Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of "Quality Parts,Customers Priority,Honest Operation,and Considerate Service",our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!
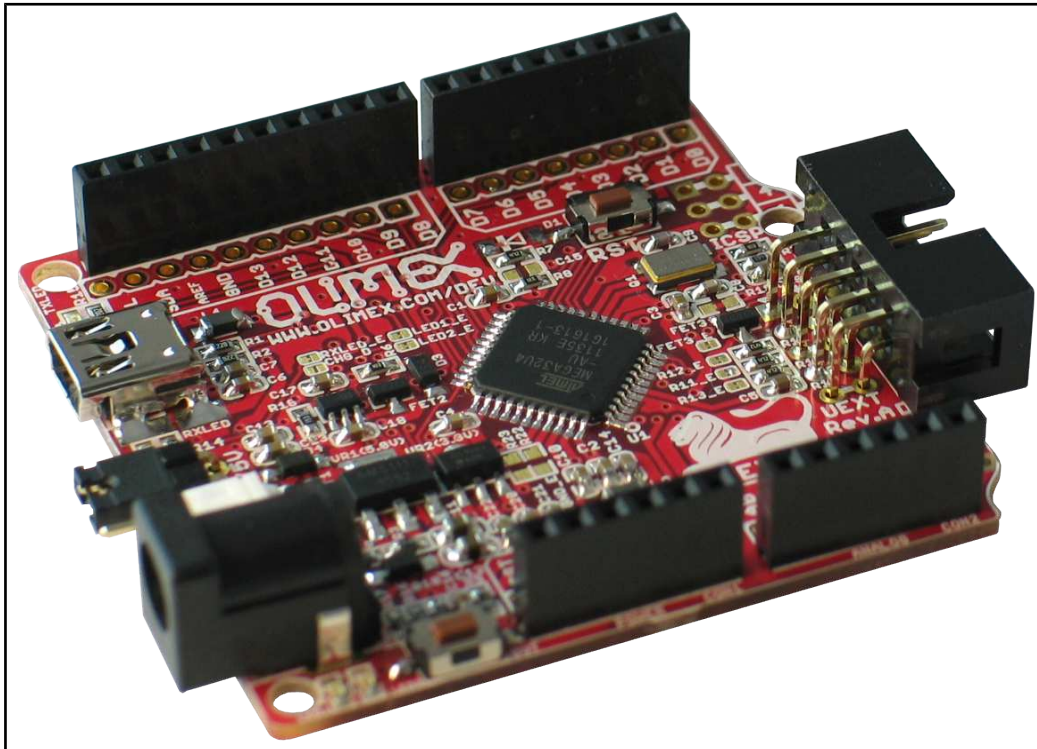

Contact us

OLIMEX

OLIMEXINO-32U4 development board
USER'S MANUAL

Revision H, April 2017
Designed by OLIMEX Ltd, 2013

lead-free

All boards produced by Olimex LTD are ROHS compliant

Table of Contents

CHAPTER 1 OVERVIEW

1. Introduction to the chapter

Thank you for choosing the OLIMEXINO-32U4 development board from Olimex! This document provides a User's Guide for the Olimex OLIMEXINO-32U4 development board. As an overview, this chapter gives the scope of this document and lists the board's features. The document's organization is then detailed.

The OLIMEXINO-32U4 development board enables code development of applications running on the ATmega32U4 microcontroller, manufactured by Atmel Corporation.

