ADC is not used, it should be externally connected to $V_{CC}$. If the ADC is used, it should be connected to $V_{CC}$ through a low-pass filter.

#### 2.2.18 AREF

This is the analog reference pin (input) for the A/D Converter.

## 3. About Code Examples

This documentation contains simple code examples that briefly show how to use various parts of the device. 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.

These code examples assume that the part specific header file is included before compilation. 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 "SBRS", "SBRC", "SBR", and "CBR".

# 4. 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) | Reserved | - | - | - | - | - | - | - | - | |
| (0xF5) | Reserved | - | - | - | - | - | - | - | - | |
| (0xF4) | UEINT | - | EPINT6:0 | | | | | | | |
| (0xF3) | UEBCHX | - | - | - | - | - | BYCT10:8 | | | |
| (0xF2) | UEBCLX | BYCT7:0 | | | | | | | | |
| (0xF1) | UEDATX | DAT7:0 | | | | | | | | |
| (0xF0) | UEIENX | FLERRE | NAKINE | - | NAKOUTE | RXSTPE | RXOUTE | STALLEDE | TXINE | |
| (0xEF) | UESTA1X | - | - | - | - | - | CTRLDIR | CURRBK1:0 | | |
| (0xEE) | UESTA0X | CFGOK | OVERFI | UNDERFI | - | DTSEQ1:0 | | NBUSYBK1:0 | | |
| (0xED) | UECFG1X | | EPSIZE2:0 | | | EPBK1:0 | | ALLOC | - | |
| (0xEC) | UECFG0X | EPTYPE1:0 | | - | - | - | - | - | EPDIR | |
| (0xEB) | UECONX | - | - | STALLRQ | STALLRQC | RSTDT | - | - | EPEN | |
| (0xEA) | UERST | - | EPRST6:0 | | | | | | | |
| (0xE9) | UENUM | - | - | - | - | - | EPNUM2:0 | | | |
| (0xE8) | UEINTX | FIFOCON | NAKINI | RWAL | NAKOUTI | RXSTPI | RXOUTI | STALLEDI | TXINI | |
| (0xE7) | Reserved | | | - | - | - | - | | | |
| (0xE6) | UDMFN | - | - | - | FNCERR | - | - | - | - | |
| (0xE5) | UDFNUMH | - | - | - | - | - | FNUM10:8 | | | |
| (0xE4) | UDFNUML | FNUM7:0 | | | | | | | | |
| (0xE3) | UDADDR | ADDEN | UADD6:0 | | | | | | | |
| (0xE2) | UDIEN | - | UPRSME | EORSME | WAKEUPE | EORSTE | SOFE | MSOFE | SUSPE | |
| (0xE1) | UDINT | - | UPRSMI | EORSMI | WAKEUPI | EORSTI | SOFI | MSOFI | SUSPI | |
| (0xE0) | UDCON | - | - | - | - | RSTCPU | LSM | RMWKUP | DETACH | |
| (0xDF) | Reserved | | | | | | | | | |
| (0xDE) | Reserved | | | | | | | | | |
| (0xDD) | Reserved | | | | | | | | | |
| (0xDC) | Reserved | | | | | | | | | |
| (0xDB) | Reserved | | | | | | | | | |
| (0xDA) | USBINT | - | - | - | - | - | - | - | VBUSTI | |
| (0xD9) | USBSTA | - | - | - | - | - | - | ID | VBUS | |
| (0xD8) | USBCON | USBE | - | FRZCLK | OTGPADE | - | - | - | VBUSTE | |
| (0xD7) | UHWCON | - | - | - | - | - | - | - | UVREGE | |
| (0xD6) | Reserved | | | | | | | | | |
| (0xD5) | Reserved | | | | | | | | | |
| (0xD4) | DT4 | DT4H3 | DT4H2 | DT4H1 | DT4H0 | DT4L3 | DT4L2 | DT4L1 | DT4L0 | |
| (0xD3) | Reserved | | | | | | | | | |
| (0xD2) | OCR4D | Timer/Counter4 - Output Compare Register D | | | | | | | | |
| (0xD1) | OCR4C | Timer/Counter4 - Output Compare Register C | | | | | | | | |
| (0xD0) | OCR4B | Timer/Counter4 - Output Compare Register B | | | | | | | | |
| (0xCF) | OCR4A | Timer/Counter4 - Output Compare Register A | | | | | | | | |
| (0xCE) | UDR1 | USART1 I/O Data Register | | | | | | | | |
| (0xCD) | UBRR1H | - | - | - | - | USART1 Baud Rate Register High Byte | | | | |
| (0xCC) | UBRR1L | USART1 Baud Rate Register Low Byte | | | | | | | | |
| (0xCB) | Reserved | - | - | - | - | - | - | - | - | |
| (0xCA) | UCSR1C | UMSEL11 | UMSEL10 | UPM11 | UPM10 | USBS1 | UCSZ11 | UCSZ10 | UCPOL1 | |
| (0xC9) | UCSR1B | RXCIE1 | TXCIE1 | UDRIE1 | RXEN1 | TXEN1 | UCSZ12 | RXB81 | TXB81 | |
| (0xC8) | UCSR1A | RXC1 | TXC1 | UDRE1 | FE1 | DOR1 | PE1 | U2X1 | MPCM1 | |
| (0xC7) | CLKSTA | - | - | - | - | - | - | RCON | EXTON | |
| (0xC6) | CLKSEL1 | RCCKSEL3 | RCCKSEL2 | RCCKSEL1 | RCCKSEL0 | EXCKSEL3 | EXCKSEL2 | EXCKSEL1 | EXCKSEL0 | |
| (0xC5) | CLKSEL0 | RCSUT1 | RCSUT0 | EXSUT1 | EXSUT0 | RCE | EXTE | - | CLKS | |
| (0xC4) | TCCR4E | TLOCK4 | ENHC4 | OC4OE5 | OC4OE4 | OC4OE3 | OC4OE2 | OC4OE1 | OC4OE0 | |
| (0xC3) | TCCR4D | FPIE4 | FPEN4 | FPNC4 | FPES4 | FPAC4 | FPF4 | WGM41 | WGM40 | |
| (0xC2) | TCCR4C | COM4A1S | COM4A0S | COM4B1S | COM4B0S | COM4D1S | COM4D0S | FOC4D | PWM4D | |
| (0xC1) | TCCR4B | PWM4X | PSR4 | DTPS41 | DTPS40 | CS43 | CS42 | CS41 | CS40 | |
| (0xC0) | TCCR4A | COM4A1 | COM4A0 | COM4B1 | COM4B0 | FOC4A | FOC4B | PWM4A | PWM4B | |
| (0xBF) | TC4H | - | - | - | - | - | Timer/Counter4 High Byte | | | |

