

PIC32-PINGUINO development board User's Manual



All boards produced by Olimex are ROHS compliant

Revision C, March 2012
Designed by OLIMEX Ltd, 2011

INTRODUCTION:

What is Arduino?

Arduino is an open-source electronics prototyping platform, designed to make the process of using electronics in multidisciplinary projects more accessible. The hardware consists of a simple open hardware design for the Arduino board with an Atmel AVR processor and on-board I/O support. The software consists of a standard programming language and the boot loader that runs on the board.

Arduino hardware is programmed using a Wiring-based language (syntax + libraries), similar to C++ with some simplifications and modifications, and a Processing-based IDE.

The project began in Ivrea, Italy in 2005 to make a device for controlling student-built interaction design projects less expensively than other prototyping systems available at the time. As of February 2010 more than 120,000 Arduino boards had been shipped. Founders Massimo Banzi and David Cuartielles named the project after a local bar named Arduino. The name is an Italian masculine first name, meaning "strong friend". The English pronunciation is "Hardwin", a namesake of Arduino of Ivrea

More information could be found at the creators web page <http://arduino.cc/> and in the Arduino Wiki <http://en.wikipedia.org/wiki/Arduino>

To make the story short - Arduino is easy for the beginners with lack of Electronics knowledge, but also do not restrict the professionals as they can program it in C++ or mix of Arduino/C++ language.

There are thousands of projects which makes the startup easy as there is barely no field where Arduino enthusiasts to have not been already.

Arduino has inspired two other major derivatives - MAPLE and PINGUINO. Based on 8-bit AVR technology the computational power of Arduino boards are modest, this is why team from MIT developed **MAPLE** project which is based on ARM7 STM32F103RBT6 microcontroller, the board have same friendly IDE as Arduino and offers the same capabilities as hardware and software but runs the Arduino code much faster. Maple project can be found at <http://leaflabs.com>

In parallel with Arduino another project was started called **PINGUINO**. This project choose the first implementation to be with PIC microcontrollers, the reason was that AVRs were hard to find in some parts of the world like South America so you will see lot of **PINGUINO** developers are from there. **PINGUINO** project founders decided to go with Python instead Java for processing language. For the moment **PINGUINO** is much more flexible than Arduino as not limited to 8bit, currently the IDE which have GCC in background can support 8-bit PIC microcontrollers, 32bit PIC32 (MIPS) microcontrollers and ARM7/CORTEXM3 microcontrollers which makes **PINGUINO** very flexible as once you make your project you can migrate easily through different hardware platforms and not being connected to single microcontroller manufacturer. The PINGUINO project can be found at <http://www.pinguino.cc>

BOARD FEATURES:

We entered the Arduino/MAPLE field 5 years after the design was introduced, and this allowed us to see and skip all the errors and missteps the Arduino inventors made :-)

We had the possibility to read current customer feedback and to implement what they wanted to see in the original Arduino.

1. Original Arduino/MAPLE uses linear power supply, this limits the input voltage range. We designed the power supply to accept power from 9 to 30V DC thus making possible to take virtually any power supply adapter on the market, also enable application which are in industrial power supply 24VDC.
2. We carefully selected all components to work reliable in INDUSTIRAL temperature range -25+85C so the board can be used in INDUSTIRAL applications while the original design is to Commercial 0-70C operating temperature.
3. The original Arduino/MAPLE design is not good for portable applications as consumes too much power with the linear vltage regulators, we put ULTRA LOW POWER voltage regulators and the consumption is only few microamps, which enables handheld and battery powered applications.
4. We add Li-Ion rechargeable battery power supply option with BUILD-IN on board charger, so when you attach battery it is automatically charged and kept in this state until the other power source (USB or external adapter) is removed and it AUTOMATICALLY will power the board - no jumpers, no switches!
5. Our board have UEXT connector which allow many existing modules like RF, ZIGBEE, GSM, GPS to be connected.
6. Our desing allow RTC - Real Time Clock.
7. We made our design noise immune.
8. Optionally if someone need higher precision and temperature stability in Analog reading we have provision on the board for Aref preciese source.
9. The LEDs and the BUTTONs are on the edge of the board so there is easy access even if the boards have shields on them.
10. All components are LOWER than the connectors, so the shields do not interference with them.
11. mini USB connector is used which is common and used in most cell phones, so you do not have to buy other cables
12. Original Arduino design had flaw and the connectors were not spaced at 0.1" this make perfo board use impossible, to keep the compatibility we have same spacing but we add next to this connector on 0.1" which customer can use with perforated boards.
13. All signals on the connectors are printed on top and on bottom of the board, so when you check with probe you know exactly which port you are measuring.
14. 4 mount holes make board attachment easier