1.1 Features

- Leonardo design – no FTDI chip, the MCU directly handles the virtual COM
- ATMEGA32U4 microcontroller, all in one USB communication no need for external FTDI chip.
- Input power supply 7-12VDC
- ULTRA LOW POWER voltage regulators and the consumption is only few microamps, which enables hand-held and battery powered applications.
- 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!
- Works both on 3.3V and on 5V which is selectable with jumper, so 3.3V and 5V shields can be used
- UEXT connector which allow many existing modules like RF, ZIGBEE, GSM, GPS to be connected
- Allows RTC – Real Time Clock (Q1 can be replaced by RTC).
- NOISE IMMUNE design
- the LEDs and the button BUT are on the edge of the board so there is easy access even if the boards have shields on them
- all components are LOWER than the connectors, so the shields do not interference with them
- mini USB connector is used -it is common and used in most cell phones, so you don't have to buy other cables
- Original design had a flaw – the connectors were not spaced at 0.1" which made bread board use impossible, to keep the compatibility we have the original spacing but we also added a 0.1" connector which customer can use with bread boards (and jumper wires) saving the hustle of soldering
- All signals on the connectors are printed on top of the board, so when you check with probe you know exactly which port you are measuring
- 4 mount holes make board attachment easier

1.2 Similar boards

OLIMEXINO-32U4 is an ARDUINO-LEONARDO-like board with ATmega32U4.

It is a powerful board considering its low price but yet a starter board for the Arduino family.

The more powerful Arduino board we have in the range is OLIMEXINO-328 – based on Arduino Duemilanove. It has a more sophisticated supply circuit, USB OTG connector and there are more Arduino projects available.

1.3 Organization

Each section in this document covers a separate topic, organized as follow:
   − Chapter 1 is an overview of the board usage and features
   − Chapter 2 has a short introduction to Arduino platform and its derivatives Pinguino and Maple
   − Chapter 3 provides a guide for quickly setting up the board
   − Chapter 4 contains the general board diagram and layout
   − Chapter 5 describes the component that is the heart of the board: the ATmega32U4 microcontroller
   − Chapter 6 is an explanation of the control circuitry associated with the microcontroller to reset. Also shows the clocks on the board
   − Chapter 7 covers the connector pinout, peripherals and jumper description
   − Chapter 8 shows the processor diagram and memory map
   − Chapter 9 provides the schematics
   − Chapter 10 contains the revision history

CHAPTER 2 INTRODUCTION TO ARDUINO

2. What is Arduino?

Arduino is an open-source electronics prototyping platform, designed to make the process of using electronics in multidisciplinary projects easily 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 Integrated Development Environment (IDE).

The project began in Ivrea, Italy in 2005 aiming 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 beginners who lack Electronics knowledge, but also does not restrict professionals as they can program it in C++ or mix of Arduino/C++ language. There are thousands of projects which makes it easy to startup 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 is modest, this is why a team from MIT developed the MAPLE project which is based on ARM7 STM32F103RBT6 microcontroller. The board has same friendly IDE as Arduino and offers the same capabilities as hardware and software but runs the Arduino code much faster. The Maple project can be found at http://leaflabs.com

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

2.1 First steps with Arduino IDE

Depending when you purchased the board there are two scenarios since the PID (product ID) which is stored in the firmware and expected by the IDE is different between versions 1.0.0 and 1.0.1 (1.0.1 is the first release of the Arduino IDE with official support for Leonardo).

2.1.1 For purchases after the official release of Leonardo

Boards produced after the official release of Arduino Leonardo have the latest bootloader available with the adjusted bootloader PID. Download the latest Arduino IDE and extract the package. You can find the needed driver for the bootloader located in folder /drivers. You click update on the device in device manager and point the installer to the /drivers folder.

Then launch Arduino IDE and set the correct board and COM port in TOOLS menu.

From File->Open navigate to the examples folder which is found in the Arduino IDE installation folder. Choose a simple example to begin with like BlinkWithoutDelay.ino and open it. Click verify and after it compiles click "Upload". Voila, you have uploaded code to your Arduino device.

Now start exploring the world of Arduino! More info and help on the software can be found and the official Arduino web site: http://www.arduino.cc/.

2.1.2 For purchases before the official release of Leonardo

If you have purchased the OLIMEXINO-32U4 before the official release of the Arduino Leonardo board there are two scenarios for using Arduino IDE:

1) Download Arduino IDE 1.0 RC2 or Arduino IDE 1.0.0. When you extract the package check if the Arduino Leonardo section is the \hardware\arduino\boards is uncommented. If the section has number signs (#) infront remove the number signs (#).