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Page |
|---|---|---|---|---|---|---|---|---|---|---|
| (0xBE) | TCNT4 | Timer/Counter4 - Counter Register Low Byte | | | | | | | | |
| (0xBD) | TWAMR | TWAM6 | TWAM5 | TWAM4 | TWAM3 | TWAM2 | TWAM1 | TWAM0 | - | |
| (0xBC) | TWCR | TWINT | TWEA | TWSTA | TWSTO | TWWC | TWEN | - | TWIE | |
| (0xBB) | TWDR | 2-wire Serial Interface Data Register | | | | | | | | |
| (0xBA) | TWAR | TWA6 | TWA5 | TWA4 | TWA3 | TWA2 | TWA1 | TWA0 | TWGCE | |
| (0xB9) | TWSR | TWS7 | TWS6 | TWS5 | TWS4 | TWS3 | - | TWPS1 | TWPS0 | |
| (0xB8) | TWBR | 2-wire Serial Interface Bit Rate Register | | | | | | | | |
| (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) | OCR3CH | Timer/Counter3 - Output Compare Register C High Byte | | | | | | | | |
| (0x9C) | OCR3CL | Timer/Counter3 - Output Compare Register C Low Byte | | | | | | | | |
| (0x9B) | OCR3BH | Timer/Counter3 - Output Compare Register B High Byte | | | | | | | | |
| (0x9A) | OCR3BL | Timer/Counter3 - Output Compare Register B Low Byte | | | | | | | | |
| (0x99) | OCR3AH | Timer/Counter3 - Output Compare Register A High Byte | | | | | | | | |
| (0x98) | OCR3AL | Timer/Counter3 - Output Compare Register A Low Byte | | | | | | | | |
| (0x97) | ICR3H | Timer/Counter3 - Input Capture Register High Byte | | | | | | | | |
| (0x96) | ICR3L | Timer/Counter3 - Input Capture Register Low Byte | | | | | | | | |
| (0x95) | TCNT3H | Timer/Counter3 - Counter Register High Byte | | | | | | | | |
| (0x94) | TCNT3L | Timer/Counter3 - Counter Register Low Byte | | | | | | | | |
| (0x93) | Reserved | - | - | - | - | - | - | - | - | |
| (0x92) | TCCR3C | FOC3A | - | - | - | - | - | - | - | |
| (0x91) | TCCR3B | ICNC3 | ICES3 | - | WGM33 | WGM32 | CS32 | CS31 | CS30 | |
| (0x90) | TCCR3A | COM3A1 | COM3A0 | COM3B1 | COM3B0 | COM3C1 | COM3C0 | WGM31 | WGM30 | |
| (0x8F) | Reserved | - | - | - | - | - | - | - | - | |
| (0x8E) | Reserved | - | - | - | - | - | - | - | - | |
| (0x8D) | OCR1CH | Timer/Counter1 - Output Compare Register C High Byte | | | | | | | | |
| (0x8C) | OCR1CL | Timer/Counter1 - Output Compare Register C Low Byte | | | | | | | | |
| (0x8B) | OCR1BH | Timer/Counter1 - Output Compare Register B High Byte | | | | | | | | |
| (0x8A) | OCR1BL | Timer/Counter1 - Output Compare Register B Low Byte | | | | | | | | |
| (0x89) | OCR1AH | Timer/Counter1 - Output Compare Register A High Byte | | | | | | | | |
| (0x88) | OCR1AL | Timer/Counter1 - Output Compare Register A Low Byte | | | | | | | | |
| (0x87) | ICR1H | Timer/Counter1 - Input Capture Register High Byte | | | | | | | | |
| (0x86) | ICR1L | Timer/Counter1 - Input Capture Register Low Byte | | | | | | | | |
| (0x85) | TCNT1H | Timer/Counter1 - Counter Register High Byte | | | | | | | | |
| (0x84) | TCNT1L | Timer/Counter1 - Counter Register Low Byte | | | | | | | | |
| (0x83) | Reserved | - | - | - | - | - | - | - | - | |
| (0x82) | TCCR1C | FOC1A | FOC1B | FOC1C | - | - | - | - | - | |
| (0x81) | TCCR1B | ICNC1 | ICES1 | - | WGM13 | WGM12 | CS12 | CS11 | CS10 | |
| (0x80) | TCCR1A | COM1A1 | COM1A0 | COM1B1 | COM1B0 | COM1C1 | COM1C0 | WGM11 | WGM10 | |
| (0x7F) | DIDR1 | - | - | - | - | - | - | - | AIN0D | |
| (0x7E) | DIDR0 | ADC7D | ADC6D | ADC5D | ADC4D | - | - | ADC1D | ADC0D | |
| (0x7D) | DIDR2 | - | - | ADC13D | ADC12D | ADC11D | ADC10D | ADC9D | ADC8D | |

