

Features

- High Performance, Low Power AVR[®] 8-bit Microcontroller
- Advanced RISC Architecture
 - 131 Powerful Instructions - Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 1 MIPS throughput per MHz
 - On-chip 2-cycle Multiplier
- Data and Non-Volatile Program Memory
 - 16K/32K/64K Bytes Flash of In-System Programmable Program Memory
 - Endurance: 10,000 Write/Erase Cycles
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - 512/1024/2048 Bytes of In-System Programmable EEPROM
 - Endurance: 50,000 Write/Erase Cycles
- Programming Lock for Flash Program and EEPROM Data Security
- 1024/2048/4096 Bytes Internal SRAM
- On Chip Debug Interface (debugWIRE)
- CAN 2.0A/B with 6 Message Objects - ISO 16845 Certified⁽¹⁾
- LIN 2.1 and 1.3 Controller or 8-Bit UART
- One 12-bit High Speed PSC (Power Stage Controller) (only ATmega16/32/64M1)
 - Non Overlapping Inverted PWM Output Pins With Flexible Dead-Time
 - Variable PWM duty Cycle and Frequency
 - Synchronous Update of all PWM Registers
 - Auto Stop Function for Emergency Event
- Peripheral Features
 - One 8-bit General purpose Timer/Counter with Separate Prescaler, Compare Mode and Capture Mode
 - One 16-bit General purpose Timer/Counter with Separate Prescaler, Compare Mode and Capture Mode
 - One Master/Slave SPI Serial Interface
 - 10-bit ADC
 - Up To 11 Single Ended Channels and 3 Fully Differential ADC Channel Pairs
 - Programmable Gain (5x, 10x, 20x, 40x) on Differential Channels
 - Internal Reference Voltage
 - Direct Power Supply Voltage Measurement
 - 10-bit DAC for Variable Voltage Reference (Comparators, ADC)
 - Four Analog Comparators with Variable Threshold Detection
 - 100µA ±3% Current Source (LIN Node Identification)
 - Interrupt and Wake-up on Pin Change
 - Programmable Watchdog Timer with Separate On-Chip Oscillator
 - On-chip Temperature Sensor
- Special Microcontroller Features
 - Low Power Idle, Noise Reduction, and Power Down Modes
 - Power On Reset and Programmable Brown Out Detection
 - In-System Programmable via SPI Port
 - High Precision Crystal Oscillator for CAN Operations (16 MHz)

1. See certification on Atmel web site. And note on [Section 16.4.3 on page 175](#).



**8-bit AVR[®]
Microcontroller
with
16K/32K/64K
Bytes In-System
Programmable
Flash**

**ATmega16M1
ATmega32M1
ATmega64M1
ATmega32C1
ATmega64C1**

Automotive

Preliminary

Summary



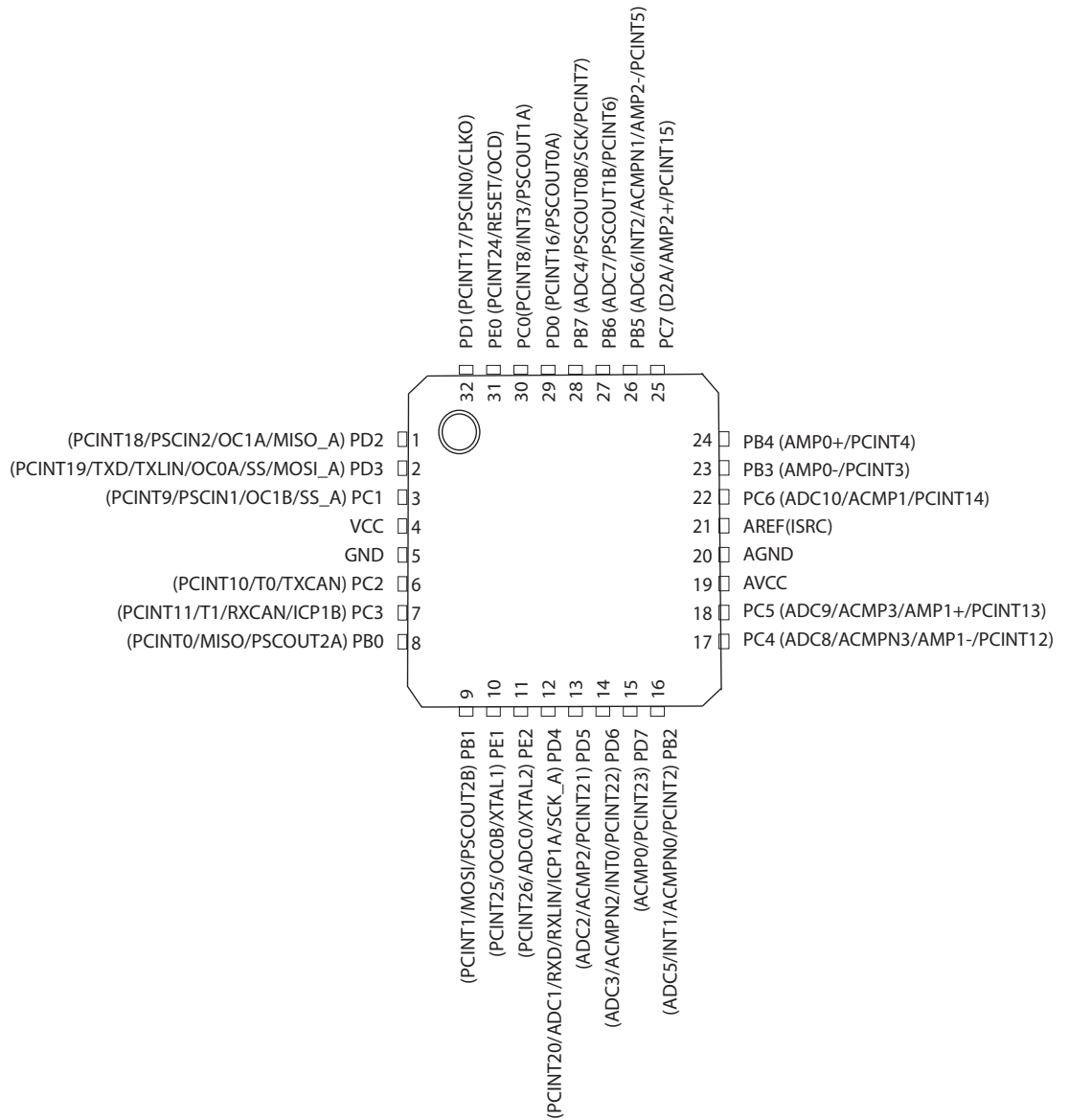
- Internal Calibrated RC Oscillator (8 MHz)
- On-chip PLL for fast PWM (32 MHz, 64 MHz) and CPU (16 MHz)
- Operating Voltage: 2.7V - 5.5V
- Extended Operating Temperature:
 - -40°C to +125°C
- Core Speed Grade:
 - 0 - 8MHz @ 2.7 - 4.5V
 - 0 - 16MHz @ 4.5 - 5.5V

ATmega32/64/M1/C1 Product Line-up

| Part number | ATmega32C1 | ATmega64C1 | ATmega16M1 | ATmega32M1 | ATmega64M1 |
|----------------------|-----------------------------|------------|------------|------------|------------|
| Flash Size | 32 Kbyte | 64 Kbyte | 16 Kbyte | 32 Kbyte | 64 Kbyte |
| RAM Size | 2048 bytes | 4096 bytes | 1024 bytes | 2048 bytes | 4096 bytes |
| EEPROM Size | 1024 bytes | 2048 bytes | 512 bytes | 1024 bytes | 2048 bytes |
| 8-bit Timer | Yes | | | | |
| 16-bit Timer | Yes | | | | |
| PSC | No | | | Yes | |
| PWM Outputs | 4 | 4 | 10 | 10 | 10 |
| Fault Inputs (PSC) | 0 | 0 | 3 | 3 | 3 |
| PLL | 32/64 MHz | | | | |
| 10-bit ADC Channels | 11 single 3 Differential | | | | |
| 10-bit DAC | Yes | | | | |
| Analog Comparators | 4 | | | | |
| Current Source | Yes | | | | |
| CAN | Yes | | | | |
| LIN/UART | Yes | | | | |
| On-Chip Temp. Sensor | Yes | | | | |
| SPI Interface | Yes | | | | |