2) If you want to use the latest version (which by the time of writing is 1.0.1) you would need to change to bootloader firmware. This will require an AVR programmer tool.

You can get the bootloader in .elf format from our web site. You can also use the original bootloader hex if you set the fuses and lock bits for the processor as follows:

low_fuses=0xff
high_fuses=0xd8
extended_fuses=0xcb

lock_bits (choose "SPM Prohibited in Boot Section") from the drop down menu which will set the bits either to EF or 2F)

After you have downloaded and extracted the desired IDE you need to install the driver for your operating system. You can find the needed driver for the bootloader located in folder /drivers. You click update on the device in device manager and point the installer to the /drivers folder.

Then launch Arduino IDE and set the correct board and COM port in TOOLS menu.

From File->Open navigate to the examples folder which is found in the Arduino IDE installation folder. Choose a simple example to begin with like BlinkWithoutDelay.ino and open it. Click verify and after it compiles click "Upload". Voila, you have uploaded code to your Arduino device.

Now start exploring the world of Arduino! More info and help on the software can be found and the official Arduino web site: http://www.arduino.cc/.

CHAPTER 3 SETTING UP THE OLIMEXINO-32U4 BOARD

3. Introduction to the chapter

This section helps you set up the OLIMEXINO-32U4 development board for the first time. Please consider first the electrostatic warning to avoid damaging the board, then discover the hardware and software required to operate the board. The procedure to power up the board is given, and a description of the default board behavior is detailed.

3.1 Electrostatic warning

OLIMEXINO-32U4 is shipped in a protective anti-static package. The board must not be exposed to high electrostatic potentials. A grounding strap or similar protective device should be worn when handling the board. Avoid touching the component pins or any other metallic element.

3.2 Requirements

In order to set up the OLIMEXINO-32U4, the following items are required:

If using the free ARDUINO IDE the ONLY requirement is a miniUSB to USB-A cable to connect to a computer.

If you want to upgrade the bootloader, or program the board without the ARDUINO-IDE there are pinouts for ICSP programmer (note that there isn't a connector mounted). You can use our AVR-ISP-MK2, or AVR-ISP500 programmer for custom programming without ARDUINO IDE.

Having an AVR programmer is highly advisable since the bootloader software is still not perfect and it is possible to destroy it only by software means (remap pins used for programming) which would require reinstalling the bootloader.

3.3 Powering the board

The OLIMEXINO-32U4 board is self-powered by the miniUSB if it is connected to a USB hub of a computer. Alternatively, it can be powered via the POWER JACK by 7Vdc to 12Vdc or via the 3.7V-LI_BAT connector by 3.7V battery.

The bootloader starts automatically.

On powering the board PWR_LED should turn on. RXLED and TXLED should turn on and monitor the send receive on the virtual COM port. LED1 and LED2 should start blinking.

3.4 Prebuilt software

The board comes with prebuilt ARDUINO bootloader. The different part between the other Arduinos and Leonardo is that Leonardo doesn't use FTDI chip nor external chip for the virtual COM port but it is handled by the ATmega32U4.

CHAPTER 4 OLIMEXINO-32U4 BOARD DESCRIPTION

4. Introduction to the chapter

Here you get acquainted with the main parts of the board. Note the names used on the board differ from the names used to describe them. For the actual names check the OLIMEXINO-32U4 board itself.

