

## Features

- High Performance, Low Power Atmel® AVR® 8-Bit Microcontroller
- Advanced RISC Architecture
  - 130 Powerful Instructions – Most Single Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 20MIPS Throughput at 20MHz
  - On-Chip 2-cycle Multiplier
- High Endurance Non-volatile Memory segments
  - In-System Self-programmable Flash Program Memory
    - 32KBytes
  - EEPROM
    - 1Kbytes
  - Internal SRAM
    - 2Kbytes
  - Write/Erase cycles: 10,000 Flash/100,000 EEPROM
  - Data retention: 20 years at 85°C/100 years at 25°C<sup>(1)</sup>
  - Optional Boot Code Section with Independent Lock Bits
    - In-System Programming by On-chip Boot Program
    - True Read-While-Write Operation
  - 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
- Peripheral Features
  - 4 x 25 Segment LCD Driver (ATmega329P)
  - 4 x 40 Segment LCD Driver (ATmega3290P)
  - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
  - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
  - Real Time Counter with Separate Oscillator
  - Four PWM Channels
  - 8-channel, 10-bit ADC
  - Programmable Serial USART
  - Master/Slave SPI Serial Interface
  - Universal Serial Interface with Start Condition Detector
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - On-chip Analog Comparator
  - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated Oscillator
  - External and Internal Interrupt Sources
  - Five Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby
- I/O and Packages
  - 54/69 Programmable I/O Lines
  - 64/100-lead TQFP, 64-pad QFN/MLF
- Speed Grade:
  - ATmega329P/ATmega3290P:
    - 0 - 16MHz @ 1.8 - 5.5V,
    - 0 - 20MHz @ 2.7 - 5.5V
- Temperature range:
  - -40°C to 85°C Industrial
- Ultra-Low Power Consumption
  - Active Mode:
    - 420µA at 1MHz, 1.8V
  - Power-down Mode:
    - 40nA at 1.8V
  - Power-save Mode:
    - 750nA at 1.8V