1. Pin Configurations

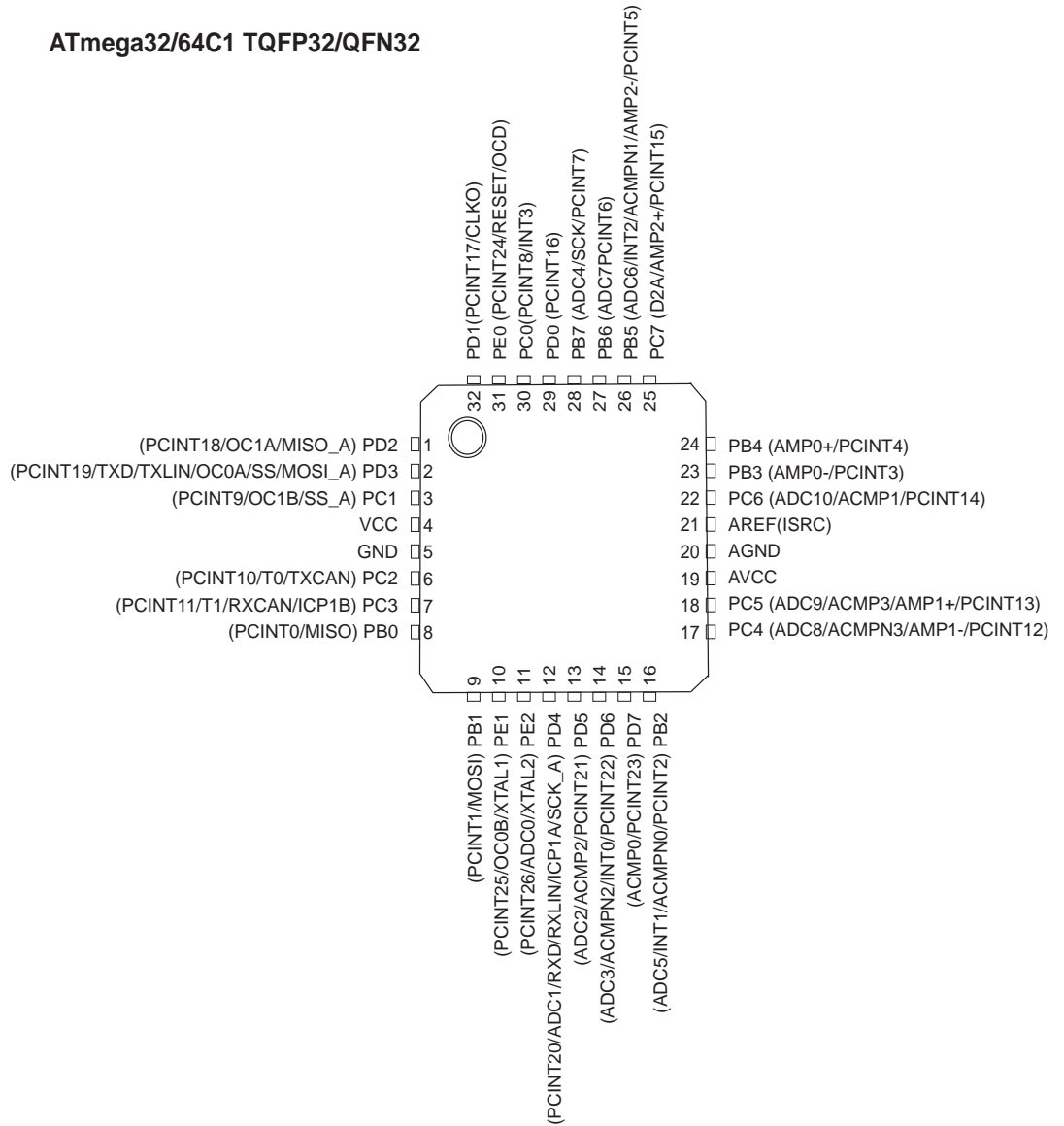
Figure 1-1. ATmega16/32/64M1 TQFP32/QFN32 (7*7 mm) Package.



Note: On the engineering samples (Parts marked AT90PWM324), the ACMPN3 alternate function is not located on PC4. It is located on PE2.

Figure 1-2. ATmega32/64C1 TQFP32/QFN32 (7*7 mm) Package

ATmega32/64C1 TQFP32/QFN32



Note: On the first engineering samples (Parts marked AT90PWM324), the ACMPN3 alternate function is not located on PC4. It is located on PE2.

1.1 Pin Descriptions

:

Table 1-1. Pin out description

| QFN32 Pin Number | Mnemonic | Type | Name, Function & Alternate Function |
|------------------|----------|-------|--|
| 5 | GND | Power | Ground: 0V reference |
| 20 | AGND | Power | Analog Ground: 0V reference for analog part |
| 4 | VCC | Power | Power Supply |
| 19 | AVCC | Power | Analog Power Supply: This is the power supply voltage for analog part For a normal use this pin must be connected. |
| 21 | AREF | Power | Analog Reference : reference for analog converter . This is the reference voltage of the A/D converter. As output, can be used by external analog ISRC (Current Source Output) |
| 8 | PB0 | I/O | MISO (SPI Master In Slave Out) PSCOUT2A ⁽¹⁾ (PSC Module 2 Output A) PCINT0 (Pin Change Interrupt 0) |
| 9 | PB1 | I/O | MOSI (SPI Master Out Slave In) PSCOUT2B ⁽¹⁾ (PSC Module 2 Output B) PCINT1 (Pin Change Interrupt 1) |
| 16 | PB2 | I/O | ADC5 (Analog Input Channel 5) INT1 (External Interrupt 1 Input) ACMPN0 (Analog Comparator 0 Negative Input) PCINT2 (Pin Change Interrupt 2) |
| 23 | PB3 | I/O | AMP0- (Analog Differential Amplifier 0 Negative Input) PCINT3 (Pin Change Interrupt 3) |
| 24 | PB4 | I/O | AMP0+ (Analog Differential Amplifier 0 Positive Input) PCINT4 (Pin Change Interrupt 4) |
| 26 | PB5 | I/O | ADC6 (Analog Input Channel 6) INT2 (External Interrupt 2 Input) ACMPN1 (Analog Comparator 1 Negative Input) AMP2- (Analog Differential Amplifier 2 Negative Input) PCINT5 (Pin Change Interrupt 5) |
| 27 | PB6 | I/O | ADC7 (Analog Input Channel 7) PSCOUT1B ⁽¹⁾ (PSC Module 1 Output A) PCINT6 (Pin Change Interrupt 6) |
| 28 | PB7 | I/O | ADC4 (Analog Input Channel 4) PSCOUT0B ⁽¹⁾ (PSC Module 0 Output B) SCK (SPI Clock) PCINT7 (Pin Change Interrupt 7) |
| 30 | PC0 | I/O | PSCOUT1A ⁽¹⁾ (PSC Module 1 Output A) INT3 (External Interrupt 3 Input) PCINT8 (Pin Change Interrupt 8) |

Table 1-1. Pin out description (Continued)

| QFN32 Pin Number | Mnemonic | Type | Name, Function & Alternate Function |
|------------------|----------|------|--|
| 3 | PC1 | I/O | PSCIN1 (PSC Digital Input 1) OC1B (Timer 1 Output Compare B) SS_A (Alternate SPI Slave Select) PCINT9 (Pin Change Interrupt 9) |
| 6 | PC2 | I/O | T0 (Timer 0 clock input) TXCAN (CAN Transmit Output) PCINT10 (Pin Change Interrupt 10) |
| 7 | PC3 | I/O | T1 (Timer 1 clock input) RXCAN (CAN Receive Input) ICP1B (Timer 1 input capture alternate B input) PCINT11 (Pin Change Interrupt 11) |
| 17 | PC4 | I/O | ADC8 (Analog Input Channel 8) AMP1- (Analog Differential Amplifier 1 Negative Input) ACMPN3 (Analog Comparator 3 Negative Input) PCINT12 (Pin Change Interrupt 12) |
| 18 | PC5 | I/O | ADC9 (Analog Input Channel 9) AMP1+ (Analog Differential Amplifier 1 Positive Input) ACMP3 (Analog Comparator 3 Positive Input) PCINT13 (Pin Change Interrupt 13) |
| 22 | PC6 | I/O | ADC10 (Analog Input Channel 10) ACMP1 (Analog Comparator 1 Positive Input) PCINT14 (Pin Change Interrupt 14) |
| 25 | PC7 | I/O | D2A (DAC output) AMP2+ (Analog Differential Amplifier 2 Positive Input) PCINT15 (Pin Change Interrupt 15) |
| 29 | PD0 | I/O | PSCOUT0A ⁽¹⁾ (PSC Module 0 Output A) PCINT16 (Pin Change Interrupt 16) |
| 32 | PD1 | I/O | PSCIN0 (PSC Digital Input 0) CLKO (System Clock Output) PCINT17 (Pin Change Interrupt 17) |
| 1 | PD2 | I/O | OC1A (Timer 1 Output Compare A) PSCIN2 (PSC Digital Input 2) MISO_A (Programming & alternate SPI Master In Slave Out) PCINT18 (Pin Change Interrupt 18) |
| 2 | PD3 | I/O | TXD (UART Tx data) TXLIN (LIN Transmit Output) OC0A (Timer 0 Output Compare A) SS (SPI Slave Select) MOSI_A (Programming & alternate Master Out SPI Slave In) PCINT19 (Pin Change Interrupt 19) |

Table 1-1. Pin out description (Continued)

| QFN32 Pin Number | Mnemonic | Type | Name, Function & Alternate Function |
|------------------|----------|----------|---|
| 12 | PD4 | I/O | ADC1 (Analog Input Channel 1) RXD (UART Rx data) RXLIN (LIN Receive Input) ICP1A (Timer 1 input capture alternate A input) SCK_A (Programming & alternate SPI Clock) PCINT20 (Pin Change Interrupt 20) |
| 13 | PD5 | I/O | ADC2 (Analog Input Channel 2) ACMP2 (Analog Comparator 2 Positive Input) PCINT21 (Pin Change Interrupt 21) |
| 14 | PD6 | I/O | ADC3 (Analog Input Channel 3) ACMPN2 (Analog Comparator 2 Negative Input) INT0 (External Interrupt 0 Input) PCINT22 (Pin Change Interrupt 22) |
| 15 | PD7 | I/O | ACMP0 (Analog Comparator 0 Positive Input) PCINT23 (Pin Change Interrupt 23) |
| 31 | PE0 | I/O or I | RESET (Reset Input) OCD (On Chip Debug I/O) PCINT24 (Pin Change Interrupt 24) |
| 10 | PE1 | I/O | XTAL1 (XTAL Input) OC0B (Timer 0 Output Compare B) PCINT25 (Pin Change Interrupt 25) |
| 11 | PE2 | I/O | XTAL2 (XTAL Output) ADC0 (Analog Input Channel 0) PCINT26 (Pin Change Interrupt 26) |