4.1 Layout (top view)
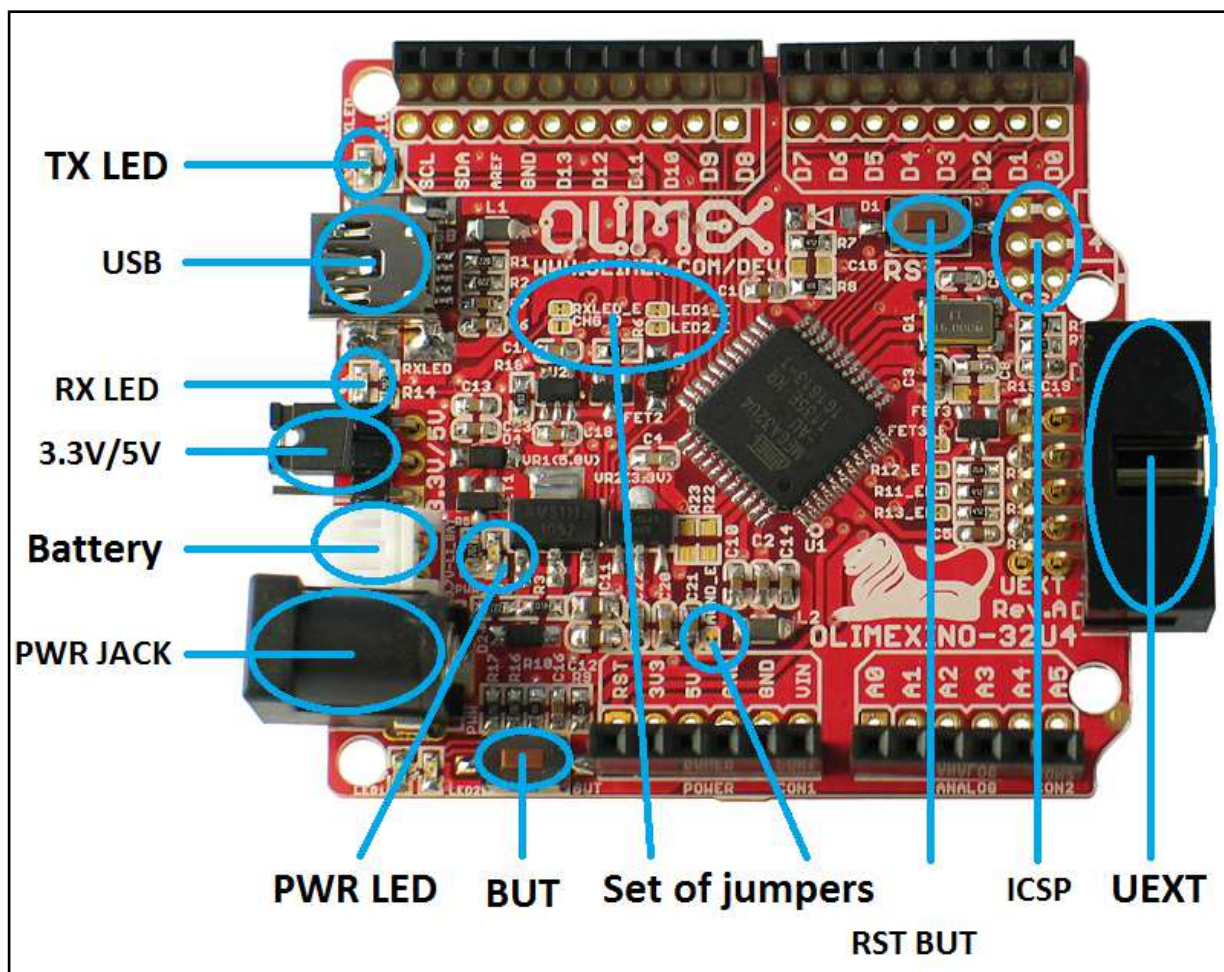
CHAPTER 5 THE ATMEGA32U4 MICROCONTROLLER

5. Introduction to the chapter

In this chapter is located the information about the heart of OLIMEXINO-32U4 – its microcontroller. The information is a modified version of the datasheet provided by its manufacturers.