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Page |
|---|---|---|---|---|---|---|---|---|---|---|
| (0x7C) | ADMUX | REFS1 | REFS0 | ADLAR | MUX4 | MUX3 | MUX2 | MUX1 | MUX0 | |
| (0x7B) | ADCSRB | ADHSM | ACME | MUX5 | - | ADTS3 | ADTS2 | ADTS1 | ADTS0 | |
| (0x7A) | ADCSRA | ADEN | ADSC | ADATE | ADIF | ADIE | ADPS2 | ADPS1 | ADPS0 | |
| (0x79) | ADCH | ADC Data Register High byte | | | | | | | | |
| (0x78) | ADCL | ADC Data Register Low byte | | | | | | | | |
| (0x77) | Reserved | - | - | - | - | - | - | - | - | |
| (0x76) | Reserved | - | - | - | - | - | - | - | - | |
| (0x75) | Reserved | - | - | - | - | - | - | - | - | |
| (0x74) | Reserved | - | - | - | - | - | - | - | - | |
| (0x73) | Reserved | - | - | - | - | - | - | - | - | |
| (0x72) | TIMSK4 | OCIE4D | OCIE4A | OCIE4B | - | - | TOIE4 | - | - | |
| (0x71) | TIMSK3 | - | - | ICIE3 | - | OCIE3C | OCIE3B | OCIE3A | TOIE3 | |
| (0x70) | Reserved | - | - | - | - | - | - | - | - | |
| (0x6F) | TIMSK1 | - | - | ICIE1 | - | OCIE1C | OCIE1B | OCIE1A | TOIE1 | |
| (0x6E) | TIMSK0 | - | - | - | - | - | OCIE0B | OCIE0A | TOIE0 | |
| (0x6D) | Reserved | - | - | - | - | - | - | - | - | |
| (0x6C) | Reserved | - | - | - | - | - | - | - | - | |
| (0x6B) | PCMSK0 | PCINT7 | PCINT6 | PCINT5 | PCINT4 | PCINT3 | PCINT2 | PCINT1 | PCINT0 | |
| (0x6A) | EICRB | - | - | ISC61 | ISC60 | - | - | - | - | |
| (0x69) | EICRA | ISC31 | ISC30 | ISC21 | ISC20 | ISC11 | ISC10 | ISC01 | ISC00 | |
| (0x68) | PCICR | - | - | - | - | - | - | - | PCIE0 | |
| (0x67) | RCCTRL | - | - | - | - | - | - | - | RCFREQ | |
| (0x66) | OSCCAL | RC Oscillator Calibration Register | | | | | | | | |
| (0x65) | PRR1 | PRUSB | - | - | PRTIM4 | PRTIM3 | - | - | PRUSART1 | |
| (0x64) | PRR0 | PRTWI | - | PRTIM0 | - | PRTIM1 | PRSPI | - | PRADC | |
| (0x63) | Reserved | - | - | - | - | - | - | - | - | |
| (0x62) | Reserved | - | - | - | - | - | - | - | - | |
| (0x61) | CLKPR | CLKPCE | - | - | - | CLKPS3 | CLKPS2 | CLKPS1 | CLKPS0 | |
| (0x60) | WDTCSR | WDIF | WDIE | WDP3 | WDCE | WDE | WDP2 | WDP1 | WDP0 | |
| 0x3F (0x5F) | SREG | I | T | H | S | V | N | Z | C | |
| 0x3E (0x5E) | SPH | SP15 | SP14 | SP13 | SP12 | SP11 | SP10 | SP9 | SP8 | |
| 0x3D (0x5D) | SPL | SP7 | SP6 | SP5 | SP4 | SP3 | SP2 | SP1 | SP0 | |
| 0x3C (0x5C) | Reserved | - | - | - | - | - | - | - | - | |
| 0x3B (0x5B) | RAMPZ | - | - | - | - | - | - | RAMPZ1 | RAMPZ0 | |
| 0x3A (0x5A) | Reserved | - | - | - | - | - | - | - | - | |
| 0x39 (0x59) | Reserved | - | - | - | - | - | - | - | - | |
| 0x38 (0x58) | Reserved | - | - | - | - | - | - | - | - | |
| 0x37 (0x57) | SPMCSR | SPMIE | RWWSB | SIGRD | RWWSRE | BLBSET | PGWRT | PGERS | SPMEN | |
| 0x36 (0x56) | Reserved | - | - | - | - | - | - | - | - | |
| 0x35 (0x55) | MCUCR | JTD | - | - | PUD | - | - | IVSEL | IVCE | |
| 0x34 (0x54) | MCUSR | - | - | USBRF | JTRF | WDRF | BORF | EXTRF | PORF | |
| 0x33 (0x53) | SMCR | - | - | - | - | SM2 | SM1 | SM0 | SE | |
| 0x32 (0x52) | PLLFRQ | PINMUX | PLLUSB | PLLTM1 | PLLTM0 | PDIV3 | PDIV2 | PDIV1 | PDIV0 | |
| 0x31 (0x51) | OCDR/ MONDR | OCDR7 | OCDR6 | OCDR5 | OCDR4 | OCDR3 | OCDR2 | OCDR1 | OCDR0 | |
| | | Monitor Data Register | | | | | | | | |
| 0x30 (0x50) | ACSR | ACD | ACBG | ACO | ACI | ACIE | ACIC | ACIS1 | ACIS0 | |
| 0x2F (0x4F) | Reserved | - | - | - | - | - | - | - | - | |
| 0x2E (0x4E) | SPDR | SPI Data Register | | | | | | | | |
| 0x2D (0x4D) | SPSR | SPIF | WCOL | - | - | - | - | - | SPI2X | |
| 0x2C (0x4C) | SPCR | SPIE | SPE | DORD | MSTR | CPOL | CPHA | SPR1 | SPR0 | |
| 0x2B (0x4B) | GPIOR2 | General Purpose I/O Register 2 | | | | | | | | |
| 0x2A (0x4A) | GPIOR1 | General Purpose I/O Register 1 | | | | | | | | |
| 0x29 (0x49) | PLLCSR | - | - | - | PINDIV | - | - | PLLE | PLOCK | |
| 0x28 (0x48) | OCR0B | Timer/Counter0 Output Compare Register B | | | | | | | | |
| 0x27 (0x47) | OCR0A | Timer/Counter0 Output Compare Register A | | | | | | | | |
| 0x26 (0x46) | TCNT0 | Timer/Counter0 (8 Bit) | | | | | | | | |
| 0x25 (0x45) | TCCR0B | FOC0A | FOC0B | - | - | WGM02 | CS02 | CS01 | CS00 | |
| 0x24 (0x44) | TCCR0A | COM0A1 | COM0A0 | COM0B1 | COM0B0 | - | - | WGM01 | WGM00 | |
| 0x23 (0x43) | GTCCR | TSM | - | - | - | - | - | PSRASY | PSRSYNC | |
| 0x22 (0x42) | EEARH | - | - | - | - | EEPROM Address Register High Byte | | | | |
| 0x21 (0x41) | EEARL | EEPROM Address Register Low Byte | | | | | | | | |
| 0x20 (0x40) | EEDR | EEPROM Data Register | | | | | | | | |
| 0x1F (0x3F) | EECR | - | - | EEPM1 | EEPM0 | EERIE | EEMPE | EEPE | EERE | |
| 0x1E (0x3E) | GPIOR0 | General Purpose I/O Register 0 | | | | | | | | |
| 0x1D (0x3D) | EIMSK | - | INT6 | - | - | INT3 | INT2 | INT1 | INT0 | |
| 0x1C (0x3C) | EIFR | - | INTF6 | - | - | INTF3 | INTF2 | INTF1 | INTF0 | |