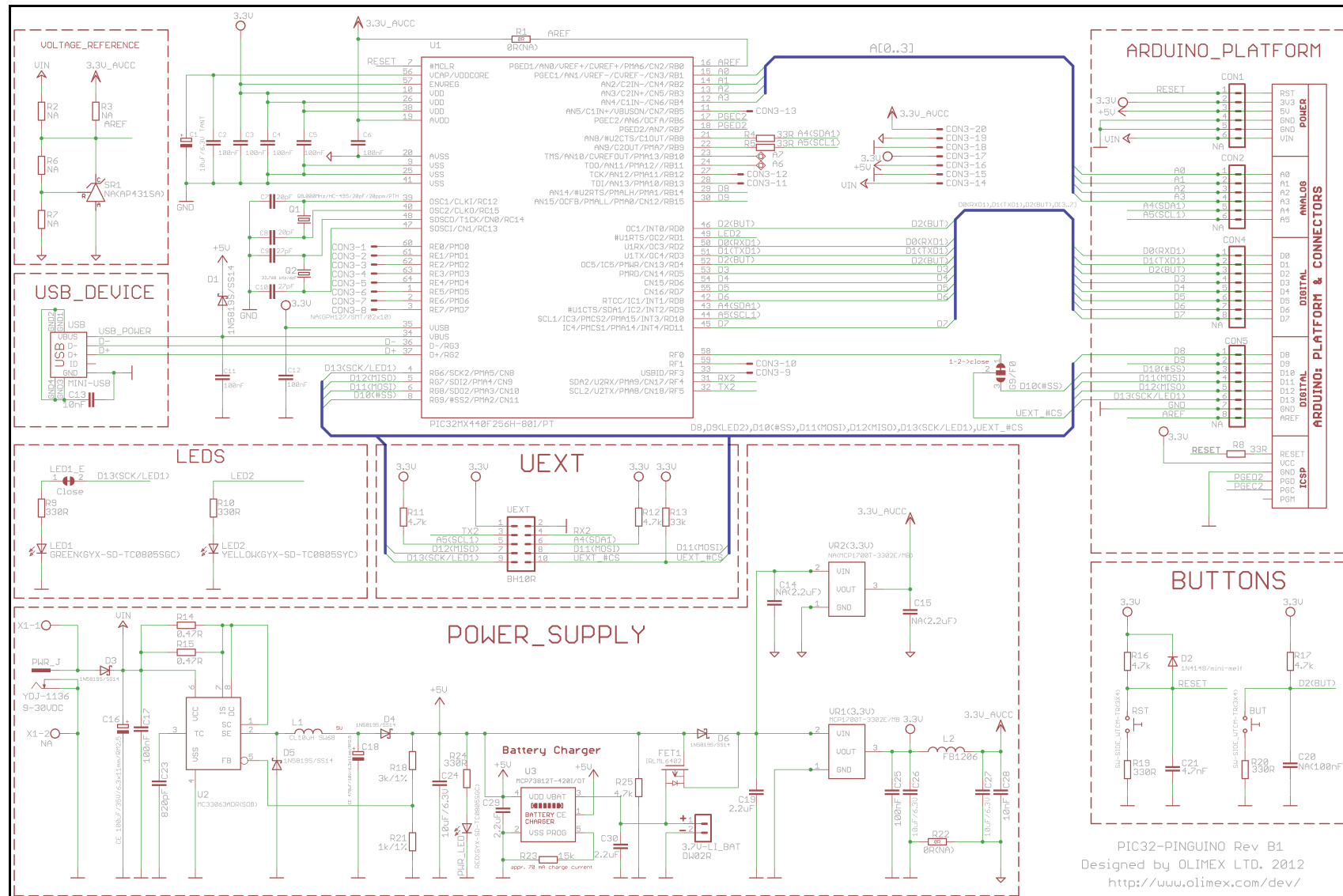
ELECTROSTATIC WARNING:

The PIC32-PINGUINO board is shipped in protective anti-static packaging. The board must not be subject to high electrostatic potentials. General practice for working with static sensitive devices should be applied when working with this board.

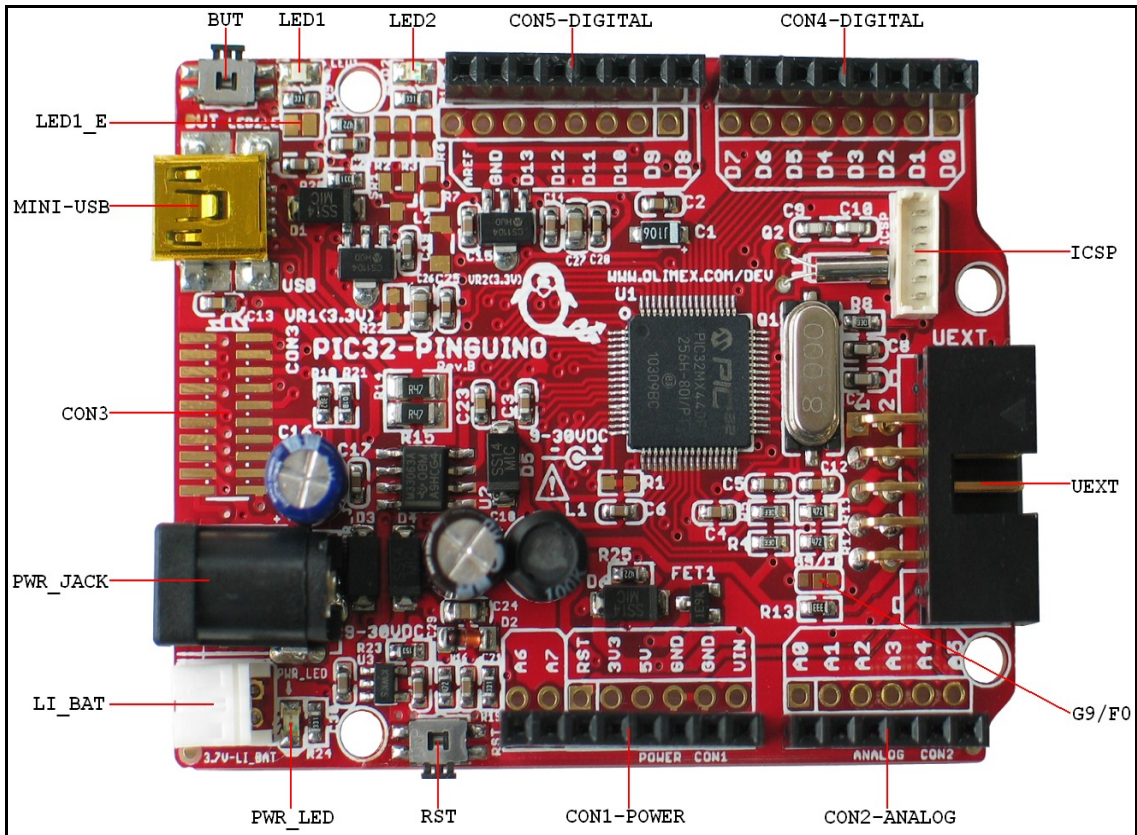
BOARD USE REQUIREMENTS:

- Cables:** mini USB cable
For programming via ICSP connector you will need PIC-ICSP connector and USB A-B cable for PIC-ICD2-POCKET, PIC-Kit3.
- Hardware:** Programmer/Debugger - PIC-ICD2, PIC-ICD2-POCKET, PIC-ICD2-TINY, PIC-Kit3, or other compatible programming/debugging tool.
- !!!Warning!!!** When you want to program this microcontroller with PICICD2, PIC-ICD2-POCKET or PIC-ICD2-TINY, before connecting the programmer to your target board, you should first connect the programmer to your computer and open MPLAB. There, first from menu Configure - Select Device - choose the microcontroller you are about to program, then from menu Programmer - Select Programmer - choose MPLAB ICD 2, wait while MPLAB is downloading operation system, and after ICD2 is connected - check in menu Programmer - Settings - Power - there is option - Power target circuit from MPLAB ICD 2 - this option should be forbidden, you could not select it. Now it is safe to connect the programmer to your target board.

SCHEMATIC:



BOARD LAYOUT:



POWER SUPPLY CIRCUIT:

PIC32-PINGUINO can take power supply from:

- external power supply (9-30) VDC.
- + 5V from USB
- 3.7 V Li-ion battery

The programmed board power consumption is about 60 mA with all peripherals enabled.

RESET CIRCUIT:

PIC32-PINGUINO reset circuit includes D2 (1N4148), R16 (4.7k Ω), R19 (330 Ω), C21 (4.7nF), PIC32MX440F256H pin 7 (#MCLR) and RESET button.

CLOCK CIRCUIT:

Quartz crystal **Q1** 8 MHz is connected to PIC32MX440F256H pin 39 (OSC1/CLKI/RC12) and pin 40 (OSC2/CLKO/RC15).

Quartz crystal **Q2** 32.768 kHz is connected to PIC32MX440F256H pin 47 (SOSCI/CN1/RC13) and pin 48 (SOSCO/T1CK/CN0/RC14).

JUMPER DESCRIPTION:

LED1_E



This jumper, when closed, enables LED1.

Default state is closed.

G9/F0



This jumper, when is in position G9 – connects UEXT pin 10 (UEXT_#CS) to CON5 pin 3 (D10(#SS)) and when is on position F0 – connects UEXT pin 10 (UEXT_#CS) to PIC32MX440F256H pin 58 (RF0).

Default state is in position F0.

INPUT/OUTPUT:

Status Led with name **LED1 (green)** connected via jumper LED1_E to PIC32MX440F256H pin 4 (RG6) – signal D13(SCK/LED1).

Status Led with name **LED2 (yellow)** connected to PIC32MX440F256H pin 49 (#U1RTS/OC2/RD1).

Power-on LED (red) with name **PWR_LED** – this LED shows that the board is power supplied.

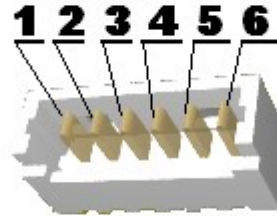
User button with name **BUT** connected to **PIC32MX440F256H** pin 46 (OC1/INT0/RD0) and pin 52 (OC5/IC5/PMWR/CN13/RD4) – signal D2(BUT).

User button with name **RST** connected to **PIC32MX440F256H** pin 7 (#MCLR).