5.1 The microcontroller's features

• High Performance, Low Power AVR® 8-Bit Microcontroller
• Advanced RISC Architecture
    – 135 Powerful Instructions – Most Single Clock Cycle Execution
    – 32×8 General Purpose Working Registers
    – Fully Static Operation
    – Up to 16 MIPS Throughput at 16 MHz
    – On-Chip 2-cycle Multiplier
• Non-volatile Program and Data Memories
    – 32K Bytes of In-System Self-Programmable Flash
    – 2.5K Bytes Internal SRAM
    – 1K Bytes Internal EEPROM
    – Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
    – Data retention: 20 years at 85°C/ 100 years at 25°C(1)
    – Optional Boot Code Section with Independent Lock Bits
        In-System Programming by On-chip Boot Program
        True Read-While-Write Operation
        All supplied parts are preprogramed with a default USB bootloader
    – Programming Lock for Software Security
• JTAG (IEEE std. 1149.1 compliant) Interface
    – Boundary-scan Capabilities According to the JTAG Standard
    – Extensive On-chip Debug Support
    – Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
• USB 2.0 Full-speed/Low Speed Device Module with Interrupt on Transfer Completion
    – Complies fully with Universal Serial Bus Specification Rev 2.0
    – Supports data transfer rates up to 12 Mbit/s and 1.5 Mbit/s
    – Endpoint 0 for Control Transfers: up to 64-bytes
    – 6 Programmable Endpoints with IN or Out Directions and with Bulk, Interrupt or Isochronous Transfers
    – Configurable Endpoints size up to 256 bytes in double bank mode
    – Fully independent 832 bytes USB DPRAM for endpoint memory allocation
    – Suspend/Resume Interrupts
    – CPU Reset possible on USB Bus Reset detection
    – 48 MHz from PLL for Full-speed Bus Operation
    – USB Bus Connection/Disconnection on Microcontroller Request
    – Crystal-less operation for Low Speed mode
• Peripheral Features

   – On-chip PLL for USB and High Speed Timer: 32 up to 96 MHz operation
   – One 8-bit Timer/Counter with Separate Prescaler and Compare Mode
   – Two 16-bit Timer/Counter with Separate Prescaler, Compare- and Capture Mode
   – One 10-bit High-Speed Timer/Counter with PLL (64 MHz) and Compare Mode
   – Four 8-bit PWM Channels
   – Four PWM Channels with Programmable Resolution from 2 to 16 Bits
   – Six PWM Channels for High Speed Operation, with Programmable Resolution from
   2 to 11 Bits
   – Output Compare Modulator
   – 12-channels, 10-bit ADC (features Differential Channels with Programmable Gain)
   –   Programmable Serial USART with Hardware Flow Control
   –   Master/Slave SPI Serial Interface
   – Byte Oriented 2-wire Serial Interface
   – Programmable Watchdog Timer with Separate On-chip Oscillator
   – On-chip Analog Comparator
   – Interrupt and Wake-up on Pin Change
   – On-chip Temperature Sensor
• Special Microcontroller Features
   – Power-on Reset and Programmable Brown-out Detection
   – Internal 8 MHz Calibrated Oscillator
   – Internal clock prescaler & On-the-fly Clock Switching (Int RC / Ext Osc)
   – External and Internal Interrupt Sources
   – Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and
   Extended Standby
• I/O and Package
   – All I/O combine CMOS outputs and LVTTL inputs
   – 26 Programmable I/O Lines
   – 44-lead QFN Package, 7×7mm
• Operating Voltages
   – 2.7 – 5.5V
• Operating temperature
   – Industrial (-40°C to +85°C)
• Maximum Frequency
   – 8 MHz at 2.7V – Industrial range
   – 16 MHz at 4.5V – Industrial range

For comprehensive information on the microcontroller visit the Atmel web page for a datasheet.

At the moment of writing the microcontroller datasheet can be found at the following link:
http://www.atmel.com/Images/doc7766.pdf

CHAPTER 6 CONTROL CIRCUITY

6. Introduction to the chapter

Here you can find information about reset circuit, power circuit and quartz crystal locations.

6.1 Reset

OLIMEXINO-32U4 reset circuit includes R7 (4.7kΩ), R8 (330Ω), D1 (Shottky diode), ATmega32U4 pin 13 (#RESET) and a RESET button. Note that it also can be found at the ICSP header pin 5 and power connector pin 1.

6.2 Clock

16MHz quartz crystal Q1 is found at pins 16 and 17 of the processor.