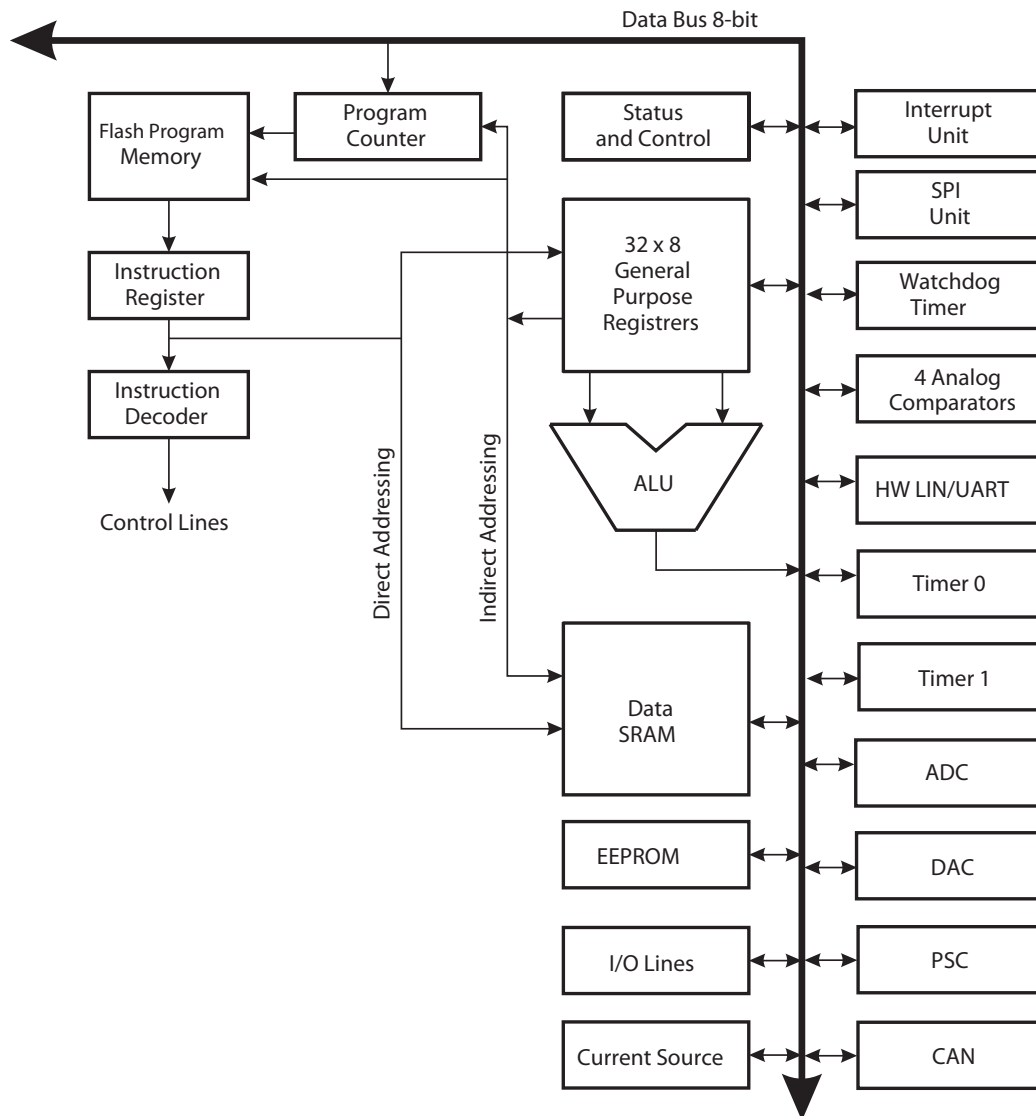
- Note:
1. Only for ATmega32/64M1.
 2. On the first engineering samples (Parts marked AT90PWM324), the ACMPN3 alternate function is not located on PC4. It is located on PE2.

2. Overview

The ATmega16/32/64/M1/C1 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 ATmega16/32/64/M1/C1 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 ATmega16/32/64/M1/C1 provides the following features: 16K/32K/64K bytes of In-System Programmable Flash with Read-While-Write capabilities, 512/1024/2048 bytes EEPROM, 1024/2048/4096 bytes SRAM, 27 general purpose I/O lines, 32 general purpose working registers, one Motor Power Stage Controller, two flexible Timer/Counters with compare modes and PWM, one UART with HW LIN, an 11-channel 10-bit ADC with two differential input stages with programmable gain, a 10-bit DAC, a programmable Watchdog Timer with Internal Individual Oscillator, an SPI serial port, an On-chip Debug system and four software selectable power saving modes.



The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, SPI ports, CAN, LIN/UART 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 ATmega16/32/64/M1/C1 is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications.

The ATmega16/32/64/M1/C1 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 Automotive Quality Grade

The ATmega16/32/64/M1/C1 have been developed and manufactured according to the most stringent requirements of the international standard ISO-TS-16949. This data sheet contains limit values extracted from the results of extensive characterization (Temperature and Voltage). The quality and reliability of the ATmega16/32/64/M1/C1 have been verified during regular product qualification as per AEC-Q100 grade 1.

As indicated in the ordering information paragraph, the products are available in only one temperature grade.

Table 2-1. Temperature Grade Identification for Automotive Products

| Temperature | Temperature Identifier | Comments |
|-------------|------------------------|-----------------------------------|
| -40 ; +125 | Z | Full Automotive Temperature Range |

2.3 Pin Descriptions

2.3.1 VCC

Digital supply voltage.

2.3.2 GND

Ground.

2.3.3 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 also serves the functions of various special features of the ATmega16/32/64/M1/C1 as listed on [page 68](#).

2.3.4 Port C (PC7..PC0)

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.

Port C also serves the functions of special features of the ATmega16/32/64/M1/C1 as listed on [page 72](#).

2.3.5 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 ATmega16/32/64/M1/C1 as listed on [page 75](#).

2.3.6 Port E (PE2..0) $\overline{\text{RESET}}$ /XTAL1/ XTAL2

Port E is an 3-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.

If the RSTDISBL Fuse is programmed, PE0 is used as an I/O pin. Note that the electrical characteristics of PE0 differ from those of the other pins of Port E.

If the RSTDISBL Fuse is unprogrammed, PE0 is used as a 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 7-1 on page 46](#). Shorter pulses are not guaranteed to generate a Reset.

Depending on the clock selection fuse settings, PE1 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

Depending on the clock selection fuse settings, PE2 can be used as output from the inverting Oscillator amplifier.

The various special features of Port E are elaborated in [“Alternate Functions of Port E”](#) on page 78 and [“Clock Systems and their Distribution”](#) on page 29.

2.3.7 AVCC

AVCC is the supply voltage pin for the A/D Converter, D/A Converter, Current source. It should be externally connected to V_{CC} , even if the ADC, DAC are not used. If the ADC is used, it should be connected to V_{CC} through a low-pass filter.

2.3.8 AREF

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

2.4 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.