## 8-bit Atmel Microcontroller with 32KBytes In-System Programmable Flash

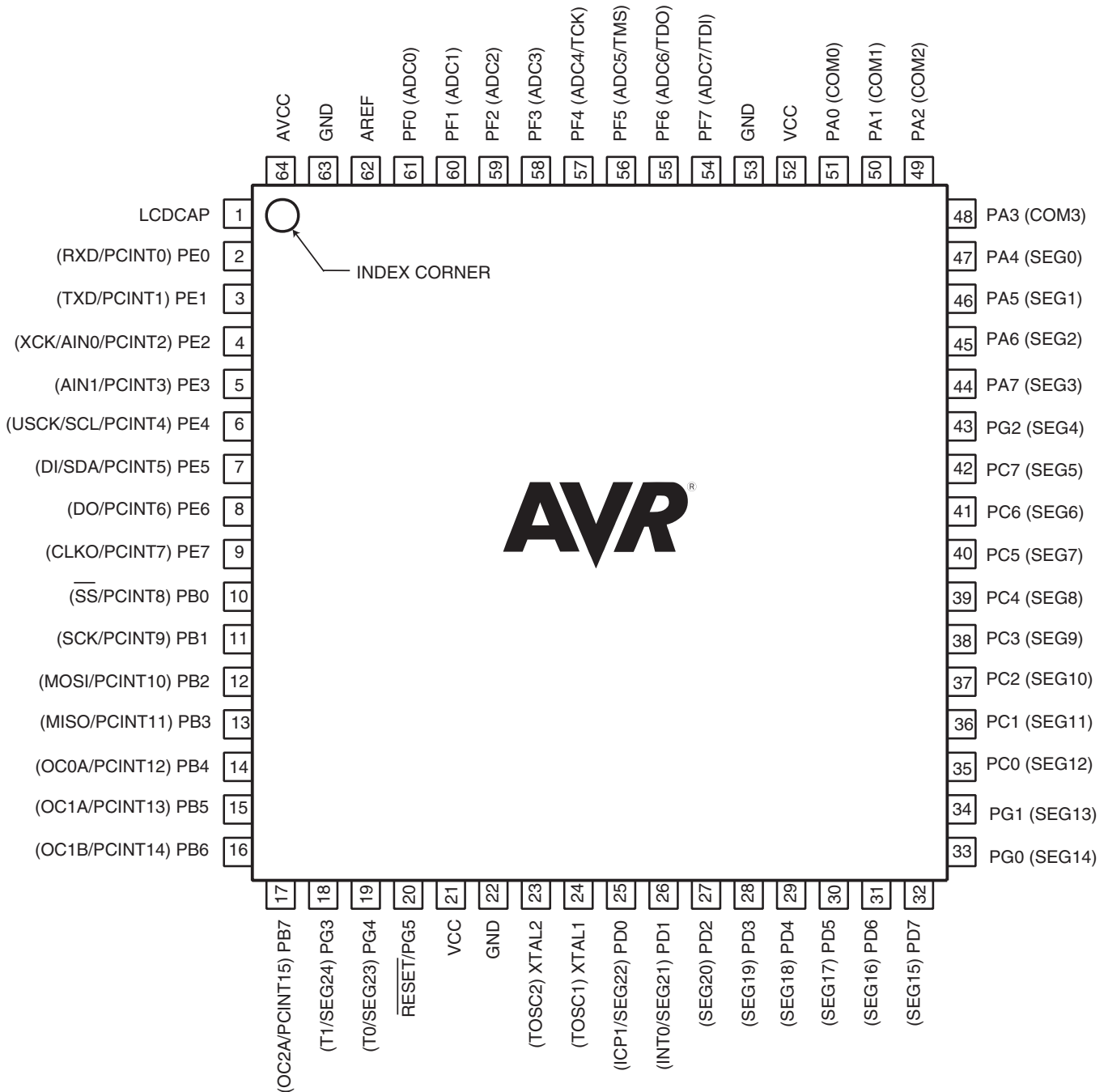
**ATmega329P**  
**ATmega3290P**

## Preliminary Summary



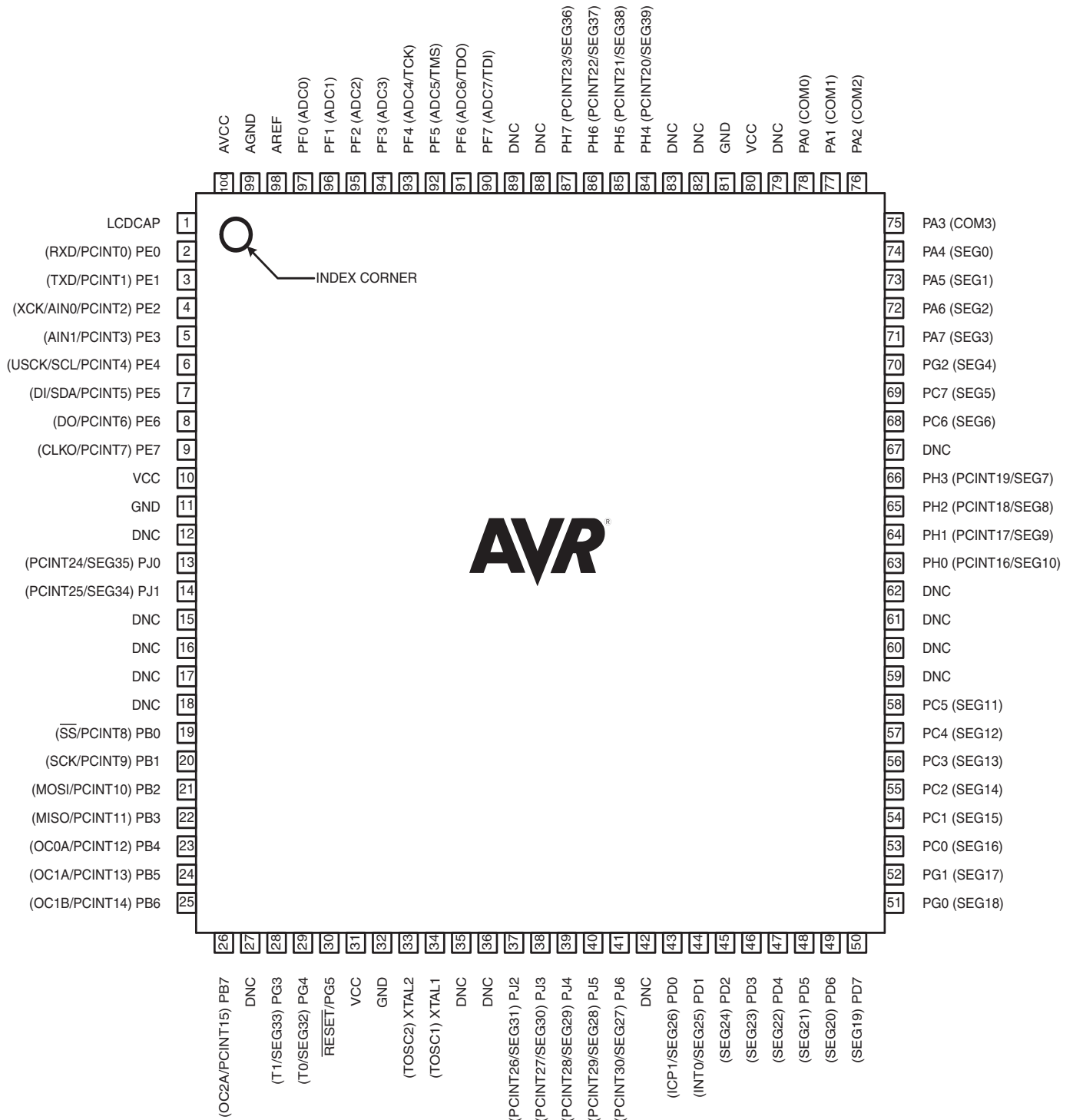
## 1. Pin Configurations

Figure 1-1. MLF/ Pinout ATmega329P



Note: The large center pad underneath the QFN/MLF packages is made of metal and internally connected to GND. It should be soldered or glued to the board to ensure good mechanical stability. If the center pad is left unconnected, the package might loosen from the board.

Figure 1-2. TQFP / Pinout ATmega3290P

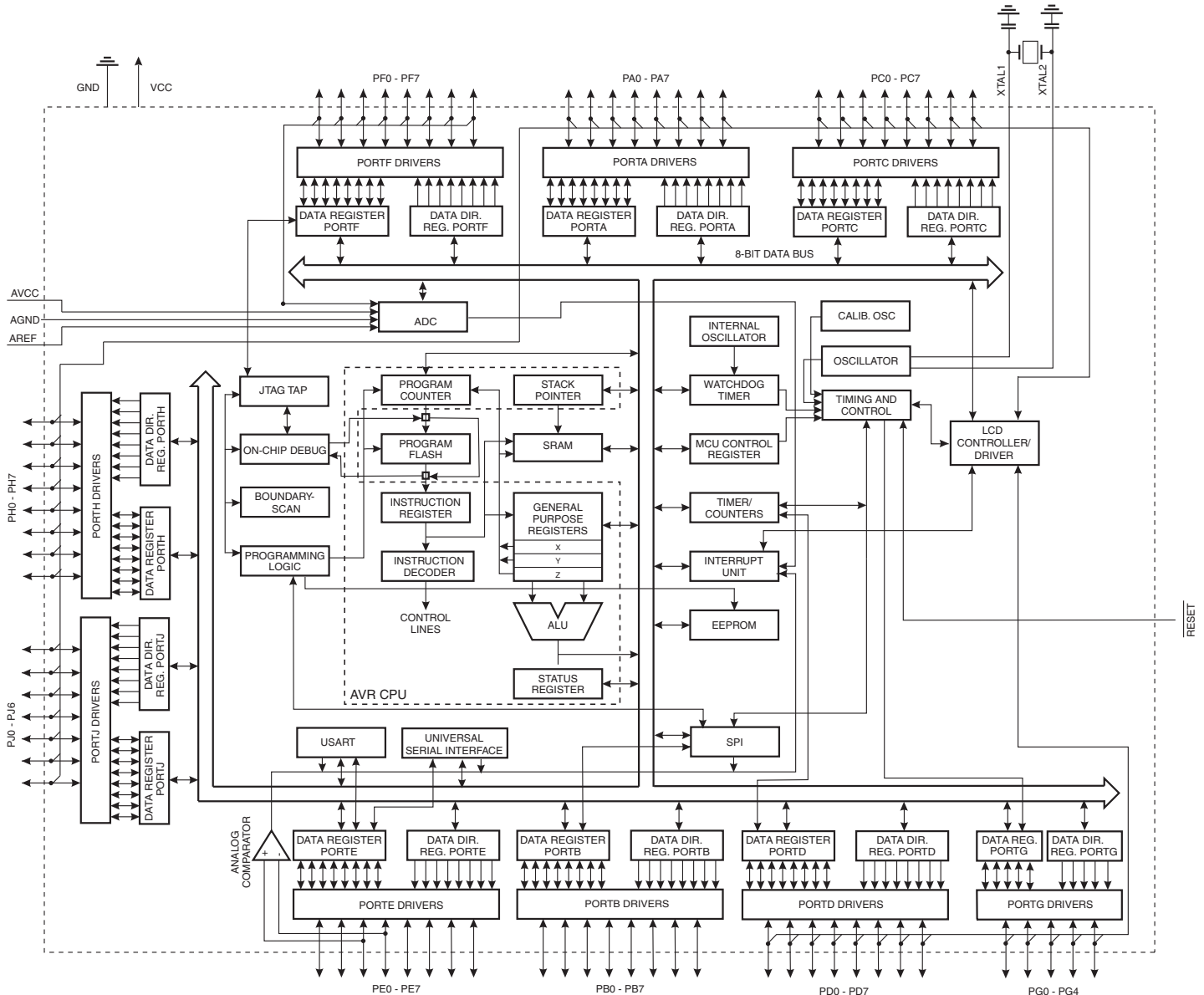


## 2. Overview

The ATmega329P/3290P 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 ATmega329P/3290P achieves throughputs approaching 1MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

### 2.1 Block Diagram

Figure 2-1. Block Diagram



The Atmel® 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 ATmel®AVR® ATmega329P/3290P provides the following features: 32K bytes of In-System Programmable Flash with Read-While-Write capabilities, 1Kbytes EEPROM, 2Kbyte SRAM, 54/69 general purpose I/O lines, 32 general purpose working registers, a JTAG interface for Boundary-scan, On-chip Debugging support and programming, a complete On-chip LCD controller with internal contrast control, three flexible Timer/Counters with compare modes, internal and external interrupts, a serial programmable USART, Universal Serial Interface with Start Condition Detector, an 8-channel, 10-bit ADC, a programmable Watchdog Timer with internal Oscillator, an SPI serial port, and five 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. In Power-save mode, the asynchronous timer and the LCD controller continues to run, allowing the user to maintain a timer base and operate the LCD display while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except asynchronous timer, LCD controller and 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 non-volatile memory technology. The On-chip In-System re-Programmable (ISP) Flash allows the program memory to be re-programmed In-System through an SPI serial interface, by a conventional non-volatile memory programmer, or by an On-chip Boot program running on the AVR core. The Boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash section will continue to run while the Application Flash section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega329P/3290P is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications.

The ATmega329P/3290P 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 Comparison between ATmega329P, and ATmega3290P.

The ATmega329P, and ATmega3290P differ only in pin count and pinout. [Table 2-1 on page 5](#) summarizes the different configurations for the four devices.

**Table 2-1.** Configuration Summary

| Device      | Flash    | EEPROM  | RAM     | LCD Segments | General Purpose I/O Pins |
|-------------|----------|---------|---------|--------------|--------------------------|
| ATmega329P  | 32Kbytes | 1Kbytes | 2Kbytes | 4 x 25       | 54                       |
| ATmega3290P | 32Kbytes | 1Kbytes | 2Kbytes | 4 x 40       | 69                       |

## 2.3 Pin Descriptions

The following section describes the I/O-pin special functions.

### 2.3.1 $V_{CC}$

Digital supply voltage.

### 2.3.2 GND

Ground.

### 2.3.3 Port A (PA7...PA0)

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

Port A also serves the functions of various special features of the ATmega329P/3290P as listed on [page 71](#).

### 2.3.4 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 ATmega329P/3290P as listed on [page 72](#).

### 2.3.5 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 ATmega329P/3290P as listed on [page 75](#).

### 2.3.6 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 ATmega329P/3290P as listed on [page 76](#).

## 2.3.7 Port E (PE7...PE0)

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.

