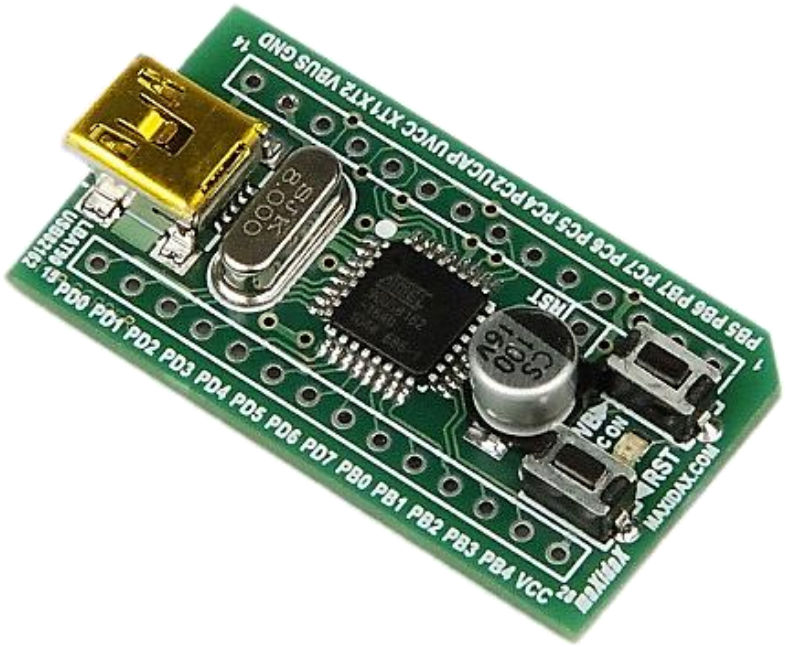
LBAT90USB162

Atmel®
AT90USB162
Development Board

User's manual



1. INTRODUCTION

Thank you for choosing the LBAT90USB162 – Atmel® AT90USB162 development board. This board is designed to give quick and cost-effective start to develop code and for hardware prototyping and testing.

1.1. OVERVIEW

This document describes the LBAT90USB162 development board – a cost-effective yet feature rich, highly compatible, flexible and easily configurable development tool, designed to allow easy coding and prototyping.

One of today's most widely used MCUs offering native USB support is the Atmel® AT90USB162. It provides an easy way to add USB functionality to any new design along with hundreds of different projects that are available.

The LBAT90USB162 is an Atmel® AT90USB162 development board. This board is a cost-effective yet highly compatible, flexible and easy-to-use development tool, designed to give a quick start in developing code and for hardware prototyping and testing. It is clean in design and provides flexibility and convenience. The board has all the basic circuitry needed to work with the AT90USB162: USB connector and circuit, crystal, Reset and HWB buttons, power LED.

The board offers flexible power: both 3.3 V and 5 V USB -powered or from external supply.

The board also features 100 mil headers, making it breadboard-friendly and easily connectable to any universal board or prototyping environment.

Board design makes it compatible with other similar development boards featuring same or similar MCUs and virtually all design/development software and libraries.

MCU comes pre-programmed with a bootloader allowing code to be programmed into the chip without any external programmer – simply by using the USB interface and the FLIP software from Atmel. To enter the bootloader mode, the RST button should be pressed and hold, then the HWB button pressed and hold, and then the RST button released.

Any development board will probably see many versions of firmware. What if the bootloader gets destroyed cutting the way to reprogram the device via USB? Using external programmer connected via ISP or parallel is the only solution. Although this design doesn't have a dedicated ISP port, the ISP port of the MCU still can be used - simply disconnect any circuitry from lines PB1/SCLK, PB2/MOSI, PB3/MISO and use them together with the dedicated Reset (RST) header to reprogram the device (see LBAT90USB162 User's manual and AT90USB162 datasheet). This option isn't directly available with many other development boards. Hardware debugger like JTAGICE can also be used via the RST header.