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Page |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x1B (0x3B) | PCIFR | - | - | - | - | - | - | - | PCIF0 | |
| 0x1A (0x3A) | Reserved | - | - | - | - | - | - | - | - | |
| 0x19 (0x39) | TIFR4 | OCF4D | OCF4A | OCF4B | - | - | TOV4 | - | - | |
| 0x18 (0x38) | TIFR3 | - | - | ICF3 | - | OCF3C | OCF3B | OCF3A | TOV3 | |
| 0x17 (0x37) | Reserved | - | - | - | - | - | - | - | - | |
| 0x16 (0x36) | TIFR1 | - | - | ICF1 | - | OCF1C | OCF1B | OCF1A | TOV1 | |
| 0x15 (0x35) | TIFR0 | - | - | - | - | - | OCF0B | OCF0A | TOV0 | |
| 0x14 (0x34) | Reserved | - | - | - | - | - | - | - | - | |
| 0x13 (0x33) | Reserved | - | - | - | - | - | - | - | - | |
| 0x12 (0x32) | Reserved | - | - | - | - | - | - | - | - | |
| 0x11 (0x31) | PORTF | PORTF7 | PORTF6 | PORTF5 | PORTF4 | - | - | PORTF1 | PORTF0 | |
| 0x10 (0x30) | DDRF | DDF7 | DDF6 | DDF5 | DDF4 | - | - | DDF1 | DDF0 | |
| 0x0F (0x2F) | PINF | PINF7 | PINF6 | PINF5 | PINF4 | - | - | PINF1 | PINF0 | |
| 0x0E (0x2E) | PORTE | - | PORTE6 | - | - | - | PORTE2 | - | - | |
| 0x0D (0x2D) | DDRE | - | DDE6 | - | - | - | DDE2 | - | - | |
| 0x0C (0x2C) | PINE | - | PINE6 | - | - | - | PINE2 | - | - | |
| 0x0B (0x2B) | PORTD | PORTD7 | PORTD6 | PORTD5 | PORTD4 | PORTD3 | PORTD2 | PORTD1 | PORTD0 | |
| 0x0A (0x2A) | DDRD | DDD7 | DDD6 | DDD5 | DDD4 | DDD3 | DDD2 | DDD1 | DDD0 | |
| 0x09 (0x29) | PIND | PIND7 | PIND6 | PIND5 | PIND4 | PIND3 | PIND2 | PIND1 | PIND0 | |
| 0x08 (0x28) | PORTC | PORTC7 | PORTC6 | - | - | - | - | - | - | |
| 0x07 (0x27) | DDRC | DDC7 | DDC6 | - | - | - | - | - | - | |
| 0x06 (0x26) | PINC | PINC7 | PINC6 | - | - | - | - | - | - | |
| 0x05 (0x25) | PORTB | PORTB7 | PORTB6 | PORTB5 | PORTB4 | PORTB3 | PORTB2 | PORTB1 | PORTB0 | |
| 0x04 (0x24) | DDRB | DDB7 | DDB6 | DDB5 | DDB4 | DDB3 | DDB2 | DDB1 | DDB0 | |
| 0x03 (0x23) | PINB | PINB7 | PINB6 | PINB5 | PINB4 | PINB3 | PINB2 | PINB1 | PINB0 | |
| 0x02 (0x22) | Reserved | - | - | - | - | - | - | - | - | |
| 0x01 (0x21) | Reserved | - | - | - | - | - | - | - | - | |
| 0x00 (0x20) | Reserved | - | - | - | - | - | - | - | - | |

Note:
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 ATmega16U4/ATmega32U4 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 - $1FF in SRAM, only the ST/STS/STD and LD/LDS/LDD instructions can be used.