Port E also serves the functions of various special features of the ATmega329P/3290P as listed on [page 78](#).

## 2.3.8 Port F (PF7...PF0)

Port F serves as the 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 is 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. 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.

Port F also serves the functions of the JTAG interface.

## 2.3.9 Port G (PG5...PG0)

Port G is a 6-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port G output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port G pins that are externally pulled low will source current if the pull-up resistors are activated. The Port G pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port G also serves the functions of various special features of the ATmega329P/3290P as listed on [page 82](#).

## 2.3.10 Port H (PH7...PH0)

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

Port H also serves the functions of various special features of the ATmega3290P as listed on [page 84](#).

## 2.3.11 Port J (PJ6...PJ0)

Port J is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port J output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port J pins that are externally pulled low will source current if the pull-up resistors are activated. The Port J pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port J also serves the functions of various special features of the ATmega3290P as listed on [page 87](#).

## 2.3.12 **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 "System and Reset Characteristics" on page 336. Shorter pulses are not guaranteed to generate a reset.

## 2.3.13 **XTAL1**

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

## 2.3.14 **XTAL2**

Output from the inverting Oscillator amplifier.

## 2.3.15 **AVCC**

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

## 2.3.16 **AREF**

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

## 2.3.17 **LCDCAP**

An external capacitor (typical > 470nF) must be connected to the LCDCAP pin as shown in [Figure 23-2](#), if the LCD module is enabled and configured to use internal power. This capacitor acts as a reservoir for LCD power ( $V_{LCD}$ ). A large capacitance reduces ripple on  $V_{LCD}$  but increases the time until  $V_{LCD}$  reaches its target value.

### 3. Resources

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

### 4. Data Retention

Reliability Qualification results show that the projected data retention failure rate is much less than 1 PPM over 20 years at 85°C or 100 years at 25°C.

### 5. About Code Examples

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

For I/O Registers located in extended I/O map, “IN”, “OUT”, “SBIS”, “SBIC”, “CBI”, and “SBI” instructions must be replaced with instructions that allow access to extended I/O. Typically “LDS” and “STS” combined with “SBR”, “SBRC”, “SBR”, and “CBR”.

## 6. Register Summary

Note: Registers with bold type only available in ATmega3290P.

| Address | Name           | Bit 7                          | Bit 6         | Bit 5         | Bit 4         | Bit 3         | Bit 2         | Bit 1         | Bit 0         | Page |
|---------|----------------|--------------------------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|------|
| (0xFF)  | <b>LCDDR19</b> | <b>SEG339</b>                  | <b>SEG338</b> | <b>SEG337</b> | <b>SEG336</b> | <b>SEG335</b> | <b>SEG334</b> | <b>SEG333</b> | <b>SEG332</b> | 247  |
| (0xFE)  | LCDDR18        | <b>SEG331</b>                  | <b>SEG330</b> | <b>SEG329</b> | <b>SEG328</b> | <b>SEG327</b> | <b>SEG326</b> | <b>SEG325</b> | <b>SEG324</b> | 247  |
| (0xFD)  | LCDDR17        | SEG323                         | SEG322        | SEG321        | SEG320        | SEG319        | SEG318        | SEG317        | SEG316        | 247  |
| (0xFC)  | LCDDR16        | SEG315                         | SEG314        | SEG313        | SEG312        | SEG311        | SEG310        | SEG309        | SEG308        | 247  |
| (0xFB)  | LCDDR15        | SEG307                         | SEG306        | SEG305        | SEG304        | SEG303        | SEG302        | SEG301        | SEG300        | 247  |
| (0xFA)  | <b>LCDDR14</b> | <b>SEG239</b>                  | <b>SEG238</b> | <b>SEG237</b> | <b>SEG236</b> | <b>SEG235</b> | <b>SEG234</b> | <b>SEG233</b> | <b>SEG232</b> | 247  |
| (0xF9)  | LCDDR13        | <b>SEG231</b>                  | <b>SEG230</b> | <b>SEG229</b> | <b>SEG228</b> | <b>SEG227</b> | <b>SEG226</b> | <b>SEG225</b> | SEG224        | 247  |
| (0xF8)  | LCDDR12        | SEG223                         | SEG222        | SEG221        | SEG220        | SEG219        | SEG218        | SEG217        | SEG216        | 247  |
| (0xF7)  | LCDDR11        | SEG215                         | SEG214        | SEG213        | SEG212        | SEG211        | SEG210        | SEG209        | SEG208        | 247  |
| (0xF6)  | LCDDR10        | SEG207                         | SEG206        | SEG205        | SEG204        | SEG203        | SEG202        | SEG201        | SEG200        | 247  |
| (0xF5)  | <b>LCDDR09</b> | <b>SEG139</b>                  | <b>SEG138</b> | <b>SEG137</b> | <b>SEG136</b> | <b>SEG135</b> | <b>SEG134</b> | <b>SEG133</b> | <b>SEG132</b> | 247  |
| (0xF4)  | LCDDR08        | <b>SEG131</b>                  | <b>SEG130</b> | <b>SEG129</b> | <b>SEG128</b> | <b>SEG127</b> | <b>SEG126</b> | <b>SEG125</b> | SEG124        | 247  |
| (0xF3)  | LCDDR07        | SEG123                         | SEG122        | SEG121        | SEG120        | SEG119        | SEG118        | SEG117        | SEG116        | 247  |
| (0xF2)  | LCDDR06        | SEG115                         | SEG114        | SEG113        | SEG112        | SEG111        | SEG110        | SEG109        | SEG108        | 247  |
| (0xF1)  | LCDDR05        | SEG107                         | SEG106        | SEG105        | SEG104        | SEG103        | SEG102        | SEG101        | SEG100        | 247  |
| (0xF0)  | <b>LCDDR04</b> | <b>SEG039</b>                  | <b>SEG038</b> | <b>SEG037</b> | <b>SEG036</b> | <b>SEG035</b> | <b>SEG034</b> | <b>SEG033</b> | <b>SEG032</b> | 247  |
| (0xEF)  | LCDDR03        | <b>SEG031</b>                  | <b>SEG030</b> | <b>SEG029</b> | <b>SEG028</b> | <b>SEG027</b> | <b>SEG026</b> | <b>SEG025</b> | SEG024        | 247  |
| (0xEE)  | LCDDR02        | SEG023                         | SEG022        | SEG021        | SEG020        | SEG019        | SEG018        | SEG017        | SEG016        | 247  |
| (0xED)  | LCDDR01        | SEG015                         | SEG014        | SEG013        | SEG012        | SEG011        | SEG010        | SEG009        | SEG008        | 247  |
| (0xEC)  | LCDDR00        | SEG007                         | SEG006        | SEG005        | SEG004        | SEG003        | SEG002        | SEG001        | SEG000        | 247  |
| (0xEB)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xEA)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xE9)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xE8)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xE7)  | LCDCCR         | LCDDC2                         | LCDDC1        | LCDDC0        | LCDMDT        | LCDC3         | LCDC2         | LCDC1         | LCDC0         | 245  |
| (0xE6)  | LCDFRR         | -                              | LCDPS2        | LCDPS1        | LCDPS0        | -             | LCDCD2        | LCDCD1        | LCDCD0        | 243  |
| (0xE5)  | LCDCRB         | LCDCS                          | LCDB2B        | LCDMUX1       | LCDMUX0       | <b>LCDPM3</b> | LCDPM2        | LCDPM1        | LCDPM0        | 242  |
| (0xE4)  | LCDCRA         | LCDEN                          | LCDAB         | -             | LCDIF         | LCDIE         | LCDBD         | LCDCCD        | LCDBL         | 241  |
| (0xE3)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xE2)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xE1)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xE0)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xDF)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xDE)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xDD)  | <b>PORTJ</b>   | -                              | <b>PORTJ6</b> | <b>PORTJ5</b> | <b>PORTJ4</b> | <b>PORTJ3</b> | <b>PORTJ2</b> | <b>PORTJ1</b> | <b>PORTJ0</b> | 93   |
| (0xDC)  | <b>DDRJ</b>    | -                              | <b>DDJ6</b>   | <b>DDJ5</b>   | <b>DDJ4</b>   | <b>DDJ3</b>   | <b>DDJ2</b>   | <b>DDJ1</b>   | <b>DDJ0</b>   | 93   |
| (0xDB)  | <b>PINJ</b>    | -                              | <b>PINJ6</b>  | <b>PINJ5</b>  | <b>PINJ4</b>  | <b>PINJ3</b>  | <b>PINJ2</b>  | <b>PINJ1</b>  | <b>PINJ0</b>  | 93   |
| (0xDA)  | <b>PORTH</b>   | <b>PORTH7</b>                  | <b>PORTH6</b> | <b>PORTH5</b> | <b>PORTH4</b> | <b>PORTH3</b> | <b>PORTH2</b> | <b>PORTH1</b> | <b>PORTH0</b> | 92   |
| (0xD9)  | <b>DDRH</b>    | <b>DDH7</b>                    | <b>DDH6</b>   | <b>DDH5</b>   | <b>DDH4</b>   | <b>DDH3</b>   | <b>DDH2</b>   | <b>DDH1</b>   | <b>DDH0</b>   | 93   |
| (0xD8)  | <b>PINH</b>    | <b>PINH7</b>                   | <b>PINH6</b>  | <b>PINH5</b>  | <b>PINH4</b>  | <b>PINH3</b>  | <b>PINH2</b>  | <b>PINH1</b>  | <b>PINH0</b>  | 93   |
| (0xD7)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xD6)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xD5)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xD4)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xD3)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xD2)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xD1)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xD0)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xCF)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xCE)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xCD)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xCC)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xCB)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xCA)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xC9)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xC8)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xC7)  | Reserved       | -                              | -             | -             | -             | -             | -             | -             | -             |      |
| (0xC6)  | UDR0           | USART0 Data Register           |               |               |               |               |               |               |               | 189  |
| (0xC5)  | UBRR0H         | USART0 Baud Rate Register High |               |               |               |               |               |               |               | 193  |