1.2. FEATURES

- Flexible power – 3.3 V or 5 V from USB bus or external 3.0 V to 5.5 V. **Important:** To power the board properly external connections are needed – please see section 2.2.6;
- All MCU I/O pins are accessible;
- 100 mil extension headers for connection to universal or breadboards, or direct interfacing with other devices, providing access to all I/O pins and virtually all signals;
- Mini USB connector – USB 2.0 full speed (12 Mbps) Device mode supported;
- Built-in 0.9..16 MHz crystal (default 8 MHz);
- Internal or external clock;
- Optional transient voltage suppression for USB data lines;
- Hardware boot (HWB) button – allows forcing bootloader (stock Atmel® or third-party) execution at reset (see AT90USB162 datasheet) – allows MCU programming via USB without external programmer. I/O pin is still useable for other purposes;
- Reset (RST) button;
- Reset (RST) header – allows using MCU's ISP port to reprogram the device, if the bootloader gets damaged or destroyed (see AT90USB162 datasheet). It also makes possible the use of hardware debugger like JTAGICE.

1.3. SPECIFICATIONS

Processor	AT90USB162
Max. frequency	8 MHz at 2.7 V, 16 MHz at 4.5 V
Flash memory	16 kBytes (10,000 write cycles)
RAM	512 Bytes (static)
EEPROM	512 Bytes (100,000 write cycles)
USB	USB 2.0 full speed (12 Mbps) Device mode
USB DPRAM	176 Bytes
I/O lines	22
Timers	1 8-bit, 1 16-bit
PWM channels	5
USART	1
SPI port	1
Programming	Via USB and hardware-initiated bootloader or ISP connector
Debugging	Via debugWIRE interface (ISP connector)
Operating voltage	2.7 V to 5.5 V
Operating temperature	Industrial -40°C to +85°C
Dimensions	41 x 23 x 8 mm (1.6 x 0.9 x 0.3 in) FR-4 1.5 mm

1.4. COMPATIBILITY

- As all MCU I/O pins are accessible and all MCU powering and clocking options are available the SBAT90USB162a is compatible with virtually every project and development tool designed for AT90USB162 and particularly for AVR MCUs.

1.5. MCU OVERVIEW

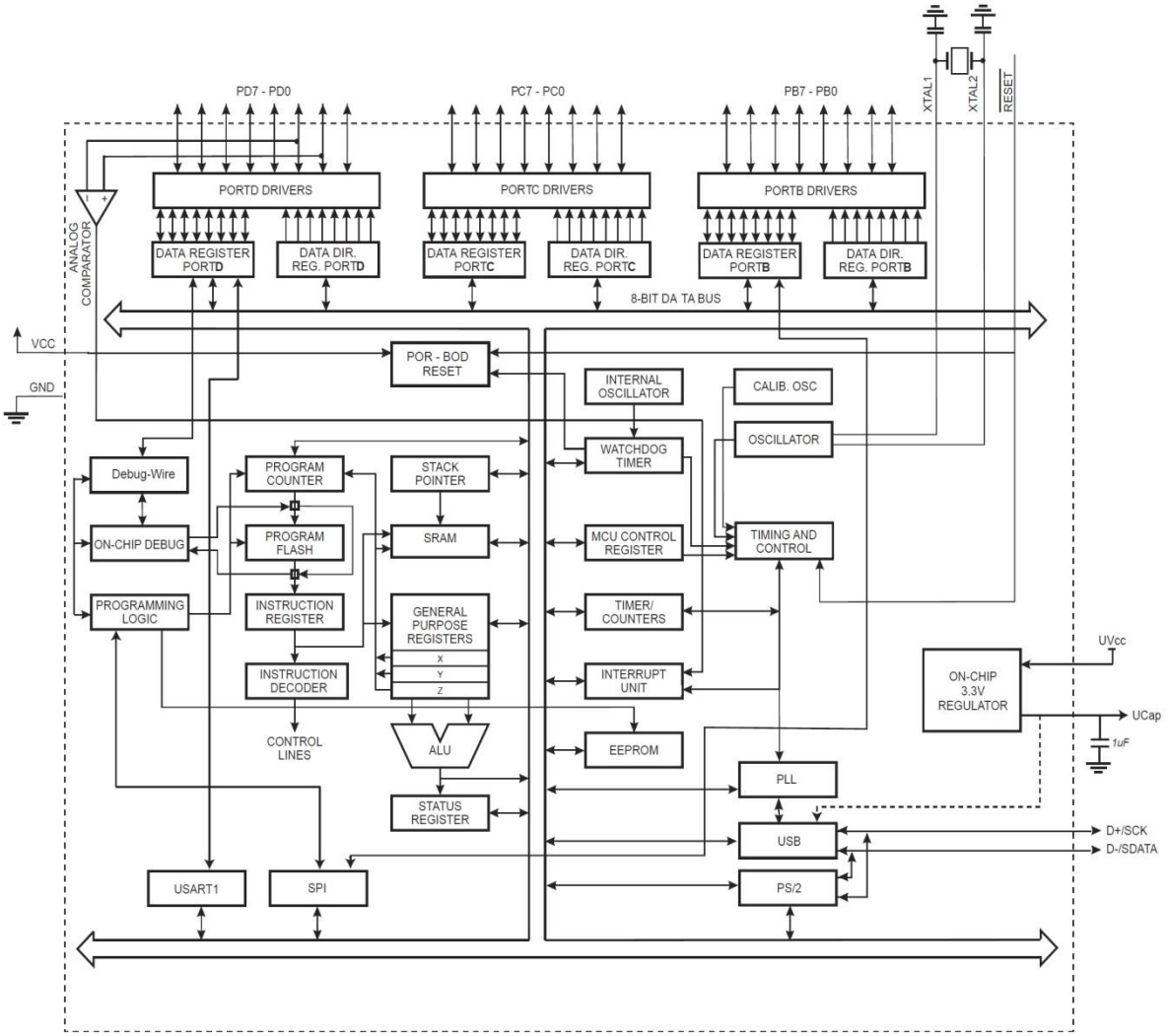
The AT90USB162 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the AT90USB162 achieves throughputs approaching 1 MIPS per MHz allowing optimization of power consumption versus processing speed.