# 5. Instruction Set Summary

| Mnemonics | Operands | Description | Operation | Flags | #Clocks |
|---|---|---|---|---|---|
| ARITHMETIC AND LOGIC INSTRUCTIONS | | | | | |
| ADD | Rd, Rr | Add two Registers | Rd ← Rd + Rr | Z,C,N,V,H | 1 |
| ADC | Rd, Rr | Add with Carry two Registers | Rd ← Rd + Rr + C | Z,C,N,V,H | 1 |
| ADIW | Rdl,K | Add Immediate to Word | Rdh:Rdl ← Rdh:Rdl + K | Z,C,N,V,S | 2 |
| SUB | Rd, Rr | Subtract two Registers | Rd ← Rd - Rr | Z,C,N,V,H | 1 |
| SUBI | Rd, K | Subtract Constant from Register | Rd ← Rd - K | Z,C,N,V,H | 1 |
| SBC | Rd, Rr | Subtract with Carry two Registers | Rd ← Rd - Rr - C | Z,C,N,V,H | 1 |
| SBCI | Rd, K | Subtract with Carry Constant from Reg. | Rd ← Rd - K - C | Z,C,N,V,H | 1 |
| SBIW | Rdl,K | Subtract Immediate from Word | Rdh:Rdl ← Rdh:Rdl - K | Z,C,N,V,S | 2 |
| AND | Rd, Rr | Logical AND Registers | Rd ← Rd • Rr | Z,N,V | 1 |
| ANDI | Rd, K | Logical AND Register and Constant | Rd ← Rd • K | Z,N,V | 1 |
| OR | Rd, Rr | Logical OR Registers | Rd ← Rd v Rr | Z,N,V | 1 |
| ORI | Rd, K | Logical OR Register and Constant | Rd ← Rd v K | Z,N,V | 1 |
| EOR | Rd, Rr | Exclusive OR Registers | Rd ← Rd ⊕ Rr | Z,N,V | 1 |
| COM | Rd | One's Complement | Rd ← 0xFF – Rd | Z,C,N,V | 1 |
| NEG | Rd | Two's Complement | Rd ← 0x00 – Rd | Z,C,N,V,H | 1 |
| SBR | Rd,K | Set Bit(s) in Register | Rd ← Rd v K | Z,N,V | 1 |
| CBR | Rd,K | Clear Bit(s) in Register | Rd ← Rd • (0xFF - K) | Z,N,V | 1 |
| INC | Rd | Increment | Rd ← Rd + 1 | Z,N,V | 1 |
| DEC | Rd | Decrement | Rd ← Rd – 1 | Z,N,V | 1 |
| TST | Rd | Test for Zero or Minus | Rd ← Rd • Rd | Z,N,V | 1 |
| CLR | Rd | Clear Register | Rd ← Rd ⊕ Rd | Z,N,V | 1 |
| SER | Rd | Set Register | Rd ← 0xFF | None | 1 |
| MUL | Rd, Rr | Multiply Unsigned | R1:R0 ← Rd x Rr | Z,C | 2 |
| MULS | Rd, Rr | Multiply Signed | R1:R0 ← Rd x Rr | Z,C | 2 |
| MULSU | Rd, Rr | Multiply Signed with Unsigned | R1:R0 ← Rd x Rr | Z,C | 2 |
| FMUL | Rd, Rr | Fractional Multiply Unsigned | R1:R0 ← (Rd x Rr) << 1 | Z,C | 2 |
| FMULS | Rd, Rr | Fractional Multiply Signed | R1:R0 ← (Rd x Rr) << 1 | Z,C | 2 |
| FMULSU | Rd, Rr | Fractional Multiply Signed with Unsigned | R1:R0 ← (Rd x Rr) << 1 | Z,C | 2 |
| BRANCH INSTRUCTIONS | | | | | |
| RJMP | k | Relative Jump | PC ← PC + k + 1 | None | 2 |
| IJMP | | Indirect Jump to (Z) | PC ← Z | None | 2 |
| EIJMP | | Extended Indirect Jump to (Z) | PC ←(EIND:Z) | None | 2 |
| JMP | k | Direct Jump | PC ← k | None | 3 |
| RCALL | k | Relative Subroutine Call | PC ← PC + k + 1 | None | 4 |
| ICALL | | Indirect Call to (Z) | PC ← Z | None | 4 |
| EICALL | | Extended Indirect Call to (Z) | PC ←(EIND:Z) | None | 4 |
| CALL | k | Direct Subroutine Call | PC ← k | None | 5 |
| RET | | Subroutine Return | PC ← STACK | None | 5 |
| RETI | | Interrupt Return | PC ← STACK | I | 5 |
| CPSE | Rd,Rr | Compare, Skip if Equal | if (Rd = Rr) PC ← 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 ← PC + 2 or 3 | None | 1/2/3 |
| SBRS | Rr, b | Skip if Bit in Register is Set | if (Rr(b)=1) PC ← PC + 2 or 3 | None | 1/2/3 |
| SBIC | P, b | Skip if Bit in I/O Register Cleared | if (P(b)=0) PC ← 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 ← PC + 2 or 3 | None | 1/2/3 |
| BRBS | s, k | Branch if Status Flag Set | if (SREG(s) = 1) then PC←PC+k + 1 | None | 1/2 |
| BRBC | s, k | Branch if Status Flag Cleared | if (SREG(s) = 0) then PC←PC+k + 1 | None | 1/2 |
| BREQ | k | Branch if Equal | if (Z = 1) then PC ← PC + k + 1 | None | 1/2 |
| BRNE | k | Branch if Not Equal | if (Z = 0) then PC ← PC + k + 1 | None | 1/2 |
| BRCS | k | Branch if Carry Set | if (C = 1) then PC ← PC + k + 1 | None | 1/2 |
| BRCC | k | Branch if Carry Cleared | if (C = 0) then PC ← PC + k + 1 | None | 1/2 |
| BRSH | k | Branch if Same or Higher | if (C = 0) then PC ← PC + k + 1 | None | 1/2 |
| BRLO | k | Branch if Lower | if (C = 1) then PC ← PC + k + 1 | None | 1/2 |
| BRMI | k | Branch if Minus | if (N = 1) then PC ← PC + k + 1 | None | 1/2 |
| BRPL | k | Branch if Plus | if (N = 0) then PC ← PC + k + 1 | None | 1/2 |
| BRGE | k | Branch if Greater or Equal, Signed | if (N ⊕ V= 0) then PC ← PC + k + 1 | None | 1/2 |
| BRLT | k | Branch if Less Than Zero, Signed | if (N ⊕ V= 1) then PC ← PC + k + 1 | None | 1/2 |
| BRHS | k | Branch if Half Carry Flag Set | if (H = 1) then PC ← PC + k + 1 | None | 1/2 |
| BRHC | k | Branch if Half Carry Flag Cleared | if (H = 0) then PC ← PC + k + 1 | None | 1/2 |
| BRTS | k | Branch if T Flag Set | if (T = 1) then PC ← PC + k + 1 | None | 1/2 |
| BRTC | k | Branch if T Flag Cleared | if (T = 0) then PC ← PC + k + 1 | None | 1/2 |
| BRVS | k | Branch if Overflow Flag is Set | if (V = 1) then PC ← PC + k + 1 | None | 1/2 |