| Address | Name     | Bit 7   | Bit 6  | Bit 5  | Bit 4  | Bit 3   | Bit 2   | Bit 1   | Bit 0   | Page |
|---------|----------|---|--------|--------|--------|---------|---------|---------|---------|------|
| (0xC4)  | UBRR0L   | USART0 Baud Rate Register Low                 |        |        |        |         |         |         |         | 193  |
| (0xC3)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xC2)  | UCSR0C   | -   | UMSEL0 | UPM01  | UPM00  | USBS0   | UCSZ01  | UCSZ00  | UCPOL0  | 191  |
| (0xC1)  | UCSR0B   | RXCIE0  | TXCIE0 | UDRIE0 | RXEN0  | TXEN0   | UCSZ02  | RXB80   | TXB80   | 190  |
| (0xC0)  | UCSR0A   | RXC0  | TXC0   | UDRE0  | FE0    | DOR0    | UPE0    | U2X0    | MPCM0   | 189  |
| (0xBF)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xBE)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xBD)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xBC)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xBB)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xBA)  | USIDR    | USI Data Register                             |        |        |        |         |         |         |         | 206  |
| (0xB9)  | USISR    | USISIF  | USIOIF | USIPF  | USIDC  | USICNT3 | USICNT2 | USICNT1 | USICNT0 | 206  |
| (0xB8)  | USICR    | USISIE  | USIOIE | USIWM1 | USIWM0 | USICS1  | USICS0  | USICLK  | USITC   | 207  |
| (0xB7)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xB6)  | ASSR     | -   | -      | -      | EXCLK  | AS2     | TCN2UB  | OCR2UB  | TCR2UB  | 157  |
| (0xB5)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xB4)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xB3)  | OCR2A    | Timer/Counter 2 Output Compare Register A     |        |        |        |         |         |         |         | 157  |
| (0xB2)  | TCNT2    | Timer/Counter2                                |        |        |        |         |         |         |         | 157  |
| (0xB1)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xB0)  | TCCR2A   | FOC2A   | WGM20  | COM2A1 | COM2A0 | WGM21   | CS22    | CS21    | CS20    | 155  |
| (0xAF)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xAE)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xAD)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xAC)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xAB)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xAA)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xA9)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xA8)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xA7)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xA6)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xA5)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xA4)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xA3)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xA2)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xA1)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0xA0)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x9F)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x9E)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x9D)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x9C)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x9B)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x9A)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x99)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x98)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x97)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x96)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x95)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x94)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x93)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x92)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x91)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x90)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x8F)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x8E)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x8D)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x8C)  | Reserved | -   | -      | -      | -      | -       | -       | -       | -       |      |
| (0x8B)  | OCR1BH   | Timer/Counter1 Output Compare Register B High |        |        |        |         |         |         |         | 131  |
| (0x8A)  | OCR1BL   | Timer/Counter1 Output Compare Register B Low  |        |        |        |         |         |         |         | 131  |
| (0x89)  | OCR1AH   | Timer/Counter1 Output Compare Register A High |        |        |        |         |         |         |         | 131  |
| (0x88)  | OCR1AL   | Timer/Counter1 Output Compare Register A Low  |        |        |        |         |         |         |         | 131  |
| (0x87)  | ICR1H    | Timer/Counter1 Input Capture Register High    |        |        |        |         |         |         |         | 132  |
| (0x86)  | ICR1L    | Timer/Counter1 Input Capture Register Low     |        |        |        |         |         |         |         | 132  |