3. 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) | CANMSG | MSG 7 | MSG 6 | MSG 5 | MSG 4 | MSG 3 | MSG 2 | MSG 1 | MSG 0 | page 199 |
| (0xF9) | CANSTMPH | TIMSTM15 | TIMSTM14 | TIMSTM13 | TIMSTM12 | TIMSTM11 | TIMSTM10 | TIMSTM9 | TIMSTM8 | page 199 |
| (0xF8) | CANSTMPL | TIMSTM7 | TIMSTM6 | TIMSTM5 | TIMSTM4 | TIMSTM3 | TIMSTM2 | TIMSTM1 | TIMSTM0 | page 199 |
| (0xF7) | CANIDM1 | IDMSK28 | IDMSK27 | IDMSK26 | IDMSK25 | IDMSK24 | IDMSK23 | IDMSK22 | IDMSK21 | page 198 |
| (0xF6) | CANIDM2 | IDMSK20 | IDMSK19 | IDMSK18 | IDMSK17 | IDMSK16 | IDMSK15 | IDMSK14 | IDMSK13 | page 198 |
| (0xF5) | CANIDM3 | IDMSK12 | IDMSK11 | IDMSK10 | IDMSK9 | IDMSK8 | IDMSK7 | IDMSK6 | IDMSK5 | page 198 |
| (0xF4) | CANIDM4 | IDMSK4 | IDMSK3 | IDMSK2 | IDMSK1 | IDMSK0 | RTRMSK | – | IDEMSK | page 198 |
| (0xF3) | CANIDT1 | IDT28 | IDT27 | IDT26 | IDT25 | IDT24 | IDT23 | IDT22 | IDT21 | page 196 |
| (0xF2) | CANIDT2 | IDT20 | IDT19 | IDT18 | IDT17 | IDT16 | IDT15 | IDT14 | IDT13 | page 196 |
| (0xF1) | CANIDT3 | IDT12 | IDT11 | IDT10 | IDT9 | IDT8 | IDT7 | IDT6 | IDT5 | page 196 |
| (0xF0) | CANIDT4 | IDT4 | IDT3 | IDT2 | IDT1 | IDT0 | RTRTAG | RB1TAG | RB0TAG | page 196 |
| (0xEF) | CANCDMOB | CONMOB1 | CONMOB0 | RPLV | IDE | DLC3 | DLC2 | DLC1 | DLC0 | page 195 |
| (0xEE) | CANSTMOB | DLCW | TXOK | RXOK | BERR | SERR | CERR | FERR | AERR | page 194 |
| (0xED) | CANPAGE | MOBNB3 | MOBNB2 | MOBNB1 | MOBNB0 | AINC | INDX2 | INDX1 | INDX0 | page 194 |
| (0xEC) | CANHPMOB | HPMOB3 | HPMOB2 | HPMOB1 | HPMOB0 | CGP3 | CGP2 | CGP1 | CGP0 | page 193 |
| (0xEB) | CANREC | REC7 | REC6 | REC5 | REC4 | REC3 | REC2 | REC1 | REC0 | page 193 |
| (0xEA) | CANTEC | TEC7 | TEC6 | TEC5 | TEC4 | TEC3 | TEC2 | TEC1 | TEC0 | page 193 |
| (0xE9) | CANTTCH | TIMTTC15 | TIMTTC14 | TIMTTC13 | TIMTTC12 | TIMTTC11 | TIMTTC10 | TIMTTC9 | TIMTTC8 | page 193 |
| (0xE8) | CANTTCL | TIMTTC7 | TIMTTC6 | TIMTTC5 | TIMTTC4 | TIMTTC3 | TIMTTC2 | TIMTTC1 | TIMTTC0 | page 193 |
| (0xE7) | CANTIMH | CANTIM15 | CANTIM14 | CANTIM13 | CANTIM12 | CANTIM11 | CANTIM10 | CANTIM9 | CANTIM8 | page 193 |
| (0xE6) | CANTIML | CANTIM7 | CANTIM6 | CANTIM5 | CANTIM4 | CANTIM3 | CANTIM2 | CANTIM1 | CANTIM0 | page 193 |
| (0xE5) | CANTCON | TPRSC7 | TPRSC6 | TPRSC5 | TPRSC4 | TPRSC3 | TPRSC2 | TPRSC1 | TPRSC0 | page 192 |
| (0xE4) | CANBT3 | – | PHS22 | PHS21 | PHS20 | PHS12 | PHS11 | PHS10 | SMP | page 192 |
| (0xE3) | CANBT2 | – | SJW1 | SJW0 | – | PRS2 | PRS1 | PRS0 | – | page 191 |
| (0xE2) | CANBT1 | – | BRP5 | BRP4 | BRP3 | BRP2 | BRP1 | BRP0 | – | page 190 |
| (0xE1) | CANSIT1 | – | – | – | – | – | – | – | – | page 190 |
| (0xE0) | CANSIT2 | – | – | SIT5 | SIT4 | SIT3 | SIT2 | SIT1 | SIT0 | page 190 |
| (0xDF) | CANIE1 | – | – | – | – | – | – | – | – | page 190 |
| (0xDE) | CANIE2 | – | – | IEMOB5 | IEMOB4 | IEMOB3 | IEMOB2 | IEMOB1 | IEMOB0 | page 190 |
| (0xDD) | CANEN1 | – | – | – | – | – | – | – | – | page 189 |
| (0xDC) | CANEN2 | – | – | ENMOB5 | ENMOB4 | ENMOB3 | ENMOB2 | ENMOB1 | ENMOB0 | page 189 |
| (0xDB) | CANGIE | ENIT | ENBOFF | ENRX | ENTX | ENERR | ENBX | ENERG | ENOVRT | page 188 |
| (0xDA) | CANGIT | CANIT | BOFFIT | OVRTIM | BXOK | SERG | CERG | FERG | AERG | page 187 |
| (0xD9) | CANGSTA | – | OVRG | – | TXBSY | RXBSY | ENFG | BOFF | ERRP | page 186 |
| (0xD8) | CANGCON | ABRQ | OVRQ | TTC | SYNTTC | LISTEN | TEST | ENA/STB | SWRES | page 185 |
| (0xD7) | Reserved | – | – | – | – | – | – | – | – | |
| (0xD6) | Reserved | – | – | – | – | – | – | – | – | |
| (0xD5) | Reserved | – | – | – | – | – | – | – | – | |
| (0xD4) | Reserved | – | – | – | – | – | – | – | – | |
| (0xD3) | Reserved | – | – | – | – | – | – | – | – | |
| (0xD2) | LINDAT | LDATA7 | LDATA6 | LDATA5 | LDATA4 | LDATA3 | LDATA2 | LDATA1 | LDATA0 | page 226 |
| (0xD1) | LINSEL | – | – | – | – | /LAINC | LINDX2 | LINDX1 | LINDX0 | page 226 |
| (0xD0) | LINIDR | LP1 | LP0 | LID5 / LDL1 | LID4 / LDL0 | LID3 | LID2 | LID1 | LID0 | page 225 |
| (0xCF) | LINDLR | LTXDL3 | LTXDL2 | LTXDL1 | LTXDL0 | LRXDL3 | LRXDL2 | LRXDL1 | LRXDL0 | page 224 |
| (0xCE) | LINBRRH | – | – | – | – | LDIV11 | LDIV10 | LDIV9 | LDIV8 | page 224 |
| (0xCD) | LINBRRL | LDIV7 | LDIV6 | LDIV5 | LDIV4 | LDIV3 | LDIV2 | LDIV1 | LDIV0 | page 224 |
| (0xCC) | LINBTR | LDISR | – | LBT5 | LBT4 | LBT3 | LBT2 | LBT1 | LBT0 | page 224 |
| (0xCB) | LINERR | LABORT | LTOERR | LOVERR | LFERR | LSERR | LPERR | LCERR | LBERR | page 223 |
| (0xCA) | LINENIR | – | – | – | – | LENERR | LENIDOK | LENTXOK | LENRXOK | page 222 |
| (0xC9) | LINSIR | LIDST2 | LIDST1 | LIDST0 | LBUSY | LERR | LIDOK | LTXOK | LRXOK | page 221 |
| (0xC8) | LINCR | LSWRES | LIN13 | LCONF1 | LCONF0 | LENA | LCMD2 | LCMD1 | LCMD0 | page 220 |
| (0xC7) | Reserved | – | – | – | – | – | – | – | – | |
| (0xC6) | Reserved | – | – | – | – | – | – | – | – | |
| (0xC5) | Reserved | – | – | – | – | – | – | – | – | |
| (0xC4) | Reserved | – | – | – | – | – | – | – | – | |
| (0xC3) | Reserved | – | – | – | – | – | – | – | – | |
| (0xC2) | Reserved | – | – | – | – | – | – | – | – | |
| (0xC1) | Reserved | – | – | – | – | – | – | – | – | |
| (0xC0) | Reserved | – | – | – | – | – | – | – | – | |
| (0xBF) | Reserved | – | – | – | – | – | – | – | – | |



| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Page |
|-----------------------|----------|-------------|-------------|----------|----------|-----------|-----------|-------------|-------------|--------------------------|
| (0xBE) | Reserved | – | – | – | – | – | – | – | – | |
| (0xBD) | Reserved | – | – | – | – | – | – | – | – | |
| (0xBC) ⁽⁵⁾ | PIFR | – | – | – | – | PEV2 | PEV1 | PEV0 | PEOP | page 155 |
| (0xBB) ⁽⁵⁾ | PIM | – | – | – | – | PEVE2 | PEVE1 | PEVE0 | PEOPE | page 155 |
| (0xBA) ⁽⁵⁾ | PMIC2 | POVEN2 | PISEL2 | PELEV2 | PFLTE2 | PAOC2 | PRFM22 | PRFM21 | PRFM20 | page 154 |
| (0xB9) ⁽⁵⁾ | PMIC1 | POVEN1 | PISEL1 | PELEV1 | PFLTE1 | PAOC1 | PRFM12 | PRFM11 | PRFM10 | page 154 |
| (0xB8) ⁽⁵⁾ | PMIC0 | POVEN0 | PISEL0 | PELEV0 | PFLTE0 | PAOC0 | PRFM02 | PRFM01 | PRFM00 | page 154 |
| (0xB7) ⁽⁵⁾ | PCTL | PPRE1 | PPRE0 | PCLKSEL | – | – | – | PCCYC | PRUN | page 153 |
| (0xB6) ⁽⁵⁾ | POC | – | – | POEN2B | POEN2A | POEN1B | POEN1A | POEN0B | POEN0A | page 149 |
| (0xB5) ⁽⁵⁾ | PCNF | – | – | PULOCK | PMODE | POPB | POPA | – | – | page 152 |
| (0xB4) ⁽⁵⁾ | PSYNC | – | – | PSYNC21 | PSYNC20 | PSYNC11 | PSYNC10 | PSYNC01 | PSYNC00 | page 150 |
| (0xB3) ⁽⁵⁾ | POCR_RBH | – | – | – | – | POCR_RB11 | POCR_RB10 | POCR_RB9 | POCR_RB8 | page 152 |
| (0xB2) ⁽⁵⁾ | POCR_RBL | POCR_RB7 | POCR_RB6 | POCR_RB5 | POCR_RB4 | POCR_RB3 | POCR_RB2 | POCR_RB1 | POCR_RB0 | page 152 |
| (0xB1) ⁽⁵⁾ | POCR2SBH | – | – | – | – | POCR2SB11 | POCR2SB10 | POCR2SB9 | POCR2SB8 | page 152 |
| (0xB0) ⁽⁵⁾ | POCR2SBL | POCR2SB7 | POCR2SB6 | POCR2SB5 | POCR2SB4 | POCR2SB3 | POCR2SB2 | POCR2SB1 | POCR2SB0 | page 152 |
| (0xAF) ⁽⁵⁾ | POCR2RAH | – | – | – | – | POCR2RA11 | POCR2RA10 | POCR2RA9 | POCR2RA8 | page 151 |
| (0xAE) ⁽⁵⁾ | POCR2RAL | POCR2RA7 | POCR2RA6 | POCR2RA5 | POCR2RA4 | POCR2RA3 | POCR2RA2 | POCR2RA1 | POCR2RA0 | page 151 |
| (0xAD) ⁽⁵⁾ | POCR2SAH | – | – | – | – | POCR2SA11 | POCR2SA10 | POCR2SA9 | POCR2SA8 | page 151 |
| (0xAC) ⁽⁵⁾ | POCR2SAL | POCR2SA7 | POCR2SA6 | POCR2SA5 | POCR2SA4 | POCR2SA3 | POCR2SA2 | POCR2SA1 | POCR2SA0 | page 151 |
| (0xAB) ⁽⁵⁾ | POCR1SBH | – | – | – | – | POCR1SB11 | POCR1SB10 | POCR1SB9 | POCR1SB8 | page 152 |
| (0xAA) ⁽⁵⁾ | POCR1SBL | POCR1SB7 | POCR1SB6 | POCR1SB5 | POCR1SB4 | POCR1SB3 | POCR1SB2 | POCR1SB1 | POCR1SB0 | page 152 |
| (0xA9) ⁽⁵⁾ | POCR1RAH | – | – | – | – | POCR1RA11 | POCR1RA10 | POCR1RA9 | POCR1RA8 | page 151 |
| (0xA8) ⁽⁵⁾ | POCR1RAL | POCR1RA7 | POCR1RA6 | POCR1RA5 | POCR1RA4 | POCR1RA3 | POCR1RA2 | POCR1RA1 | POCR1RA0 | page 151 |
| (0xA7) ⁽⁵⁾ | POCR1SAH | – | – | – | – | POCR1SA11 | POCR1SA10 | POCR1SA9 | POCR1SA8 | page 151 |
| (0xA6) ⁽⁵⁾ | POCR1SAL | POCR1SA7 | POCR1SA6 | POCR1SA5 | POCR1SA4 | POCR1SA3 | POCR1SA2 | POCR1SA1 | POCR1SA0 | page 151 |
| (0xA5) ⁽⁵⁾ | POCR0SBH | – | – | – | – | POCR0SB11 | POCR0SB10 | POCR0SB9 | POCR0SB8 | page 152 |
| (0xA4) ⁽⁵⁾ | POCR0SBL | POCR0SB7 | POCR0SB6 | POCR0SB5 | POCR0SB4 | POCR0SB3 | POCR0SB2 | POCR0SB1 | POCR0SB0 | page 152 |
| (0xA3) ⁽⁵⁾ | POCR0RAH | – | – | – | – | POCR0RA11 | POCR0RA10 | POCR0RA9 | POCR0RA8 | page 151 |
| (0xA2) ⁽⁵⁾ | POCR0RAL | POCR0RA7 | POCR0RA6 | POCR0RA5 | POCR0RA4 | POCR0RA3 | POCR0RA2 | POCR0RA1 | POCR0RA0 | page 151 |
| (0xA1) ⁽⁵⁾ | POCR0SAH | – | – | – | – | POCR0SA11 | POCR0SA10 | POCR0SA9 | POCR0SA8 | page 151 |
| (0xA0) ⁽⁵⁾ | POCR0SAL | POCR0SA7 | POCR0SA6 | POCR0SA5 | POCR0SA4 | POCR0SA3 | POCR0SA2 | POCR0SA1 | POCR0SA0 | page 151 |
| (0x9F) | Reserved | – | – | – | – | – | – | – | – | |
| (0x9E) | Reserved | – | – | – | – | – | – | – | – | |
| (0x9D) | Reserved | – | – | – | – | – | – | – | – | |
| (0x9C) | Reserved | – | – | – | – | – | – | – | – | |
| (0x9B) | Reserved | – | – | – | – | – | – | – | – | |
| (0x9A) | Reserved | – | – | – | – | – | – | – | – | |
| (0x99) | Reserved | – | – | – | – | – | – | – | – | |
| (0x98) | Reserved | – | – | – | – | – | – | – | – | |
| (0x97) | AC3CON | AC3EN | AC3IE | AC3IS1 | AC3IS0 | – | AC3M2 | AC3M1 | AC3M0 | page 263 |
| (0x96) | AC2CON | AC2EN | AC2IE | AC2IS1 | AC2IS0 | – | AC2M2 | AC2M1 | AC2M0 | page 263 |
| (0x95) | AC1CON | AC1EN | AC1IE | AC1IS1 | AC1IS0 | AC1ICE | AC1M2 | AC1M1 | AC1M0 | page 262 |
| (0x94) | AC0CON | AC0EN | AC0IE | AC0IS1 | AC0IS0 | ACCKSEL | AC0M2 | AC0M1 | AC0M0 | page 261 |
| (0x93) | Reserved | – | – | – | – | – | – | – | – | |
| (0x92) | DACH | - / DAC9 | - / DAC8 | - / DAC7 | - / DAC6 | - / DAC5 | - / DAC4 | DAC9 / DAC3 | DAC8 / DAC2 | page 270 |
| (0x91) | DACL | DAC7 / DAC1 | DAC6 / DAC0 | DAC5 / - | DAC4 / - | DAC3 / - | DAC2 / - | DAC1 / - | DAC0 / | page 270 |
| (0x90) | DACON | DAATE | DATS2 | DATS1 | DATS0 | – | DALA | DAOE | DAEN | page 269 |
| (0x8F) | Reserved | – | – | – | – | – | – | – | – | |
| (0x8E) | Reserved | – | – | – | – | – | – | – | – | |
| (0x8D) | Reserved | – | – | – | – | – | – | – | – | |
| (0x8C) | Reserved | – | – | – | – | – | – | – | – | |
| (0x8B) | OCR1BH | OCR1B15 | OCR1B14 | OCR1B13 | OCR1B12 | OCR1B11 | OCR1B10 | OCR1B9 | OCR1B8 | page 131 |
| (0x8A) | OCR1BL | OCR1B7 | OCR1B6 | OCR1B5 | OCR1B4 | OCR1B3 | OCR1B2 | OCR1B1 | OCR1B0 | page 132 |
| (0x89) | OCR1AH | OCR1A15 | OCR1A14 | OCR1A13 | OCR1A12 | OCR1A11 | OCR1A10 | OCR1A9 | OCR1A8 | page 131 |
| (0x88) | OCR1AL | OCR1A7 | OCR1A6 | OCR1A5 | OCR1A4 | OCR1A3 | OCR1A2 | OCR1A1 | OCR1A0 | page 131 |
| (0x87) | ICR1H | ICR115 | ICR114 | ICR113 | ICR112 | ICR111 | ICR110 | ICR19 | ICR18 | page 133 |
| (0x86) | ICR1L | ICR17 | ICR16 | ICR15 | ICR14 | ICR13 | ICR12 | ICR11 | ICR10 | page 133 |
| (0x85) | TCNT1H | TCNT115 | TCNT114 | TCNT113 | TCNT112 | TCNT111 | TCNT110 | TCNT19 | TCNT18 | page 131 |
| (0x84) | TCNT1L | TCNT17 | TCNT16 | TCNT15 | TCNT14 | TCNT13 | TCNT12 | TCNT11 | TCNT10 | page 131 |
| (0x83) | Reserved | – | – | – | – | – | – | – | – | |
| (0x82) | TCCR1C | FOC1A | FOC1B | – | – | – | – | – | – | page 131 |
| (0x81) | TCCR1B | ICNC1 | ICES1 | – | WGM13 | WGM12 | CS12 | CS11 | CS10 | page 130 |
| (0x80) | TCCR1A | COM1A1 | COM1A0 | COM1B1 | COM1B0 | – | – | WGM11 | WGM10 | page 127 |
| (0x7F) | DIDR1 | – | AMP2PD | ACMP0D | AMP0PD | AMP0ND | ADC10D | ADC9D | ADC8D | page 246 |
| (0x7E) | DIDR0 | ADC7D | ADC6D | ADC5D | ADC4D | ADC3D | ADC2D | ADC1D | ADC0D | page 246 |
| (0x7D) | Reserved | – | – | – | – | – | – | – | – | |