| Mnemonics | Operands | Description | Operation | Flags | #Clocks |
|---|---|---|---|---|---|
| BRVC | k | Branch if Overflow Flag is Cleared | if (V = 0) then PC ← PC + k + 1 | None | 1/2 |
| 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 |
| ELPM | | Extended Load Program Memory | R0 ← (RAMPZ:Z) | None | 3 |
| ELPM | Rd, Z | Extended Load Program Memory | Rd ← (Z) | None | 3 |
| ELPM | Rd, Z+ | Extended Load Program Memory | Rd ← (RAMPZ:Z), RAMPZ:Z ←RAMPZ:Z+1 | None | 3 |

| Mnemonics | Operands | Description | Operation | Flags | #Clocks |
|---|---|---|---|---|---|
| SPM | | Store Program Memory | (Z) ← R1:R0 | None | - |
| IN | Rd, P | In Port | Rd ← P | None | 1 |
| OUT | P, Rr | Out Port | P ← Rr | None | 1 |
| PUSH | Rr | Push Register on Stack | STACK ← Rr | None | 2 |
| POP | Rd | Pop Register from Stack | Rd ← STACK | None | 2 |
| MCU CONTROL INSTRUCTIONS | | | | | |
| NOP | | No Operation | | None | 1 |
| SLEEP | | Sleep | (see specific description for Sleep function) | None | 1 |
| WDR | | Watchdog Reset | (see specific description for WDR/timer) | None | 1 |
| BREAK | | Break | For On-chip Debug Only | None | N/A |

# 6. Ordering Information

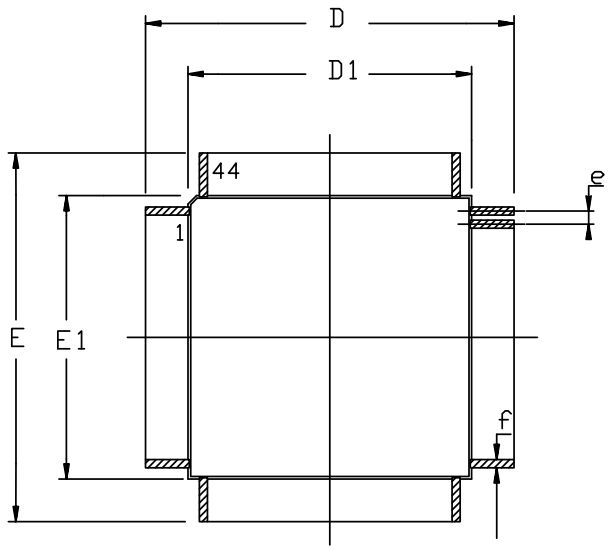
**Table 6-1.** Possible Order Entries

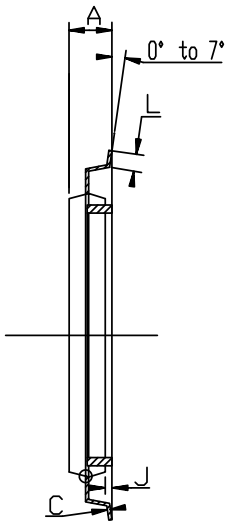
| Ordering Code | USB interface | Speed (MHz) | Power Supply (V) | Package | Operation Range | Product Marking |
|---|---|---|---|---|---|---|
| ATmega32U4-16AU | Device only | 8-16 | 2.7 - 5.5 | TQFP44 | Industrial (-40° to +85°C) Green | mega32U4-16AU |
| ATmega32U4-16MU | Device only | 8-16 | 2.7 - 5.5 | QFN44 | Industrial (-40° to +85°C) Green | mega32U4-16MU |