| Address     | Name     | Bit 7                                      | Bit 6   | Bit 5   | Bit 4   | Bit 3   | Bit 2   | Bit 1    | Bit 0   | Page      |
|-------------|----------|--|---------|---------|---------|---------|---------|----------|---------|-----------|
| (0x85)      | TCNT1H   | Timer/Counter1 High                        |         |         |         |         |         |          |         | 131       |
| (0x84)      | TCNT1L   | Timer/Counter1 Low                         |         |         |         |         |         |          |         | 131       |
| (0x83)      | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| (0x82)      | TCCR1C   | FOC1A                                      | FOC1B   | -       | -       | -       | -       | -        | -       | 130       |
| (0x81)      | TCCR1B   | ICNC1                                      | ICES1   | -       | WGM13   | WGM12   | CS12    | CS11     | CS10    | 129       |
| (0x80)      | TCCR1A   | COM1A1                                     | COM1A0  | COM1B1  | COM1B0  | -       | -       | WGM11    | WGM10   | 127       |
| (0x7F)      | DIDR1    | -  | -       | -       | -       | -       | -       | AIN1D    | AIN0D   | 212       |
| (0x7E)      | DIDR0    | ADC7D                                      | ADC6D   | ADC5D   | ADC4D   | ADC3D   | ADC2D   | ADC1D    | ADC0D   | 229       |
| (0x7D)      | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| (0x7C)      | ADMUX    | REFS1                                      | REFS0   | ADLAR   | MUX4    | MUX3    | MUX2    | MUX1     | MUX0    | 225       |
| (0x7B)      | ADCSRB   | -  | ACME    | -       | -       | -       | ADTS2   | ADTS1    | ADTS0   | 211/228   |
| (0x7A)      | ADCSRA   | ADEN                                       | ADSC    | ADATE   | ADIF    | ADIE    | ADPS2   | ADPS1    | ADPS0   | 227       |
| (0x79)      | ADCH     | ADC Data Register High                     |         |         |         |         |         |          |         | 228       |
| (0x78)      | ADCL     | ADC Data Register Low                      |         |         |         |         |         |          |         | 228       |
| (0x77)      | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| (0x76)      | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| (0x75)      | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| (0x74)      | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| (0x73)      | PCMSK3   | -  | PCINT30 | PCINT29 | PCINT28 | PCINT27 | PCINT26 | PCINT25  | PCINT24 | 62        |
| (0x72)      | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| (0x71)      | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| (0x70)      | TIMSK2   | -  | -       | -       | -       | -       | -       | OCIE2A   | TOIE2   | 158       |
| (0x6F)      | TIMSK1   | -  | -       | ICIE1   | -       | -       | OCIE1B  | OCIE1A   | TOIE1   | 132       |
| (0x6E)      | TIMSK0   | -  | -       | -       | -       | -       | -       | OCIE0A   | TOIE0   | 139       |
| (0x6D)      | PCMSK2   | PCINT23                                    | PCINT22 | PCINT21 | PCINT20 | PCINT19 | PCINT18 | PCINT17  | PCINT16 | 62        |
| (0x6C)      | PCMSK1   | PCINT15                                    | PCINT14 | PCINT13 | PCINT12 | PCINT11 | PCINT10 | PCINT9   | PCINT8  | 62        |
| (0x6B)      | PCMSK0   | PCINT7                                     | PCINT6  | PCINT5  | PCINT4  | PCINT3  | PCINT2  | PCINT1   | PCINT0  | 62        |
| (0x6A)      | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| (0x69)      | EICRA    | -  | -       | -       | -       | -       | -       | ISC01    | ISC00   | 59        |
| (0x68)      | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| (0x67)      | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| (0x66)      | OSCCAL   | Oscillator Calibration Register [CAL7...0] |         |         |         |         |         |          |         | 35        |
| (0x65)      | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| (0x64)      | PRR      | -  | PRR     | -       | PRLCD   | PRTIM1  | PRSPI   | PSUSART0 | PRADC   | 43        |
| (0x63)      | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| (0x62)      | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| (0x61)      | CLKPR    | CLKPCE                                     | -       | -       | -       | CLKPS3  | CLKPS2  | CLKPS1   | CLKPS0  | 35        |
| (0x60)      | WDTCR    | -  | -       | -       | WDCE    | WDE     | WDP2    | WDP1     | WDP0    | 50        |
| 0x3F (0x5F) | SREG     | I  | T       | H       | S       | V       | N       | Z        | C       | 12        |
| 0x3E (0x5E) | SPH      | Stack Pointer High                         |         |         |         |         |         |          |         | 14        |
| 0x3D (0x5D) | SPL      | Stack Pointer Low                          |         |         |         |         |         |          |         | 14        |
| 0x3C (0x5C) | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| 0x3B (0x5B) | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| 0x3A (0x5A) | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| 0x39 (0x59) | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| 0x38 (0x58) | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| 0x37 (0x57) | SPMCSR   | SPMIE                                      | RWWSB   | -       | RWWSRE  | BLBSET  | PGWRT   | PGERS    | SPMEN   | 294       |
| 0x36 (0x56) | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| 0x35 (0x55) | MCUCR    | JTD  | BODS    | BODSE   | PUD     | -       | -       | IVSEL    | IVCE    | 57/90/280 |
| 0x34 (0x54) | MCUSR    | -  | -       | -       | JTRF    | WDRF    | BORF    | EXTRF    | PORF    | 50        |
| 0x33 (0x53) | SMCR     | -  | -       | -       | -       | SM2     | SM1     | SM0      | SE      | 42        |
| 0x32 (0x52) | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| 0x31 (0x51) | OCDR     | IDRD/OCDR7                                 | OCDR6   | OCDR5   | OCDR4   | OCDR3   | OCDR2   | OCDR1    | OCDR0   | 253       |
| 0x30 (0x50) | ACSR     | ACD  | ACBG    | ACO     | ACI     | ACIE    | ACIC    | ACIS1    | ACIS0   | 211       |
| 0x2F (0x4F) | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| 0x2E (0x4E) | SPDR     | SPI Data Register                          |         |         |         |         |         |          |         | 169       |
| 0x2D (0x4D) | SPSR     | SPIF                                       | WCOL    | -       | -       | -       | -       | -        | SPI2X   | 168       |
| 0x2C (0x4C) | SPCR     | SPIE                                       | SPE     | DORD    | MSTR    | CPOL    | CPHA    | SPR1     | SPR0    | 167       |
| 0x2B (0x4B) | GPIOR2   | General Purpose I/O Register               |         |         |         |         |         |          |         | 26        |
| 0x2A (0x4A) | GPIOR1   | General Purpose I/O Register               |         |         |         |         |         |          |         | 26        |
| 0x29 (0x49) | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| 0x28 (0x48) | Reserved | -  | -       | -       | -       | -       | -       | -        | -       |           |
| 0x27 (0x47) | OCR0A    | Timer/Counter0 Output Compare A            |         |         |         |         |         |          |         | 139       |

| Address     | Name     | Bit 7                        | Bit 6  | Bit 5  | Bit 4  | Bit 3  | Bit 2  | Bit 1                        | Bit 0  | Page    |
|-------------|----------|------------------------------|--------|--------|--------|--------|--------|------------------------------|--------|---------|
| 0x26 (0x46) | TCNT0    | Timer/Counter0               |        |        |        |        |        |                              |        | 138     |
| 0x25 (0x45) | Reserved | -                            | -      | -      | -      | -      | -      | -                            | -      |         |
| 0x24 (0x44) | TCCR0A   | FOC0A                        | WGM00  | COM0A1 | COM0A0 | WGM01  | CS02   | CS01                         | CS00   | 136     |
| 0x23 (0x43) | GTCCR    | TSM                          | -      | -      | -      | -      | -      | PSR2                         | PSR10  | 140/159 |
| 0x22 (0x42) | EEARH    | -                            | -      | -      | -      | -      | -      | EEPROM Address Register High |        | 22      |
| 0x21 (0x41) | EEARL    | EEPROM Address Register Low  |        |        |        |        |        |                              |        | 22      |
| 0x20 (0x40) | EEDR     | EEPROM Data Register         |        |        |        |        |        |                              |        | 22      |
| 0x1F (0x3F) | EECR     | -                            | -      | -      | -      | EERIE  | EEMWE  | EEWE                         | EERE   | 23      |
| 0x1E (0x3E) | GPIOR0   | General Purpose I/O Register |        |        |        |        |        |                              |        | 26      |
| 0x1D (0x3D) | EIMSK    | PCIE                         | PCIE2  | PCIE1  | PCIE0  | -      | -      | -                            | INT0   | 60      |
| 0x1C (0x3C) | EIFR     | PCIF3                        | PCIF2  | PCIF1  | PCIF0  | -      | -      | -                            | INTF0  | 61      |
| 0x1B (0x3B) | Reserved | -                            | -      | -      | -      | -      | -      | -                            | -      |         |
| 0x1A (0x3A) | Reserved | -                            | -      | -      | -      | -      | -      | -                            | -      |         |
| 0x19 (0x39) | Reserved | -                            | -      | -      | -      | -      | -      | -                            | -      |         |
| 0x18 (0x38) | Reserved | -                            | -      | -      | -      | -      | -      | -                            | -      |         |
| 0x17 (0x37) | TIFR2    | -                            | -      | -      | -      | -      | -      | OCF2A                        | TOV2   | 159     |
| 0x16 (0x36) | TIFR1    | -                            | -      | ICF1   | -      | -      | OCF1B  | OCF1A                        | TOV1   | 133     |
| 0x15 (0x35) | TIFR0    | -                            | -      | -      | -      | -      | -      | OCF0A                        | TOV0   | 140     |
| 0x14 (0x34) | PORTG    | -                            | -      | -      | PORTG4 | PORTG3 | PORTG2 | PORTG1                       | PORTG0 | 92      |
| 0x13 (0x33) | DDRG     | -                            | -      | -      | DDG4   | DDG3   | DDG2   | DDG1                         | DDG0   | 92      |
| 0x12 (0x32) | PING     | -                            | -      | PING5  | PING4  | PING3  | PING2  | PING1                        | PING0  | 92      |
| 0x11 (0x31) | PORTF    | PORTF7                       | PORTF6 | PORTF5 | PORTF4 | PORTF3 | PORTF2 | PORTF1                       | PORTF0 | 92      |
| 0x10 (0x30) | DDRF     | DDF7                         | DDF6   | DDF5   | DDF4   | DDF3   | DDF2   | DDF1                         | DDF0   | 92      |
| 0x0F (0x2F) | PINF     | PINF7                        | PINF6  | PINF5  | PINF4  | PINF3  | PINF2  | PINF1                        | PINF0  | 92      |
| 0x0E (0x2E) | PORTE    | PORTE7                       | PORTE6 | PORTE5 | PORTE4 | PORTE3 | PORTE2 | PORTE1                       | PORTE0 | 91      |
| 0x0D (0x2D) | DDRE     | DDE7                         | DDE6   | DDE5   | DDE4   | DDE3   | DDE2   | DDE1                         | DDE0   | 91      |
| 0x0C (0x2C) | PINE     | PINE7                        | PINE6  | PINE5  | PINE4  | PINE3  | PINE2  | PINE1                        | PINE0  | 92      |
| 0x0B (0x2B) | PORTD    | PORTD7                       | PORTD6 | PORTD5 | PORTD4 | PORTD3 | PORTD2 | PORTD1                       | PORTD0 | 91      |
| 0x0A (0x2A) | DDRD     | DDD7                         | DDD6   | DDD5   | DDD4   | DDD3   | DDD2   | DDD1                         | DDD0   | 91      |
| 0x09 (0x29) | PIND     | PIND7                        | PIND6  | PIND5  | PIND4  | PIND3  | PIND2  | PIND1                        | PIND0  | 91      |
| 0x08 (0x28) | PORTC    | PORTC7                       | PORTC6 | PORTC5 | PORTC4 | PORTC3 | PORTC2 | PORTC1                       | PORTC0 | 91      |
| 0x07 (0x27) | DDRC     | DDC7                         | DDC6   | DDC5   | DDC4   | DDC3   | DDC2   | DDC1                         | DDC0   | 91      |
| 0x06 (0x26) | PINC     | PINC7                        | PINC6  | PINC5  | PINC4  | PINC3  | PINC2  | PINC1                        | PINC0  | 91      |
| 0x05 (0x25) | PORTB    | PORTB7                       | PORTB6 | PORTB5 | PORTB4 | PORTB3 | PORTB2 | PORTB1                       | PORTB0 | 90      |
| 0x04 (0x24) | DDRB     | DDB7                         | DDB6   | DDB5   | DDB4   | DDB3   | DDB2   | DDB1                         | DDB0   | 90      |
| 0x03 (0x23) | PINB     | PINB7                        | PINB6  | PINB5  | PINB4  | PINB3  | PINB2  | PINB1                        | PINB0  | 90      |
| 0x02 (0x22) | PORTA    | PORTA7                       | PORTA6 | PORTA5 | PORTA4 | PORTA3 | PORTA2 | PORTA1                       | PORTA0 | 90      |
| 0x01 (0x21) | DDRA     | DDA7                         | DDA6   | DDA5   | DDA4   | DDA3   | DDA2   | DDA1                         | DDA0   | 90      |
| 0x00 (0x20) | PINA     | PINA7                        | PINA6  | PINA5  | PINA4  | PINA3  | PINA2  | PINA1                        | PINA0  | 90      |