1.5.1. Features

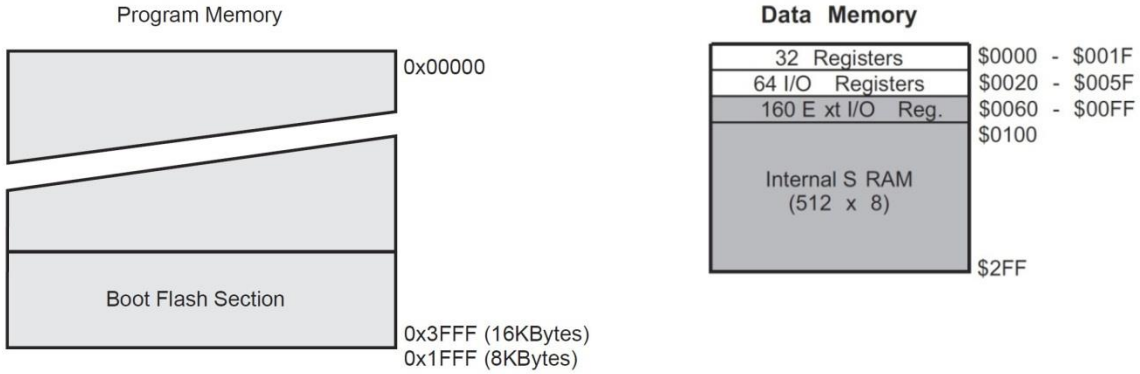
- High Performance, Low Power AVR® 8-Bit Microcontroller
- Advanced RISC Architecture
  - 125 Powerful Instructions
  - Most Single Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 16 MIPS Throughput at 16 MHz
- Non-volatile Program and Data Memories
  - 16K Bytes of In-System Self-Programmable Flash
    - Endurance: 10,000 Write/Erase Cycles
  - Optional Boot Code Section with Independent Lock Bits
    - USB boot-loader programmed by default in the factory
    - In-System Programming by on-chip Boot Program hardware-activated after reset
    - True Read-While-Write Operation
  - 512 Bytes EEPROM
    - Endurance: 100,000 Write/Erase Cycle
  - 512 Bytes Internal SRAM
  - Programming Lock for Software Security
- USB 2.0 Full-speed Device Module with Interrupt on Transfer Completion
  - Complies fully with Universal Serial Bus Specification REV 2.0
  - 48 MHz PLL for Full-speed Bus Operation: data transfer rates at 12 Mbit/s
  - Fully independent 176 bytes USB DPRAM for endpoint memory allocation
  - Endpoint 0 for Control Transfers: from 8 up to 64-bytes
  - 4 Programmable Endpoints:
    - IN or Out Directions
    - Bulk, Interrupt and Isochronous Transfers
    - Programmable maximum packet size from 8 to 64 bytes
    - Programmable single or double buffer
  - Suspend/Resume Interrupts
  - Microcontroller reset on USB Bus Reset without detach
  - USB Bus Disconnection on Microcontroller Request
  - USB pad multiplexed with PS/2 peripheral for single cable capability
- Peripheral Features
  - PS/2 compliant pad
  - One 8-bit Timer/Counters with Separate Prescaler and Compare Mode (two 8-bit PWM channels)
  - One 16-bit Timer/Counter with Separate Prescaler, Compare and Capture Mode (three 8-bit PWM channels)
  - USART with SPI master only mode and hardware flow control (RTS/CTS)