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Page |
|-------------|-----------------|------------------------------------|-------------|----------|----------|----------|----------|-------------|-------------|---------------------------------------|
| (0x7C) | ADMUX | REFS1 | REFS0 | ADLAR | – | MUX3 | MUX2 | MUX1 | MUX0 | page 242 |
| (0x7B) | ADCSRB | ADHSM | ISRCEN | AREFEN | – | ADTS3 | ADTS2 | ADTS1 | ADTS0 | page 244 |
| (0x7A) | ADCSRA | ADEN | ADSC | ADATE | ADIF | ADIE | ADPS2 | ADPS1 | ADPS0 | page 243 |
| (0x79) | ADCH | - / ADC9 | - / ADC8 | - / ADC7 | - / ADC6 | - / ADC5 | - / ADC4 | ADC9 / ADC3 | ADC8 / ADC2 | page 245 |
| (0x78) | ADCL | ADC7 / ADC1 | ADC6 / ADC0 | ADC5 / - | ADC4 / - | ADC3 / - | ADC2 / - | ADC1 / - | ADC0 / - | page 245 |
| (0x77) | AMP2CSR | AMP2EN | AMP2IS | AMP2G1 | AMP2G0 | AMPCMP2 | AMP2TS2 | AMP2TS1 | AMP2TS0 | page 252 |
| (0x76) | AMP1CSR | AMP1EN | AMP1IS | AMP1G1 | AMP1G0 | AMPCMP1 | AMP1TS2 | AMP1TS1 | AMP1TS0 | page 252 |
| (0x75) | AMP0CSR | AMP0EN | AMP0IS | AMP0G1 | AMP0G0 | AMPCMP0 | AMP0TS2 | AMP0TS1 | AMP0TS0 | page 251 |
| (0x74) | Reserved | – | – | – | – | – | – | – | – | |
| (0x73) | Reserved | – | – | – | – | – | – | – | – | |
| (0x72) | Reserved | – | – | – | – | – | – | – | – | |
| (0x71) | Reserved | – | – | – | – | – | – | – | – | |
| (0x70) | Reserved | – | – | – | – | – | – | – | – | |
| (0x6F) | TIMSK1 | – | – | ICIE1 | – | – | OCIE1B | OCIE1A | TOIE1 | page 133 |
| (0x6E) | TIMSK0 | – | – | – | – | – | OCIE0B | OCIE0A | TOIE0 | page 105 |
| (0x6D) | PCMSK3 | – | – | – | – | – | PCINT26 | PCINT25 | PCINT24 | page 85 |
| (0x6C) | PCMSK2 | PCINT23 | PCINT22 | PCINT21 | PCINT20 | PCINT19 | PCINT18 | PCINT17 | PCINT16 | page 86 |
| (0x6B) | PCMSK1 | PCINT15 | PCINT14 | PCINT13 | PCINT12 | PCINT11 | PCINT10 | PCINT9 | PCINT8 | page 86 |
| (0x6A) | PCMSK0 | PCINT7 | PCINT6 | PCINT5 | PCINT4 | PCINT3 | PCINT2 | PCINT1 | PCINT0 | page 86 |
| (0x69) | EICRA | ISC31 | ISC30 | ISC21 | ISC20 | ISC11 | ISC10 | ISC01 | ISC00 | page 83 |
| (0x68) | PCICR | – | – | – | – | PCIE3 | PCIE2 | PCIE1 | PCIE0 | page 84 |
| (0x67) | Reserved | – | – | – | – | – | – | – | – | |
| (0x66) | OSCCAL | – | CAL6 | CAL5 | CAL4 | CAL3 | CAL2 | CAL1 | CAL0 | page 33 |
| (0x65) | Reserved | – | – | – | – | – | – | – | – | |
| (0x64) | PRR | – | PRCAN | PRPSC | PRTIM1 | PRTIM0 | PRSPI | PRLIN | PRADC | page 42 |
| (0x63) | Reserved | – | – | – | – | – | – | – | – | |
| (0x62) | Reserved | – | – | – | – | – | – | – | – | |
| (0x61) | CLKPR | CLKPCE | – | – | – | CLKPS3 | CLKPS2 | CLKPS1 | CLKPS0 | page 38 |
| (0x60) | WDTCR | WDIF | WDIE | WDP3 | WDCE | WDE | WDP2 | WDP1 | WDP0 | page 53 |
| 0x3F (0x5F) | SREG | I | T | H | S | V | N | Z | C | page 14 |
| 0x3E (0x5E) | SPH | SP15 | SP14 | SP13 | SP12 | SP11 | SP10 | SP9 | SP8 | page 16 |
| 0x3D (0x5D) | SPL | SP7 | SP6 | SP5 | SP4 | SP3 | SP2 | SP1 | SP0 | page 16 |
| 0x3C (0x5C) | Reserved | – | – | – | – | – | – | – | – | |
| 0x3B (0x5B) | Reserved | – | – | – | – | – | – | – | – | |
| 0x3A (0x5A) | Reserved | – | – | – | – | – | – | – | – | |
| 0x39 (0x59) | Reserved | – | – | – | – | – | – | – | – | |
| 0x38 (0x58) | Reserved | – | – | – | – | – | – | – | – | |
| 0x37 (0x57) | SPMCSR | SPMIE | RWWWSB | SIGRD | RWWSRE | BLBSET | PGWRT | PGERS | SPMEN | page 281 |
| 0x36 (0x56) | Reserved | – | – | – | – | – | – | – | – | |
| 0x35 (0x55) | MCUCR | SPIPS | – | – | PUD | – | – | IVSEL | IVCE | page 59 & page 68 |
| 0x34 (0x54) | MCUSR | – | – | – | – | WDRF | BORF | EXTRF | PORF | page 49 |
| 0x33 (0x53) | SMCR | – | – | – | – | SM2 | SM1 | SM0 | SE | page 40 |
| 0x32 (0x52) | MSMCR | Monitor Stop Mode Control Register | | | | | | | | reserved |
| 0x31 (0x51) | MONDR | Monitor Data Register | | | | | | | | reserved |
| 0x30 (0x50) | ACSR | AC3IF | AC2IF | AC1IF | AC0IF | AC3O | AC2O | AC1O | AC0O | page 265 |
| 0x2F (0x4F) | Reserved | – | – | – | – | – | – | – | – | |
| 0x2E (0x4E) | SPDR | SPD7 | SPD6 | SPD5 | SPD4 | SPD3 | SPD2 | SPD1 | SPD0 | page 165 |
| 0x2D (0x4D) | SPSR | SPIF | WCOL | – | – | – | – | – | SPI2X | page 164 |
| 0x2C (0x4C) | SPCR | SPIE | SPE | DORD | MSTR | CPOL | CPHA | SPR1 | SPR0 | page 163 |
| 0x2B (0x4B) | Reserved | – | – | – | – | – | – | – | – | |
| 0x2A (0x4A) | Reserved | – | – | – | – | – | – | – | – | |
| 0x29 (0x49) | PLLCSR | – | – | – | – | – | PLLF | PLLE | PLOCK | page 36 |
| 0x28 (0x48) | OCR0B | OCR0B7 | OCR0B6 | OCR0B5 | OCR0B4 | OCR0B3 | OCR0B2 | OCR0B1 | OCR0B0 | page 105 |
| 0x27 (0x47) | OCR0A | OCR0A7 | OCR0A6 | OCR0A5 | OCR0A4 | OCR0A3 | OCR0A2 | OCR0A1 | OCR0A0 | page 105 |
| 0x26 (0x46) | TCNT0 | TCNT07 | TCNT06 | TCNT05 | TCNT04 | TCNT03 | TCNT02 | TCNT01 | TCNT00 | page 105 |
| 0x25 (0x45) | TCCR0B | FOCOA | FOC0B | – | – | WGM02 | CS02 | CS01 | CS00 | page 103 |
| 0x24 (0x44) | TCCR0A | COM0A1 | COM0A0 | COM0B1 | COM0B0 | – | – | WGM01 | WGM00 | page 101 |
| 0x23 (0x43) | GTCCR | TSM | ICPSEL1 | – | – | – | – | – | PSRSYNC | page 88 |
| 0x22 (0x42) | EEARH | – | – | – | – | – | – | EEAR9 | EEAR8 | page 23 |
| 0x21 (0x41) | EEARL | EEAR7 | EEAR6 | EEAR5 | EEAR4 | EEAR3 | EEAR2 | EEAR1 | EEAR0 | page 23 |
| 0x20 (0x40) | EEDR | EEDR7 | EEDR6 | EEDR5 | EEDR4 | EEDR3 | EEDR2 | EEDR1 | EEDR0 | page 23 |
| 0x1F (0x3F) | EEDR | – | – | – | – | – | EERIE | EEMWE | EERE | page 23 |
| 0x1E (0x3E) | GPOR0 | GPOR07 | GPOR06 | GPOR05 | GPOR04 | GPOR03 | GPOR02 | GPOR01 | GPOR00 | page 28 |
| 0x1D (0x3D) | EIMSK | – | – | – | – | INT3 | INT2 | INT1 | INT0 | page 83 |
| 0x1C (0x3C) | EIFR | – | – | – | – | INTF3 | INTF2 | INTF1 | INTF0 | page 84 |
| 0x1B (0x3B) | PCIFR | – | – | – | – | PCIF3 | PCIF2 | PCIF1 | PCIF0 | page 85 |



| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Page |
|-------------|-----------------|---------|---------|---------|---------|---------|---------|---------|---------|--------------------------|
| 0x1A (0x3A) | GPIOR2 | GPIOR27 | GPIOR26 | GPIOR25 | GPIOR24 | GPIOR23 | GPIOR22 | GPIOR21 | GPIOR20 | page 28 |
| 0x19 (0x39) | GPIOR1 | GPIOR17 | GPIOR16 | GPIOR15 | GPIOR14 | GPIOR13 | GPIOR12 | GPIOR11 | GPIOR10 | page 28 |
| 0x18 (0x38) | Reserved | – | – | – | – | – | – | – | – | |
| 0x17 (0x37) | Reserved | – | – | – | – | – | – | – | – | |
| 0x16 (0x36) | TIFR1 | – | – | ICF1 | – | – | OCF1B | OCF1A | TOV1 | page 134 |
| 0x15 (0x35) | TIFR0 | – | – | – | – | – | OCF0B | OCF0A | TOV0 | page 106 |
| 0x14 (0x34) | Reserved | – | – | – | – | – | – | – | – | |
| 0x13 (0x33) | Reserved | – | – | – | – | – | – | – | – | |
| 0x12 (0x32) | Reserved | – | – | – | – | – | – | – | – | |
| 0x11 (0x31) | Reserved | – | – | – | – | – | – | – | – | |
| 0x10 (0x30) | Reserved | – | – | – | – | – | – | – | – | |
| 0x0F (0x2F) | Reserved | – | – | – | – | – | – | – | – | |
| 0x0E (0x2E) | PORTE | – | – | – | – | – | PORTE2 | PORTE1 | PORTE0 | page 81 |
| 0x0D (0x2D) | DDRE | – | – | – | – | – | DDE2 | DDE1 | DDE0 | page 81 |
| 0x0C (0x2C) | PINE | – | – | – | – | – | PINE2 | PINE1 | PINE0 | page 81 |
| 0x0B (0x2B) | PORTD | PORTD7 | PORTD6 | PORTD5 | PORTD4 | PORTD3 | PORTD2 | PORTD1 | PORTD0 | page 80 |
| 0x0A (0x2A) | DDRD | DDD7 | DDD6 | DDD5 | DDD4 | DDD3 | DDD2 | DDD1 | DDD0 | page 80 |
| 0x09 (0x29) | PIND | PIND7 | PIND6 | PIND5 | PIND4 | PIND3 | PIND2 | PIND1 | PIND0 | page 81 |
| 0x08 (0x28) | PORTC | PORTC7 | PORTC6 | PORTC5 | PORTC4 | PORTC3 | PORTC2 | PORTC1 | PORTC0 | page 80 |
| 0x07 (0x27) | DDRC | DDC7 | DDC6 | DDC5 | DDC4 | DDC3 | DDC2 | DDC1 | DDC0 | page 80 |
| 0x06 (0x26) | PINC | PINC7 | PINC6 | PINC5 | PINC4 | PINC3 | PINC2 | PINC1 | PINC0 | page 80 |
| 0x05 (0x25) | PORTB | PORTB7 | PORTB6 | PORTB5 | PORTB4 | PORTB3 | PORTB2 | PORTB1 | PORTB0 | page 80 |
| 0x04 (0x24) | DDRB | DDB7 | DDB6 | DDB5 | DDB4 | DDB3 | DDB2 | DDB1 | DDB0 | page 80 |
| 0x03 (0x23) | PINB | PINB7 | PINB6 | PINB5 | PINB4 | PINB3 | PINB2 | PINB1 | PINB0 | page 80 |
| 0x02 (0x22) | Reserved | – | – | – | – | – | – | – | – | |
| 0x01 (0x21) | Reserved | – | – | – | – | – | – | – | – | |
| 0x00 (0x20) | Reserved | – | – | – | – | – | – | – | – | |

- Note:
- For compatibility with future devices, reserved bits should be written to zero if accessed. Reserved I/O memory addresses should never be written.
 - I/O Registers within the address range 0x00 - 0x1F 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.
 - Some of the status flags are cleared by writing a logical one to them. Note that, unlike most other AVR's, the CBI and SBI instructions will only operate on the specified bit, and can therefore be used on registers containing such status flags. The CBI and SBI instructions work with registers 0x00 to 0x1F only.
 - When using the I/O specific commands IN and OUT, the I/O addresses 0x00 - 0x3F must be used. When addressing I/O Registers as data space using LD and ST instructions, 0x20 must be added to these addresses. The ATmega16/32/64/M1/C1 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 0x60 - 0xFF in SRAM, only the ST/STS/STD and LD/LDS/LDD instructions can be used.
 - These registers are only available on ATmega32/64M1. For other products described in this datasheet, these locations are reserved.

4. 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 | $PC \leftarrow STACK$ | None | 4 |
| RETI | | Interrupt Return | $PC \leftarrow STACK$ | 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 ⊕ V = 0) then $PC \leftarrow PC + k + 1$ | None | 1/2 |
| BRLT | k | Branch if Less Than Zero, Signed | if (N ⊕ 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 |

| 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 |
| OUT | P, Rr | Out Port | P ← Rr | None | 1 |
| PUSH | Rr | Push Register on Stack | STACK ← Rr | None | 2 |

| Mnemonics | Operands | Description | Operation | Flags | #Clocks |
|---------------------------------|----------|-------------------------|--|-------|---------|
| POP | Rd | Pop Register from Stack | Rd ← 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 |

Note: 1. These Instructions are only available in “16K and 32K parts”

5. Errata

5.1 Errata Summary

5.1.1 ATmega32M1/C1 Rev. C (Mask Revision)

- The AMPCMPx bits return 0

5.1.2 ATmega32M1/C1 Rev. B (Mask Revision)

- The AMPCMPx bits return 0
- No comparison when amplifier is used as comparator input and ADC input
- CRC calculation of diagnostic frames in LIN 2.x.
- Wrong TSOFFSET manufacturing calibration value
- PD0-PD3 set to outputs and PD4 pulled down following power-on with external reset active.

5.1.3 ATmega32M1/C1 Rev. A (Mask Revision)

- Inopportune reset of the CANIDM registers.
- The AMPCMPx bits return 0
- No comparison when amplifier is used as comparator input and ADC input
- CRC calculation of diagnostic frames in LIN 2.x.
- PD0-PD3 set to outputs and PD4 pulled down following power-on with external reset active.

5.1.4 Errata Description

1. Inopportune reset of the CANIDM registers

After the reception of a CAN frame in a MOB, the ID mask registers are reset.

Problem fix / workaround

Before enabling a MOB in reception, re-initialize the ID mask registers - **CANIDM[4..1]**.

2. The AMPCMPx bits return 0

When they are read the AMPCMPx bits in AMPxCSR registers return 0.

Problem fix / workaround

If the reading of the AMPCMPx bits is required, store the AMPCMPx value in a variable in memory before writing in the AMPxCSR register and read the variable when necessary.

3. No comparison when amplifier is used as comparator input and ADC input

When it is selected as ADC input, an amplifier receives no clock signal when the ADC is stopped. In that case, if the amplifier is also used as comparator input, no analog signal is propagated and no comparison is done.

Problem fix / workaround

Select another ADC channel rather than the working amplified channel.

4. CRC calculation of diagnostic frames in LIN 2.x.

Diagnostic frames of LIN 2.x use “classic checksum” calculation. Unfortunately, the setting of the checksum model is enabled when the HEADER is transmitted/received. Usually, in LIN 2.x the LIN/UART controller is initialized to process “enhanced checksums” and a slave task does not know what kind of frame it will work on before checking the ID.

Problem fix / workaround

This workaround is to be implemented only in case of transmission/reception of diagnostics frames.

- a. Slave task of master node:
Before enabling the HEADER, the master must set the appropriate LIN13 bit value in LINCR register.

- b. For slaves nodes, the workaround is in 2 parts:

– Before enabling the RESPONSE, use the following function:

```
void lin_wa_head(void) {
    unsigned char temp;
    temp = LINBTR;
    LINCR = 0x00; // It is not a RESET !
    LINBTR = (1<<LDISR) | temp;
    LINCR = (1<<LIN13) | (1<<LENA) | (0<<LCMD2) | (0<<LCMD1) | (0<<LCMD0);
    LINDLR = 0x88; // If it isn't already done
}
```

– Once the RESPONSE is received or sent (having RxOK or TxOK as well as LERR), use the following function:

```
void lin_wa_tail(void) {
    LINCR = 0x00; // It is not a RESET !
    LINBTR = 0x00;
    LINCR = (0<<LIN13) | (1<<LENA) | (0<<LCMD2) | (0<<LCMD1) | (0<<LCMD0);
}
```

The time-out counter is disabled during the RESPONSE when the workaround is set.

5. Wrong TSOFFSET manufacturing calibration value.

Erroneous value of TSOFFSET programmed in signature byte.
(TSOFFSET was introduced from REVB silicon).

Problem fix / workaround

To identify RevB with wrong TSOFFSET value, check device signature byte at address 0X3F if value is not 0X42 (Ascii code 'B') then use the following formula.

$TS_OFFSET(True) = (150 * (1 - TS_GAIN)) + TS_OFFSET.$

6. PD0-PD3 set to outputs and PD4 pulled down following power-on with external reset active.

At power-on with the external reset signal active the four I/O lines PD0-PD3 may be forced into an output state. Normally these lines should be in an input state. PD4 may be pulled down with internal 220 kOhm resistor. Following release of the reset line (whatever is the startup time) with the clock running the I/Os PD0-PD4 will adopt their intended input state.

Problem fix / workaround

None

6. Ordering Information

Figure 6-1. ATmega32M1 engineering samples delivery only. Automotive qualification not yet fully completed.