# 7. Package Information

| Package Type | |
|---|---|
| ML | ML, 44 - Lead, 10 x 10 mm Body Size, 1.0 mm Body Thickness<br>0.8 mm Lead Pitch, Thin Profile Plastic Quad Flat Package (TQFP) |
| PW | PW, 44 - Lead 7.0 x 7.0 mm Body, 0.50 mm Pitch<br>Quad Flat No Lead Package (QFN) |

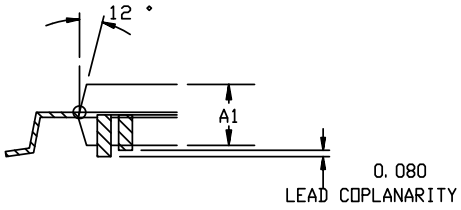
## 7.1 TQFP44




COMMON DIMENSIONS IN MM

| SYMBOL | Min | Max | NOTES |
|---|---|---|---|
| A | ---- | 1.20 | |
| A1 | 0.95 | 1.05 | |
| C | 0.09 | 0.20 | |
| D | 12.00 BSC | | |
| D1 | 10.00 BSC | | |
| E | 12.00 BSC | | |
| E1 | 10.00 BSC | | |
| J | 0.05 | 0.15 | |
| L | 0.45 | 0.75 | |
| e | 0.80 BSC | | |
| f | 0.30 | 0.45 | |




07/27/07

| Atmel Nantes S.A.<br>La Chantrerie - BP 70602<br>44306 Nantes Cedex 3 - France | TITLE<br>ML, 44 - Lead, 10x10 mm Body Size, 1.0 mm Body Thickness<br>0.8 mm Lead Pitch, Thin Profile Plastic Quad Flat Package (TQFP) | DRAWING No.<br>ML | REV.<br>G |
|---|---|---|---|

NOTES: STANDARD NOTES FOR PQFP/VQFP/TQFP/DQFP

1. DIMENSIONING & TOLERANCING CONFORM TO ASME Y14.5M. – 1982.

2. "D1 AND E1" DIMENSIONS DO NOT INCLUDE MOLD PROTUSIONS
MOLD PROTUSIONS SHALL NOT EXCEED 0.25 mm (0.010 INCH) .
THE TOP PACKAGE BODY SIZE MAY BE SMALLER THAN THE BOTTOM
PACKAGE BODY SIZE BY AS MUCH AS 0.15 mm.

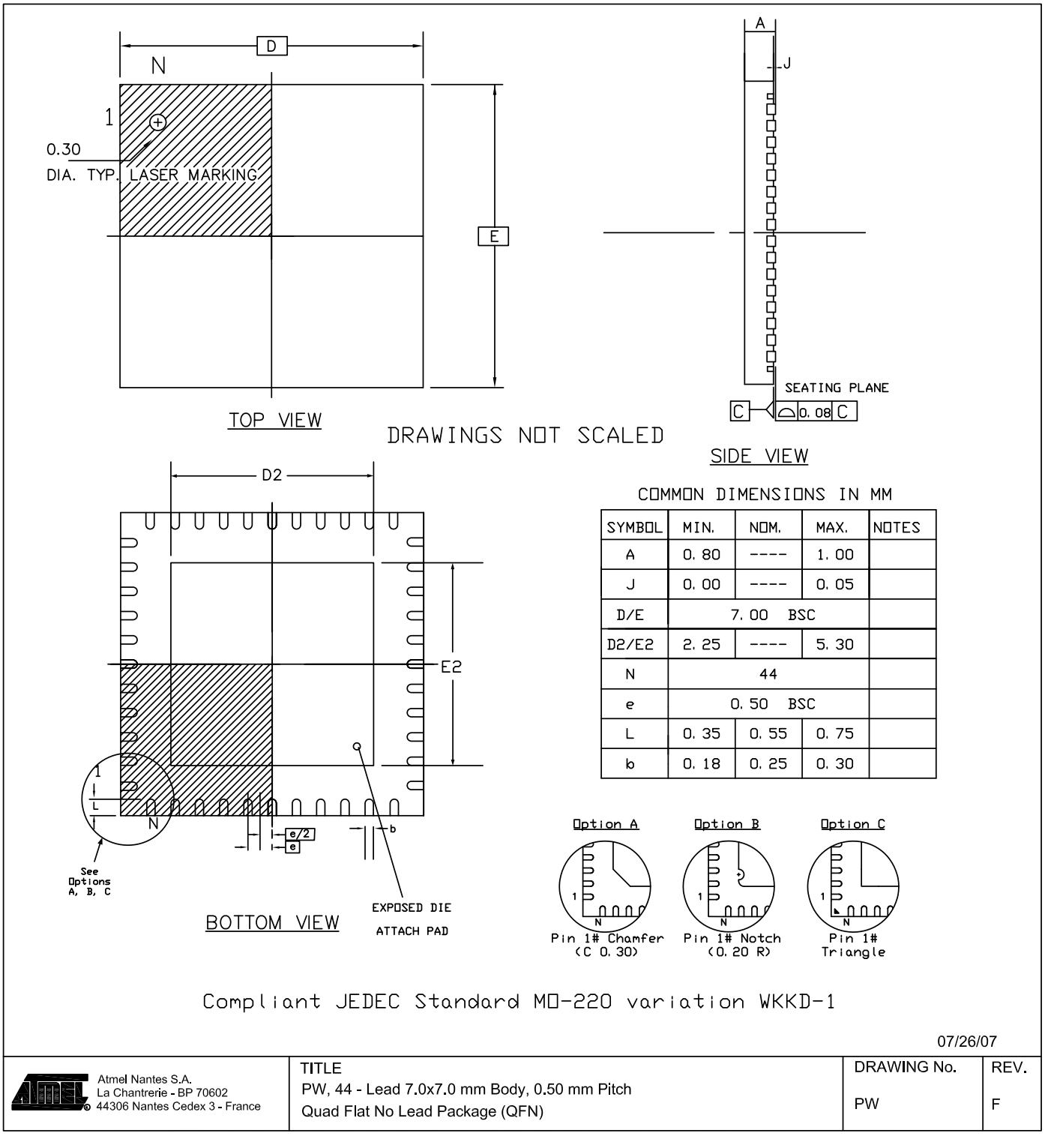
3. DATUM PLANE "H" LOCATED AT MOLD PARTING LINE AND
COINCIDENT WITH LEAD, WHERE LEAD EXISTS PLASTIC BODY AT
BOTTOM OF PARTING LINE.