- Notes:
- 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 AVRs, 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 ATmega329P/3290P 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.

## 7. Instruction Set Summary

| Mnemonics                                | Operands | Description                              | Operation   | Flags      | #Clocks |
|--|----------|--|---|------------|---------|
| <b>ARITHMETIC AND LOGIC INSTRUCTIONS</b> |          |  |   |            |         |
| 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                                     | RdI,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                                     | RdI,K    | Subtract Immediate from Word             | $Rdh:Rdl \leftarrow Rdh:Rdl - K$                      | Z,C,N,V,S  | 2       |
| AND                                      | Rd, Rr   | Logical AND Registers                    | $Rd \leftarrow Rd \bullet Rr$                         | Z,N,V      | 1       |
| ANDI                                     | Rd, K    | Logical AND Register and Constant        | $Rd \leftarrow Rd \bullet K$                          | 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                 | $Rd \leftarrow Rd \bullet (0xFF - K)$                 | 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                   | $Rd \leftarrow Rd \bullet Rd$                         | 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       |
| <b>BRANCH INSTRUCTIONS</b>               |          |  |   |            |         |
| 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 $\oplus$ V = 0) then $PC \leftarrow PC + k + 1$ | None       | 1/2     |
| BRLT                                     | k        | Branch if Less Than Zero, Signed         | if (N $\oplus$ V = 1) then $PC \leftarrow PC + k + 1$ | None       | 1/2     |
| BRHS                                     | k        | Branch if Half Carry Flag Set            | if (H = 1) then $PC \leftarrow PC + k + 1$            | None       | 1/2     |
| BRHC                                     | k        | Branch if Half Carry Flag Cleared        | if (H = 0) then $PC \leftarrow PC + k + 1$            | None       | 1/2     |
| BRTS                                     | k        | Branch if T Flag Set                     | if (T = 1) then $PC \leftarrow PC + k + 1$            | None       | 1/2     |
| BRTC                                     | k        | Branch if T Flag Cleared                 | if (T = 0) then $PC \leftarrow PC + k + 1$            | None       | 1/2     |
| BRVS                                     | k        | Branch if Overflow Flag is Set           | if (V = 1) then $PC \leftarrow PC + k + 1$            | None       | 1/2     |

| 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     |
| <b>BIT AND BIT-TEST INSTRUCTIONS</b> |          |                                    |  |         |         |
| 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       |
| <b>DATA TRANSFER INSTRUCTIONS</b>    |          |                                    |  |         |         |
| 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       |

| Mnemonics                       | Operands | Description             | Operation                                | Flags | #Clocks |
|---------------------------------|----------|-------------------------|--|-------|---------|
| PUSH                            | Rr       | Push Register on Stack  | STACK ← Rr                               | None  | 2       |
| POP                             | Rd       | Pop Register from Stack | Rd ← STACK                               | None  | 2       |
| <b>MCU CONTROL INSTRUCTIONS</b> |          |                         |  |       |         |
| 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     |

## 8. Ordering Information

### 8.1 ATmega329P

| Speed (MHz) <sup>(3)</sup> | Power Supply | Ordering Code <sup>(2)</sup>   | Package Type <sup>(1)</sup> | Operational Range                           |
|----------------------------|--------------|--|-----------------------------|---|
| 10                         | 1.8 - 5.5V   | ATmega329PV-10AU<br>ATmega329PV-10AUR <sup>(4)</sup><br>ATmega329PV-10MU<br>ATmega329PV-10MUR <sup>(4)</sup> | 64A<br>64A<br>64M1<br>64M1  | Industrial<br>(-40°C to 85°C)               |
| 20                         | 2.7 - 5.5V   | ATmega329P-20AU<br>ATmega329P-20AUR <sup>(4)</sup><br>ATmega329P-20MU<br>ATmega329P-20MUR <sup>(4)</sup>     | 64A<br>64A<br>64M1<br>64M1  |   |
| 20                         | 1.8 - 5.5V   | ATmega329P-20AN<br>ATmega329P-20ANR <sup>(4)</sup><br>ATmega329P-20MN<br>ATmega329P-20MNR <sup>(4)</sup>     | 64A<br>64A<br>64M1<br>64M1  | Extended<br>(-40°C to 105°C) <sup>(5)</sup> |

- Notes:
1. This device can also be supplied in wafer form. Please contact your local Atmel sales office for detailed ordering information and minimum quantities.
  2. Pb-free packaging complies to the European Directive for Restriction of Hazardous Substances (RoHS directive). Also Halide free and fully Green.
  3. For Speed vs.  $V_{CC}$  see [Figure 28-2 on page 334](#) and [Figure 28-3 on page 334](#).
  4. Tape & Reel
  5. See Appendix A ATmega169PA/329P/3290P 105°C

| Package Type |  |
|--------------|--|
| <b>64A</b>   | 64-lead, 14 x 14 x 1.0 mm, Thin Profile Plastic Quad Flat Package (TQFP)     |
| <b>64M1</b>  | 64-pad, 9 x 9 x 1.0 mm, Quad Flat No-Lead/Micro Lead Frame Package (QFN/MLF) |

## 8.2 ATmega3290P

| Speed (MHz) <sup>(3)</sup> | Power Supply | Ordering Code <sup>(2)</sup>   | Package Type <sup>(1)</sup> | Operational Range                           |
|----------------------------|--------------|--|-----------------------------|---|
| 10                         | 1.8 - 5.5V   | ATmega3290PV-10AU<br>ATmega3290PV-10AUR <sup>(4)</sup>   | 100A                        | Industrial<br>(-40°C to 85°C)               |
| 20                         | 2.7 - 5.5V   | ATmega3290P-20AU<br>ATmega3290P-20AUR <sup>(4)</sup>   |                             |   |
| 20                         | 1.8 - 5.5V   | ATmega3290P-20AN<br>ATmega3290P-20ANR <sup>(4)</sup><br>ATmega3290P-20MN<br>ATmega3290P-20MNR <sup>(4)</sup> | 64A<br>64A<br>64M1<br>64M1  | Extended<br>(-40°C to 105°C) <sup>(5)</sup> |