EXTERNAL CONNECTORS DESCRIPTION:

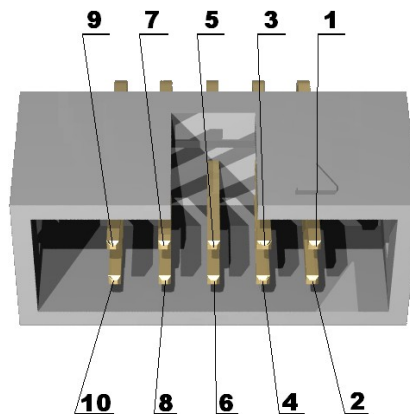
ICSP:

Pin #	Signal Name
1	RESET
2	+3.3V
3	GND
4	PGED2
5	PGEC2
6	Not connected



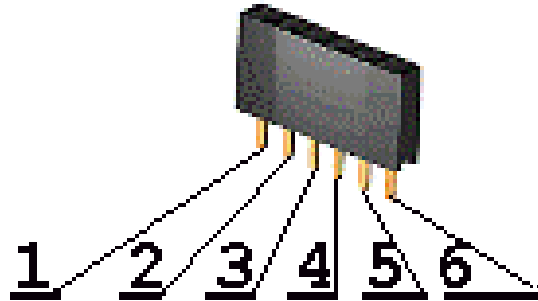
UEXT:

Pin #	Signal Name
1	+3.3V
2	GND
3	TX2
4	RX2
5	A5(SCL1)
6	A4(SDA1)
7	D12(MISO)
8	D11(MOSI)
9	D13(SCK/LED1)
10	UEXT_#CS



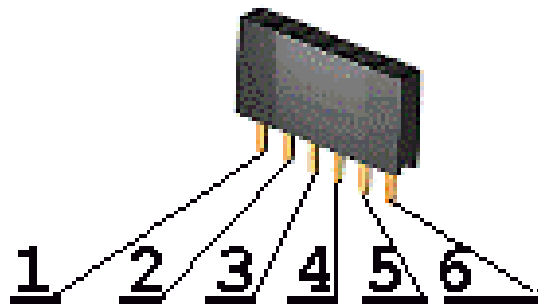
CON1 - POWER:

Pin #	Signal Name
1	RESET
2	+3.3V
3	+5V
4	GND
5	GND
6	VIN



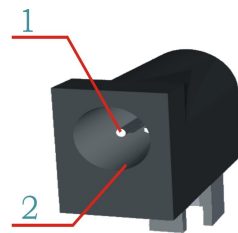
CON2 - ANALOG:

Pin #	Signal Name
1	A0
2	A1
3	A2
4	A3
5	A4(SDA1)
6	A5(SCL1)



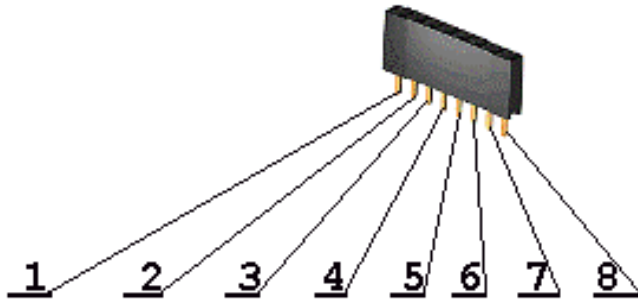
PWR JACK:

Pin #	Signal Name
1	Power Input
2	GND



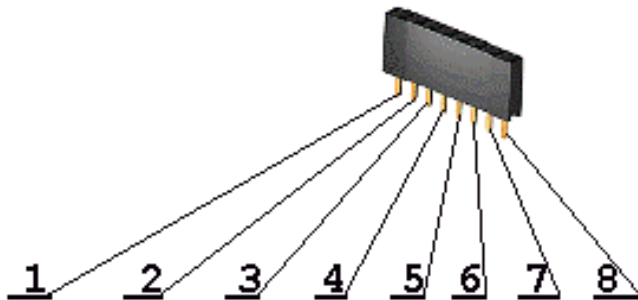
CON4 - DIGITAL:

Pin #	Signal Name
1	D0(RXD1)
2	D1(TXD1)
3	D2(BUT)
4	D3
5	D4
6	D5
7	D6
8	D7



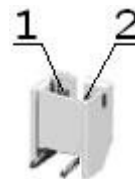
CON5 - DIGITAL:

Pin #	Signal Name
1	D8_MMC_#SS
2	D9
3	D10(#SS)
4	D11(MOSI)
5	D12(MISO)
6	D13(SCK/LED1)
7	GND
8	AREF



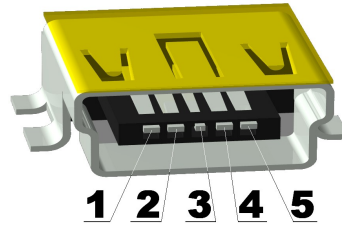
LI BAT:

Pin #	Signal Name
1	VBAT
2	GND

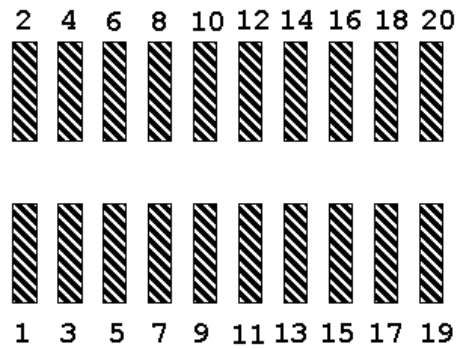


MINI-USB:

Pin #	Signal Name
1	USB_POWER
2	D -
3	D +
4	Not connected
5	GND



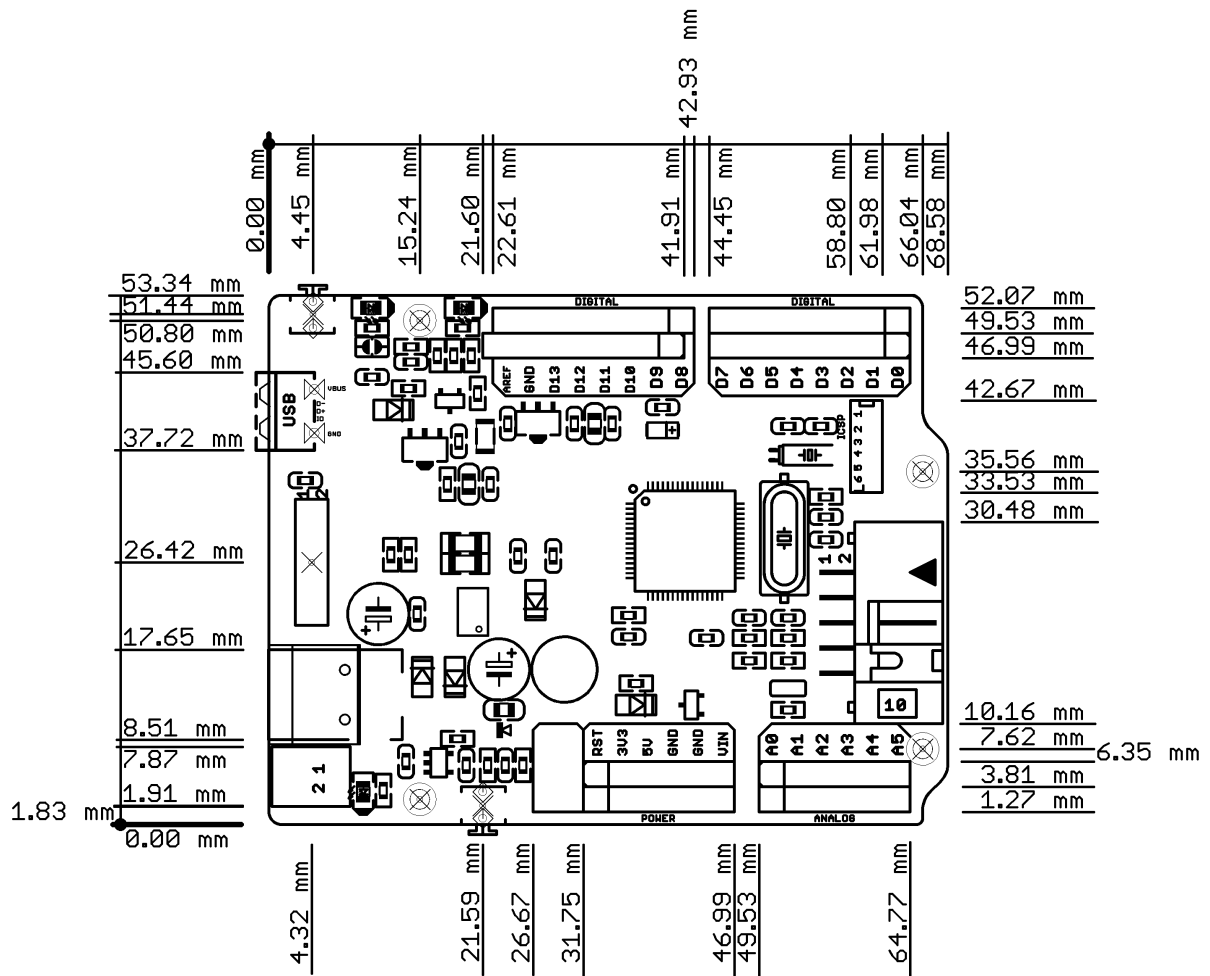
CON3:



Pin #	Signal Name	Pin #	Signal Name
1	RE0	2	RE1
3	RE2	4	RE3
5	RE4	6	RE5
7	RE6	8	RE7
9	LED2	10	RF1
11	TX2	12	RB12
13	RX2	14	VIN
15	GND	16	+5V
17	+3.3V	18	GND
19	AGND	20	VDD

Note: This connector is not mounted on the board.

MECHANICAL DIMENSIONS:



AVAILABLE DEMO SOFTWARE:

The software consists of boot loader that runs on the board and a simple blinking led project.

ORDER CODE:

PIC32-PINGUINO-Rev. B - assembled and tested board

How to order?

You can order to us directly or by any of our distributors.
Check our web www.olimex.com/dev for more info.

Revision history:

Board's revision

Rev. B, March 2011

Rev. B1, March 2012

- Removed C14 - 2.2uF, C15 - 2.2uF, VR2-MCP1700T-3302E/MB; Added L2-FB1206 – because of random hangs of the PIC32 in specific cases
- Removed C20 – was filtering not only the button but some of the higher frequencies which we decided to change

Manual's revision

Rev. A, August 2011

- At first page "Copyright(c) 2011, OLIMEX Ltd, All rights reserved" is replaces with "Designed by OLIMEX Ltd., 2011"
- In schematic "COPYRIGHT(C) 2011, OLIMEX Ltd." replaced with "DESIGNED BY OLIMEX LTD, 2011"

Rev. B, October 2011

- In "BOARD USE REQUIREMENTS" added more information about cables; "Hardware" and "Warning"

Rev.B1, March 2012

- Removed "- Separate voltage regulator for the Analog part, which allows the ADC to be read correctly without the digital noise pickup." from page 4
- Updated schematic with board version B1
- Various spelling changes and page formatting

Disclaimer:

© 2011 Olimex Ltd. Olimex®, logo and combinations thereof, are registered trademarks of Olimex Ltd. Other terms and product names may be trademarks of others.

The information in this document is provided in connection with Olimex products. No license, express or implied or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Olimex products.

Neither the whole nor any part of the information contained in or the product described in this document may be adapted or reproduced in any material from except with the prior written permission of the copyright holder.

The product described in this document is subject to continuous development and improvements. All particulars of the product and its use contained in this document are given by OLIMEX in good faith. However all warranties implied or expressed including but not limited to implied warranties of merchantability or fitness for purpose are excluded.

This document is intended only to assist the reader in the use of the product. OLIMEX Ltd. shall not be liable for any loss or damage arising from the use of any information in this document or any error or omission in such information or any incorrect use of the product.