4. DATUM "A" AND "D" TO BE DETERMINED AT DATUM PLANE H.

5. DIMENSION "f" DOES NOT INCLUDE DAMBAR PROTUSION ALLOWABLE
DAMBAR PROTUSION SHALL BE 0.08 mm/.003" TOTAL EXCESS OF THE
"f" DIMENSION AT MAXIMUM MATERIAL CONDITION.
DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT.

| | | | |
|---|---|---|---|
| Atmel Nantes S.A.<br>La Chantrerie - BP 70602<br>44306 Nantes Cedex 3 - France | TITLE | DRAWING No. | REV. |

## 7.2 QFN44




COMMON DIMENSIONS IN MM

| SYMBOL | MIN. | NOM. | MAX. | NOTES |
|---|---|---|---|---|
| A | 0.80 | ---- | 1.00 | |
| J | 0.00 | ---- | 0.05 | |
| D/E | | 7.00 BSC | | |
| D2/E2 | 2.25 | ---- | 5.30 | |
| N | | 44 | | |
| e | | 0.50 BSC | | |
| L | 0.35 | 0.55 | 0.75 | |
| b | 0.18 | 0.25 | 0.30 | |

Compliant JEDEC Standard MO-220 variation WKKD-1

07/26/07

| | | | |
|---|---|---|---|
| Atmel Nantes S.A.<br>La Chantrerie - BP 70602<br>44306 Nantes Cedex 3 - France | TITLE<br>PW, 44 - Lead 7.0x7.0 mm Body, 0.50 mm Pitch<br>Quad Flat No Lead Package (QFN) | DRAWING No.<br>PW | REV.<br>F |

# 8. Errata

The revision letter in this section refers to the revision of the ATmega16U4/ATmega32U4 device.

## 8.1 ATmega16U4/ATmega32U4 Rev A

1. **Spike on TWI pins when TWI is enabled**
   100 ns negative spike occurs on SDA and SCL pins when TWI is enabled.

   **Problem Fix/work around**
   No known work around, enable ATmega16U4/ATmega32U4 TWI first versus the others nodes of the TWI network.

2. **High current consumption in sleep mode**
   If a pending interrupt cannot wake the part up from the selected mode, the current consumption will increase during sleep when executing the SLEEP instruction directly after a SEI instruction.

   **Problem Fix/work around**
   Before entering sleep, interrupts not used to wake up the part from the sleep mode should be disabled.

3. **Extra power comsumption**
   The typical power comsumption is increased by about 30µA in power-down mode.

   **Problem Fix/work around**
   None.

4. **Internal RC oscillator start up issue.**
   When the part is configured to start on internal RC, the oscillator may not start properly after power-on.

   **Problem Fix/work around**
   Do not configure the part to start with the internal oscillator (default part configuration is to start with the external crystal oscillator).

5. **Internal RC oscillator calibration issue.**
   The default internal RC oscillator frequency may be lower that 8MHz.

   **Problem Fix/work around**
   Parts are configured so that the internal RC oscillator frequency is as close as possible to the 8MHz default target frequency.

9. **Incorrect CPU behavior for VBUSTI interrupt routine**

   The CPU core may incorrectly execute the interrupt vector related to the VBUSTI interrupt flag.

   **Problem fix/workaround**
   Do not enable this interrupt, firmware must process this USB event by polling VBUSTI.

# 9. Datasheet Revision History for ATmega16U4/ATmega32U4

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

## 9.1 Revision A.

1. Initial document version.

## 9.2 Revision B.

1. Added ATmega16U4 device.
2. Created errata section and added ATmega16U4.
3. Update High Speed Timer, asynchronous description Section 15. on page 139.

## 9.3 Revision C.

1. Update errata section.

## Headquarters

***Atmel Corporation***
2325 Orchard Parkway
San Jose, CA 95131
USA
Tel: 1(408) 441-0311
Fax: 1(408) 487-2600

## International

***Atmel Asia***
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimshatsui
East Kowloon
Hong Kong
Tel: (852) 2721-9778
Fax: (852) 2722-1369

***Atmel Europe***
Le Krebs
8, Rue Jean-Pierre Timbaud
BP 309
78054 Saint-Quentin-en-
Yvelines Cedex
France
Tel: (33) 1-30-60-70-00
Fax: (33) 1-30-60-71-11

***Atmel Japan***
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
Tel: (81) 3-3523-3551
Fax: (81) 3-3523-7581

## Product Contact

***Web Site***
www.atmel.com

***Technical Support***
avr@atmel.com

***Sales Contact***
www.atmel.com/contacts

***Literature Requests***
www.atmel.com/literature