  - Master/Slave SPI Serial Interface
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - On-chip Analog Comparator
  - Interrupt and Wake-up on Pin Change
- On Chip Debug Interface (debugWIRE)
- Special Microcontroller Features
  - Power-On Reset and Programmable Brown-out Detection
  - Internal Calibrated Oscillator
  - External and Internal Interrupt Sources
  - Five Sleep Modes: Idle, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
  - 22 Programmable I/O Lines
  - QFN32 (5x5 mm) / TQFP32 packages
- Operating Voltages
  - 2.7 - 5.5 V
- Operating temperature
  - Industrial (-40 °C to +85 °C)
- Maximum Frequency
  - 8 MHz at 2.7 V - Industrial range
  - 16 MHz at 4.5 V - Industrial range

1.5.2. Block diagram



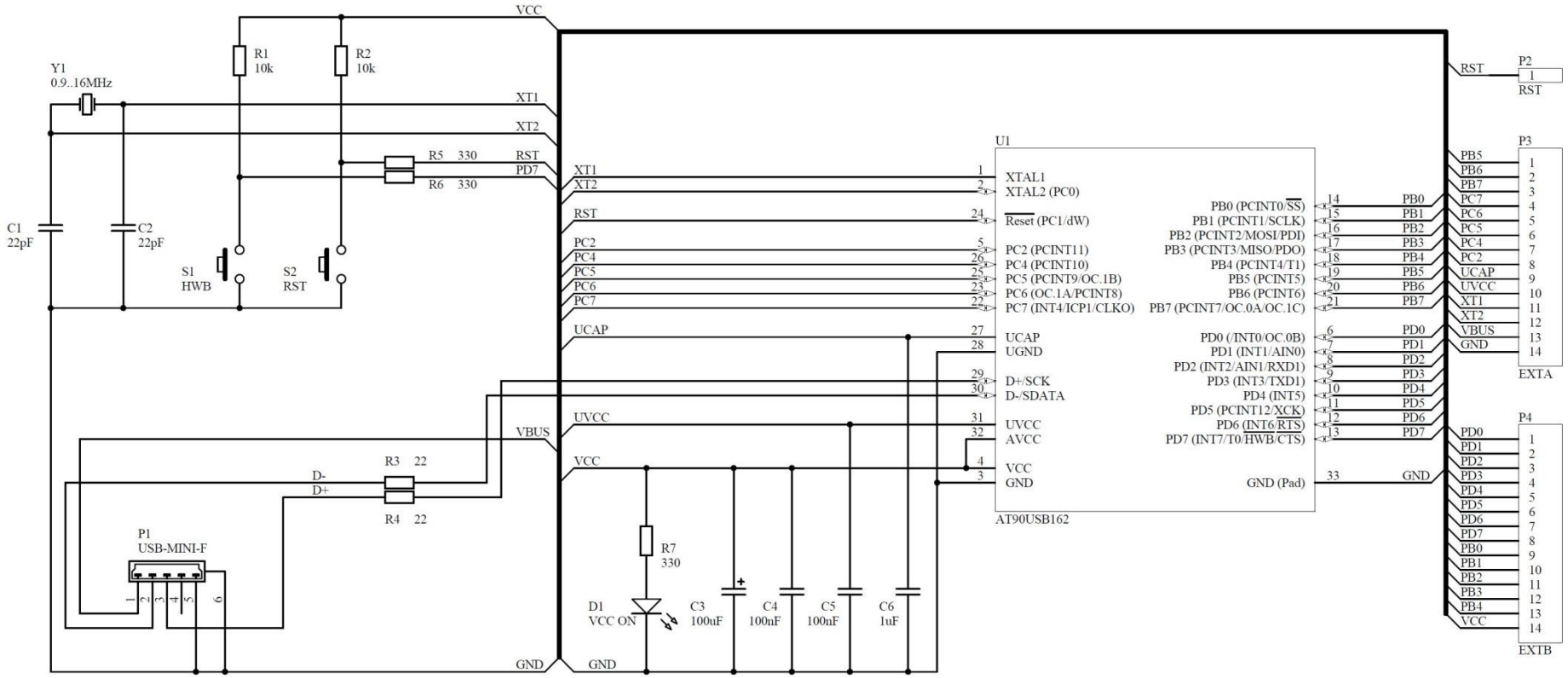
1.5.3. Memory map



2. USING THE LBAT90USB162

This section describes the board and all its features

2.1. SCHEMATIC



2.2. BOARD DESCRIPTION AND CONFIGURATION

2.2.1. Clock

By default the clock is internal from 0.9..16 MHz[1] crystal oscillator, or internal calibrated RC oscillator. External[2] clock via XT1 (P3-11) is also possible. For using external clock CKSEL fuses must be programmed (see AT90USB162 datasheet).