By default OLIMEXINO-32U4 operates at 5V, which allows the main microcontroller (ATmega32u4) to operate properly at the maximum clock speed (16 MHz). The ATmega32u4 would work properly at 16MHz only if powered by at least 4.5V.

If you decide to change the position of the jumper 3.3V/5V jumper to 3.3V position, please note that, at 3.3V, the maximum frequency that can be achieved with ATmega32u4 is around 10MHz, thus the 16MHz operation is a significant overclock. Yet, our tests proved that the board works reliably with a 16MHz quartz even at 3.3V.

IMPORTANT: If the board has a quartz crystal rotated at 45 degrees relative to the pads provided do not panic. This is normal. We have two types of such crystals – one of them requires 4 pads, the other only 2 pads. That is why we have provided 4 pads to be able to fit both crystals. All boards Olimex manufactures pass automatized optical inspection after assembly and obvious misplacement like this is impossible to occur.

CHAPTER 7 HARDWARE

7. Introduction to the chapter

In this chapter are presented the connectors that can be found on the board all together with their pinout. Jumpers functions are described. Notes and info on specific peripherals are presented. Notes regarding the interfaces are given.

7.1 Battery connector

The battery connector is suitable for 3.7V LiPo batteries. The board has a built-in charger circuit, so when the main power supply is restored the battery would get recharged automatically.

Pin #	Signal Name
1	VBAT
2	GND
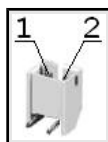
7.2 ICSP pin holes

The 6 pin ICSP header provides interface for custom programming/debugging outside of the Arduino IDE. Note the pin holes doesn't have a connector mounted over – you have to mount it/solder it yourself. The pinout can be found in the table below. It is important to check the jumper configuration for proper use of the ICSP pin hole.

ICSP pin holes			
Pin #	Signal name	Pin #	Signal name
1	D14(MISO)	4	D16(MOSI)
2	Vcc	5	RESET
3	D15(SCK)	6	GND

7.3 UEXT

OLIMEXINO-32U4 board has UEXT connector and can interface Olimex's UEXT modules.
For more information on our UEXT modules please visit:

https://www.olimex.com/Products/Modules/UEXT/

Pin #	Signal name
1	VDD out (3.3V or 5V)
2	GND
3	D1(TXD)
4	D0(RXD)
5	D3(SCL)
6	D2(SDA)
7	D14(MISO)
8	D16(MOSI)
9	D15(SCK)
10	D13(UEXT_#CS)
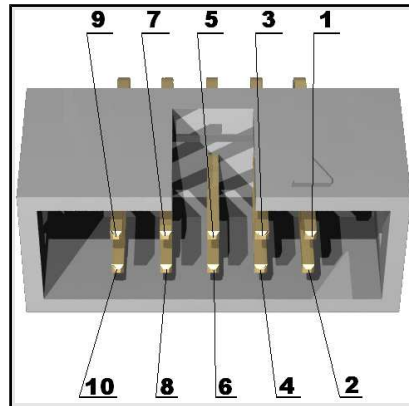
IMPORTANT: Be careful with the voltage output available at pin #1. The on-board 3.3V/5V switch also controls the voltage output available at pin #1 of the UEXT. Do not connect Olimex modules when the switch is set to 5V position, unless you are 100% sure that these UEXT modules work fine at 5V (most Olimex modules can not work properly at 5V, only at 3.3V)

It is important to note that there are 4.7k pull-up resistors on the I2C lines – SCL/SDA (D3 and D2 respectively). You can disconnect/disable the resistors by cutting between the pads of the small SMT jumpers R11_E and R12_E.

The chip select pin – CS (D13) also has a 4.7k pull-up resistor which might be disabled by cutting R13_E.

7.4 Arduino shield pin holes

For your convenience the pads are named individually near each of them. Please take extra care about the numbering but consider that there might be offset.