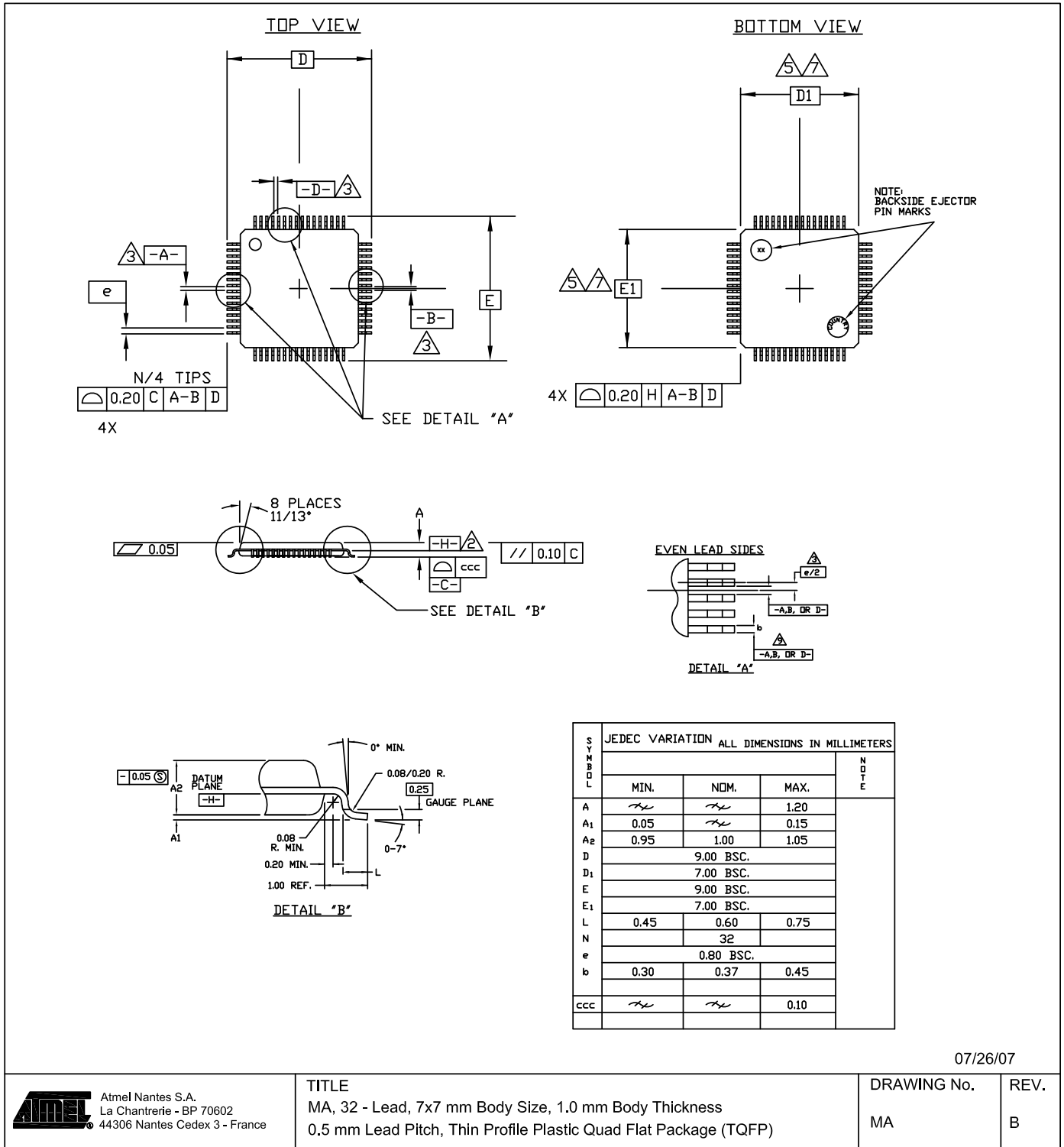
| Memory Size | PSC | Power Supply | Ordering Code | Package | Operation Range |
|-------------|-----|--------------|---------------|---------|---------------------|
| 32K | No | 2.7 - 5.5V | MEGA32C1-15AZ | MA | -40-C to 125-C |
| 32K | No | 2.7 - 5.5V | MEGA32C1-15MZ | PV | -40-C to 125-C |
| 32K | No | 2.7 - 5.5V | MEGA32C1-ESAZ | MA | Engineering Samples |
| 32K | No | 2.7 - 5.5V | MEGA32C1-ESMZ | PV | Engineering Samples |
| | | | | | |
| 32K | Yes | 2.7 - 5.5V | MEGA32M1-15AZ | MA | -40-C to 125-C |
| 32K | Yes | 2.7 - 5.5V | MEGA32M1-15MZ | PV | -40-C to 125-C |
| 32K | Yes | 2.7 - 5.5V | MEGA32M1-ESAZ | MA | Engineering Samples |
| 32K | Yes | 2.7 - 5.5V | MEGA32M1-ESMZ | PV | Engineering Samples |

Note: All packages are Pb free, fully LHF

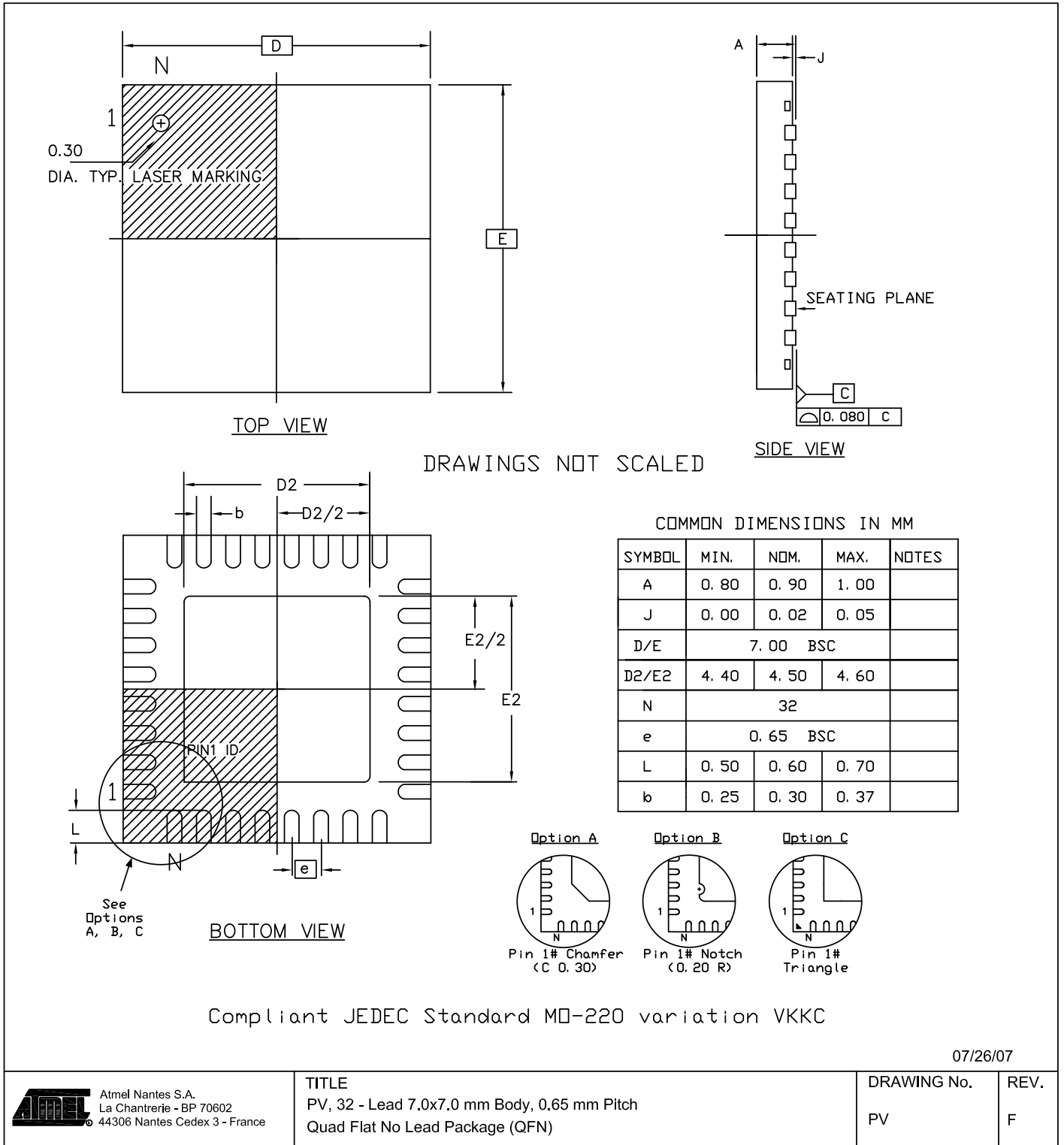
7. Package Information

| Package Type | |
|---------------------|--|
| MA | MA, 32 - Lead, 7x7 mm Body Size, 1.0 mm Body Thickness 0.5 mm Lead Pitch, Thin Profile Plastic Quad Flat Package (TQFP) |
| PV | PV, 32-Lead, 5.0x5.0 mm Body, 0.50 mm Pitch Quad Flat No Lead Package (QFN) |

7.1 TQFP32



7.2 QFN32





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

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 ATMEL'S TERMS AND CONDITIONS OF SALE LOCATED ON ATMEL'S WEB SITE, 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 OF 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 product 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's products are not intended, authorized, or warranted for use as components in applications intended to support or sustain life.

© 2008 Atmel Corporation. All rights reserved. Atmel®, logo and combinations thereof, AVR® and others are registered trademarks or trademarks of Atmel Corporation or its subsidiaries. Other terms and product names may be trademarks of others.