Notes: 1. Default is 8 MHz crystal, other is possible by request.
2. XT2 (PC0) can be used as generic I/O depending on MCU configuration.

2.2.2. HWB button (S1)

The hardware boot button allows forcing bootloader execution after reset (see AT90USB162 datasheet) thus allowing MCU programming via USB without external programmer. The HWB mode is active only when the HWBE fuse is enabled. In that case PD7/HWB pin is configured as input during reset and sampled during reset rising edge.

Programming via USB is based on pre-programmed USB bootloader, located in the on-chip memory boot section of the AT90USB162. This is the easiest and fastest way to reprogram the device directly over the USB interface, but with certain limitations. To force bootloader execution, operate as follows:

- Press both “RST” and “HWB” buttons;
- Release the “RST” button;
- Release the “HWB” button.

FLIP (Flexible In-system Programmer) is the software provided by Atmel® to do in-system programming of Flash devices through RS232, USB or CAN. For further details regarding programming via USB, please refer to FLIP documentation.

Note: In order to use FLIP, driver installation is required. USB drivers come with FLIP and can be found in its install folder. With Windows® operating system, depending on version used, digitally signed drivers might be required. As they may not be supplied by Atmel®, third-party signed drivers can be downloaded from the net. Maxidax Ltd. doesn’t supply any drivers.

Tip: After programming is complete press the “RST” button.

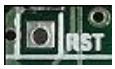
Other bootloaders are also available.

2.2.3. Reset button (S2)

Cold reset can be done manually by RST button (S2).

2.2.4. Reset (RST) header

The Reset (RST) header allows the use of ISP programmer in case the bootloader is damaged or destroyed. In that case that will be the only option to reprogram the device.



To use in-circuit programming, disconnect any circuitry from PB1/SCLK, PB2/MOSI, PB3/MISO lines and connect the ISP programmer to them and to the RST header.

In order to use the ISP interface the SPIEN fuse must be programmed.

The RST header also supports the debugWIRE interface for communication with debugging device. To enable the debugWIRE interface on an AVR device, the DWEN fuse must be programmed (DWEN = 0). AVR devices featuring debugWIRE are shipped with the DWEN fuse unprogrammed. ISP or High-Voltage Programming is required to enable debugWIRE.

2.2.5. USB connector

Mini-B USB (P1)[(1)] is available.

USB Mini connector pinout:

Pin	Signal	Description
1	VCC	+5 V
2	D-	Data -
3	D+	Data +
4	ID	NC for type B
5	GND	Ground

Note: 1. New, unprogrammed AT90USB162 device is seen via USB as “AT90USB162 DFU”. To use USB driver installation is required. USB

drivers can be downloaded from Atmel's site: http://atmel.com/*. Third-party drivers are also available on the net.

2.2.6. Powering the board

Both 3.3 V and 5 V VCC power can be used, supplied either from USB or as self-powered device from external source.

Important: **To power the board external connections are needed. Wiring diagrams below must be strictly followed in order to power the board properly. Missing that, or feeding power simply between VCC and GND will cause board to malfunction!**

Fig. 1. **Bus powered** application, **3V** I/O

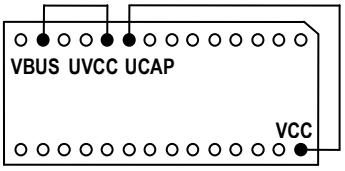


Fig. 2. **Bus powered** application, **5V** I/O

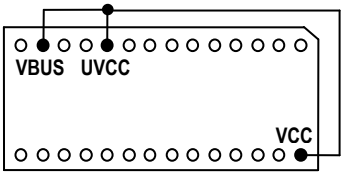


Fig. 3. **Self-powered** application, **3V** I/O

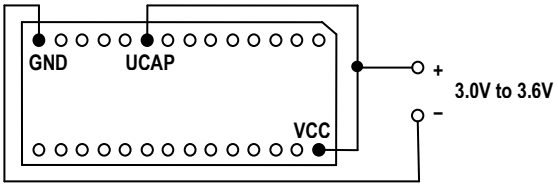
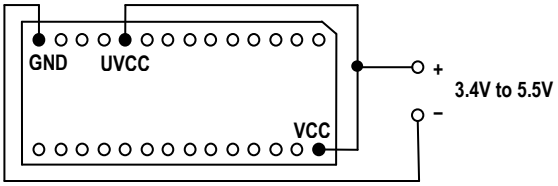