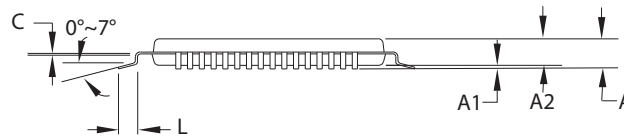
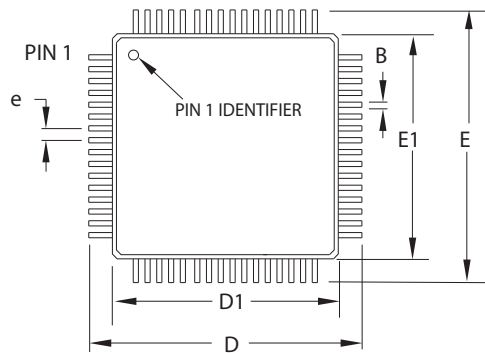
- Notes:
1. This device can also be supplied in wafer form. Please contact your local Atmel sales office for detailed ordering information and minimum quantities.
  2. Pb-free packaging complies to the European Directive for Restriction of Hazardous Substances (RoHS directive). Also Halide free and fully Green.
  3. For Speed vs.  $V_{CC}$  see [Figure 28-2 on page 334](#) and [Figure 28-3 on page 334](#).
  4. Tape & Reel
  5. See Appendix A ATmega169PA/329P/3290P 105°C

| Package Type |  |
|--------------|--|
| <b>100A</b>  | 100-lead, 14 x 14 x 1.0 mm, 0.5 mm Lead Pitch, Thin Profile Plastic Quad Flat Package (TQFP) |



## 9. Packaging Information

### 9.1 64A



COMMON DIMENSIONS  
(Unit of measure = mm)

| SYMBOL | MIN      | NOM   | MAX   | NOTE   |
|--------|----------|-------|-------|--------|
| A      | -        | -     | 1.20  |        |
| A1     | 0.05     | -     | 0.15  |        |
| A2     | 0.95     | 1.00  | 1.05  |        |
| D      | 15.75    | 16.00 | 16.25 |        |
| D1     | 13.90    | 14.00 | 14.10 | Note 2 |
| E      | 15.75    | 16.00 | 16.25 |        |
| E1     | 13.90    | 14.00 | 14.10 | Note 2 |
| B      | 0.30-    | 0.45  |       |        |
| C      | 0.09     | -     | 0.20  |        |
| L      | 0.45     | -     | 0.75  |        |
| e      | 0.80 TYP |       |       |        |

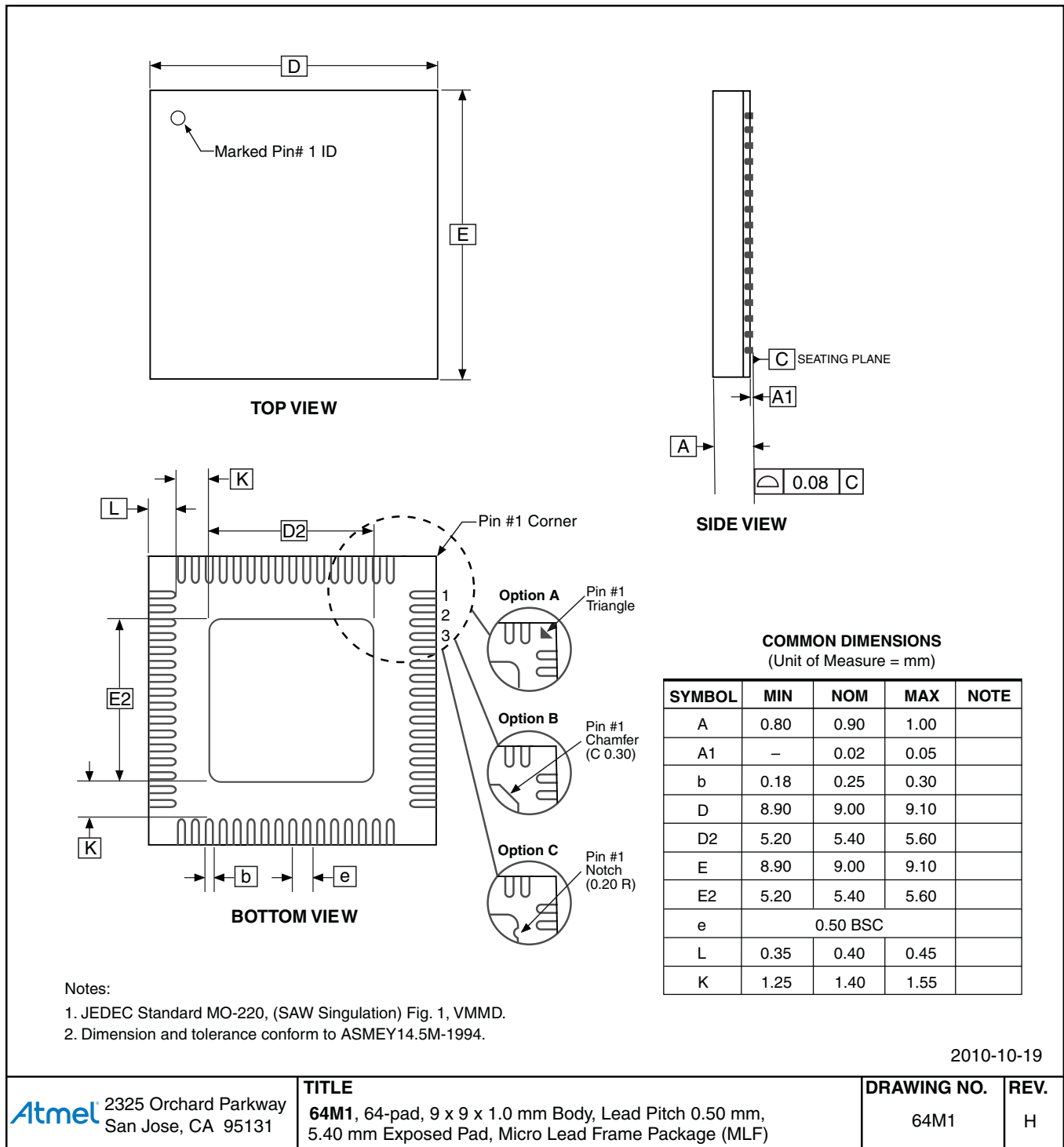
Notes:

1. This package conforms to JEDEC reference MS-026, Variation AEB.
2. Dimensions D1 and E1 do not include mold protrusion. Allowable protrusion is 0.25mm per side. Dimensions D1 and E1 are maximum plastic body size dimensions including mold mismatch.
3. Lead coplanarity is 0.10mm maximum.