Fig. 4. **Self-powered** application, **5V** I/O

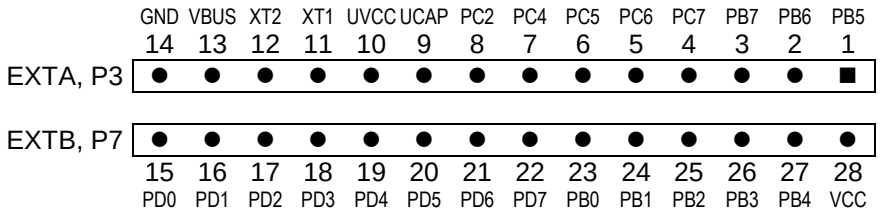


VCC presence, no matter what the power source is, is indicated by the green VCC ON LED (D1).

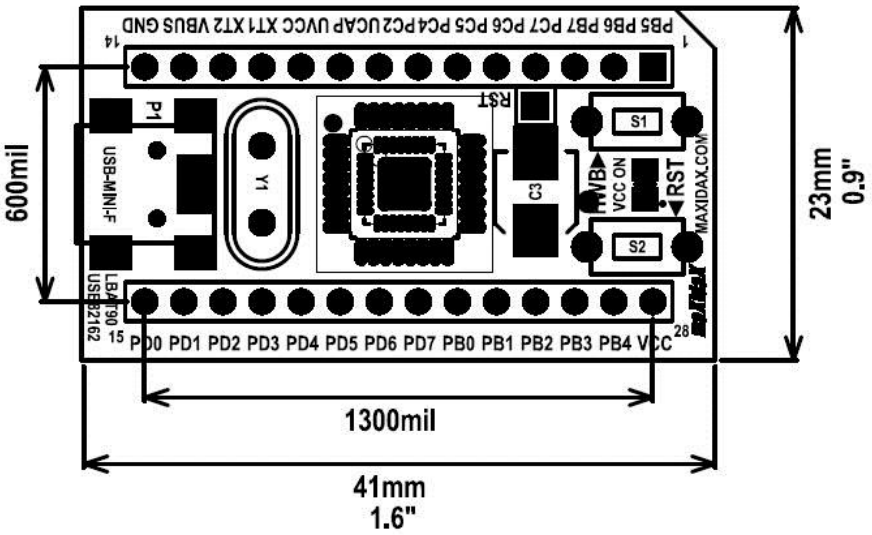
2.2.7. Extension headers EXTA, EXTB (P3, P4)

Extension headers provide connection to other devices and circuitry allowing access to all MCU I/O pins and other signals. EXTA and EXTB fit any 100mil/2.54mm linear connector, thus providing connectivity with virtually any 100mil/2.54mm universal or breadboard. Direct interfacing is of course also possible. Headers length is 1300mil/30.48mm, and spacing between them is 600mil/15.24mm (see BOARD LAYOUT AND DIMENSIONS).

Extension headers pinout:



3. BOARD LAYOUT AND DIMENSIONS



□

Disclaimer: All information in this document is provided in connection with embeddedglow.com 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 embeddedglow.com products. **EXCEPT AS SET FORTH IN EMBEDDEDGLOW.COM TERMS AND CONDITIONS OF SALE, WEBSITE OWNER ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO HIS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT WILL WEBSITE OWNER 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.**

Website owner 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. Website owner does not make any commitment to update the information contained herein.

Embeddedglow.com products are not intended, authorized, or suitable for, and shall not be used in automotive applications and/or as components in applications intended to support or sustain life.

Atmel® logo and combinations thereof, and AVR® are registered trademarks or trademarks of Atmel Corporation or its subsidiaries.

Windows® is a registered trademark of Microsoft Corporation.

*As this document provides links to web pages and/or resources, the inclusion of those links is for reader's convenience only, and should not be interpreted as an endorsement of the owner/sponsor of the site or the content of the site.