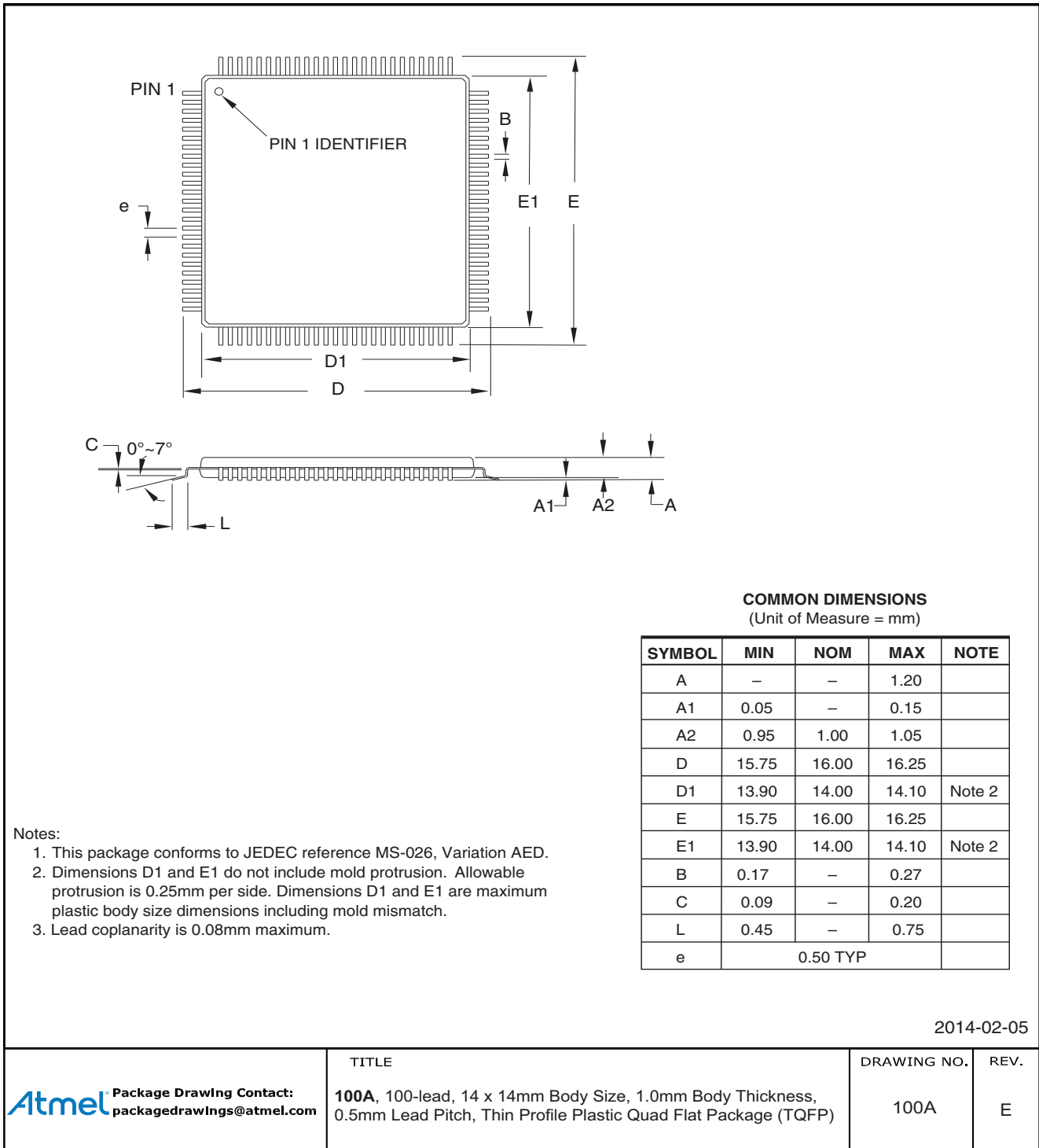
2010-10-20

|  |   |             |      |
|--|---|-------------|------|
|  2325 Orchard Parkway<br>San Jose, CA 95131 | TITLE   | DRAWING NO. | REV. |
|  | 64A, 64-lead, 14 x 14mm Body Size, 1.0mm Body Thickness,<br>0.8mm Lead Pitch, Thin Profile Plastic Quad Flat Package (TQFP) | 64A         | C    |

## 9.2 64M1



## 9.3 100A



## 10. Errata

### 10.1 ATmega329P rev. A

- Interrupts may be lost when writing the timer registers in the asynchronous timer
- Using BOD disable will make the chip reset

#### 1. Interrupts may be lost when writing the timer registers in the asynchronous timer

The interrupt will be lost if a timer register that is synchronous timer clock is written when the asynchronous Timer/Counter register (TCNTx) is 0x00.

##### Problem Fix/ Workaround

Always check that the asynchronous Timer/Counter register neither have the value 0xFF nor 0x00 before writing to the asynchronous Timer Control Register (TCCRx), asynchronous Timer Counter Register (TCNTx), or asynchronous Output Compare Register (OCRx).

#### 2. Using BOD disable will make the chip reset

If the part enters sleep with the BOD turned off with the BOD disable option enabled, a BOD reset will be generated at wakeup and the chip will reset.

##### Problem Fix/Workaround

Do not use BOD disable

### 10.2 ATmega329P rev. B

- Interrupts may be lost when writing the timer registers in the asynchronous timer

#### 1. Interrupts may be lost when writing the timer registers in the asynchronous timer

The interrupt will be lost if a timer register that is synchronous timer clock is written when the asynchronous Timer/Counter register (TCNTx) is 0x00.

##### Problem Fix/ Workaround

Always check that the asynchronous Timer/Counter register neither have the value 0xFF nor 0x00 before writing to the asynchronous Timer Control Register (TCCRx), asynchronous Timer Counter Register (TCNTx), or asynchronous Output Compare Register (OCRx).

### 10.3 ATmega329P rev. C

- Interrupts may be lost when writing the timer registers in the asynchronous timer

#### 1. Interrupts may be lost when writing the timer registers in the asynchronous timer

The interrupt will be lost if a timer register that is synchronous timer clock is written when the asynchronous Timer/Counter register (TCNTx) is 0x00.

##### Problem Fix/ Workaround

Always check that the asynchronous Timer/Counter register neither have the value 0xFF nor 0x00 before writing to the asynchronous Timer Control Register (TCCRx), asynchronous Timer Counter Register (TCNTx), or asynchronous Output Compare Register (OCRx).

## 10.4 ATmega3290P rev. A

- Interrupts may be lost when writing the timer registers in the asynchronous timer
- Using BOD disable will make the chip reset

### 1. Interrupts may be lost when writing the timer registers in the asynchronous timer

The interrupt will be lost if a timer register that is synchronous timer clock is written when the asynchronous Timer/Counter register (TCNTx) is 0x00.

#### Problem Fix/ Workaround

Always check that the asynchronous Timer/Counter register neither have the value 0xFF nor 0x00 before writing to the asynchronous Timer Control Register (TCCRx), asynchronous Timer Counter Register (TCNTx), or asynchronous Output Compare Register (OCRx).

### 2. Using BOD disable will make the chip reset

If the part enters sleep with the BOD turned off with the BOD disable option enabled, a BOD reset will be generated at wakeup and the chip will reset.

#### Problem Fix/Workaround

Do not use BOD disable

## 10.5 ATmega3290P rev. B

- Interrupts may be lost when writing the timer registers in the asynchronous timer

### 1. Interrupts may be lost when writing the timer registers in the asynchronous timer

The interrupt will be lost if a timer register that is synchronous timer clock is written when the asynchronous Timer/Counter register (TCNTx) is 0x00.

#### Problem Fix/ Workaround

Always check that the asynchronous Timer/Counter register neither have the value 0xFF nor 0x00 before writing to the asynchronous Timer Control Register (TCCRx), asynchronous Timer Counter Register (TCNTx), or asynchronous Output Compare Register (OCRx).

## 10.6 ATmega3290P rev. C

- Interrupts may be lost when writing the timer registers in the asynchronous timer

### 1. Interrupts may be lost when writing the timer registers in the asynchronous timer

The interrupt will be lost if a timer register that is synchronous timer clock is written when the asynchronous Timer/Counter register (TCNTx) is 0x00.

#### Problem Fix/ Workaround

Always check that the asynchronous Timer/Counter register neither have the value 0xFF nor 0x00 before writing to the asynchronous Timer Control Register (TCCRx), asynchronous Timer Counter Register (TCNTx), or asynchronous Output Compare Register (OCRx).

## 11. Datasheet Revision History

Refer to the complete datasheet for revision history.

**Atmel Corporation**      1600 Technology Drive, San Jose, CA 95110 USA      T: (+1)(408) 441.0311      F: (+1)(408) 436.4200      |      [www.atmel.com](http://www.atmel.com)

© 2015 Atmel Corporation. / Rev.: Atmel-8021HS-AVR-ATmega329P-3290P\_Datasheet\_07/2015.

Atmel®, Atmel logo and combinations thereof, Enabling Unlimited Possibilities®, AVR® and others are registered trademarks or trademarks of Atmel Corporation in U.S. and other countries. Other terms and product names may be trademarks of others.

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

**SAFETY-CRITICAL, MILITARY, AND AUTOMOTIVE APPLICATIONS DISCLAIMER:** Atmel products are not designed for and will not be used in connection with any applications where the failure of such products would reasonably be expected to result in significant personal injury or death ("Safety-Critical Applications") without an Atmel officer's specific written consent. Safety-Critical Applications include, without limitation, life support devices and systems, equipment or systems for the operation of nuclear facilities and weapons systems. Atmel products are not designed nor intended for use in military or aerospace applications or environments unless specifically designated by Atmel as military-grade. Atmel products are not designed nor intended for use in automotive applications unless specifically designated by Atmel as automotive-grade.

## Looking for pricing, stock, or lifecycle information?

Click below to explore more details on WIN SOURCE:

 [View ATMEGA329PV-10AU on WIN SOURCE](#)

 [Atmel Information](#)

## Optimize Your Supply Chain with WIN SOURCE Solutions

-  Global Sourcing Solution
-  Obsolete Management
-  Cost Control Management
-  Shortage Management
-  Alternative Solution
-  Excess Inventory Management