Pad Name	Signal	Pad Name	Signal
POWER CON1		DIGITAL CON2	
RST	RESET	A0	PF7/ADC7/TDI
3V3	+3.3 V	A1	PF6/ADC6/TDO
5V	+5 V	A2	PF5/ADC5/TMS
GND	GROUND	A3	PF4/ADC4/TCK
GND	GROUND	A4	PF1/ADC1
VIN	V in	A5	PF0/ADC0
DIGITAL CON3		DIGITAL CON4	
D0(RXD)	PD2/RXD1/INT2	D8(UEXT_PWR_E)	RB4/PCINT4/ADC11
D1(TXD)	PD3/TXD1/INT3	D9(LED2)	PB5/PCIN5/OC1A/#OC4B/ADC12
D2(SDA)	PD1/SDA/INT1	D10	PB6/PCINT6/OC1B/OC4B/ADC13
D3(SCL)	PD0/OC0B/SCL/INT0	D11	PB7/PCINT7/OC0A/OC1C/#RTS
D4	PD4/ICP1/ADC8	D12	PD6/T1/#OC4D/ADC9
D5	PC6/OC3A/#OC4A	D13(UEXT_#CS)	PC7/ICP3/CLK0/OC4A
D6	PD7/T0/OC4D/ADC10	GND	GND
D7(LED1)	INT6/AIN0/PE6	AREF	AREF
		SDA	D2(SDA)
		SCL	D3(SCL)

7.5 USB mini connector

Note that the USB is configured only as device.

Pin #	Signal Name
1	+5V
2	D-
3	D+
4	Not connected
5	GND
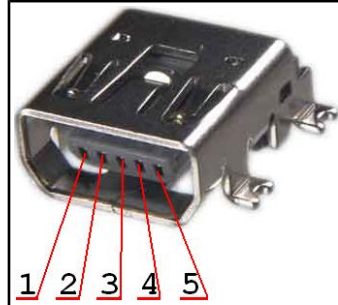
7.6 Jumper description

3.3V/5V

This jumper controls whether the board is powered by 3.3Vdc source or 5Vdc. It is provided to ensure compatibility with both original 5V shields and 3.3V modules.

Important! Changing this jumper also changes the voltage available at pin #1 of UEXT and also the voltage levels of all data signals available there (UART, SPI, I2C). Most Olimex-manufactured UEXT modules work at 3.3V – they would get damaged if the jumper is set 3.3V. When using Olimex MOD- board on this UEXT ensure the jumper is set to 3.3V, else you might damage the module.

Default position is 2-3 – 5Vdc.

AGND_E

When closed enables analog GND. SMD jumper.

Default state is closed.

FET3_E

When closed connects D8 from Arduino shield connector to FET3 so you can control the power on the UEXT by software means.

When open D8 becomes GPIO. This is a SMD type of jumper.

Default state is closed.

LED1_E

When closed enables LED1. SMD jumper.

Default state is closed.

LED2_E

When closed enables LED2. SMD jumper.

Default state is closed.

RXLED_E

When closed enables RXLED. SMD jumper.

Default state is closed.

TXLED_E

When closed enables TXLED. SMD jumper.

Default state is closed.

R11_E, R12_E, R13_E

When these jumpers are closed they enable 4.7k Ohm pull-up resistors on each of the I2C lines (R11_E is responsible for the resistor of SCL line, R12_E – for the SDA line, R13_E – for the UEXT_CS). SMD jumpers.

Default state for all three is closed.

7.7 Additional hardware components

The components below are mounted on the OLIMEXINO-32U4 board but are not discussed above. They are listed here for completeness:

General-purpose button

2 general purpose LEDs

2 LEDs showing virtual port activity

Note that connectors on the CON1, CON2, CON3, CON4 are soldered/mounted.

CHAPTER 8 MEMORY AND BLOCK DIAGRAM

8. Introduction to the chapter

Below is located the block diagram of the processor and on the next page you can find a memory map for this family of processors. It is strongly recommended to refer to the original datasheet released by Atmel for ones of higher quality.

Figure 2-1. Block Diagram
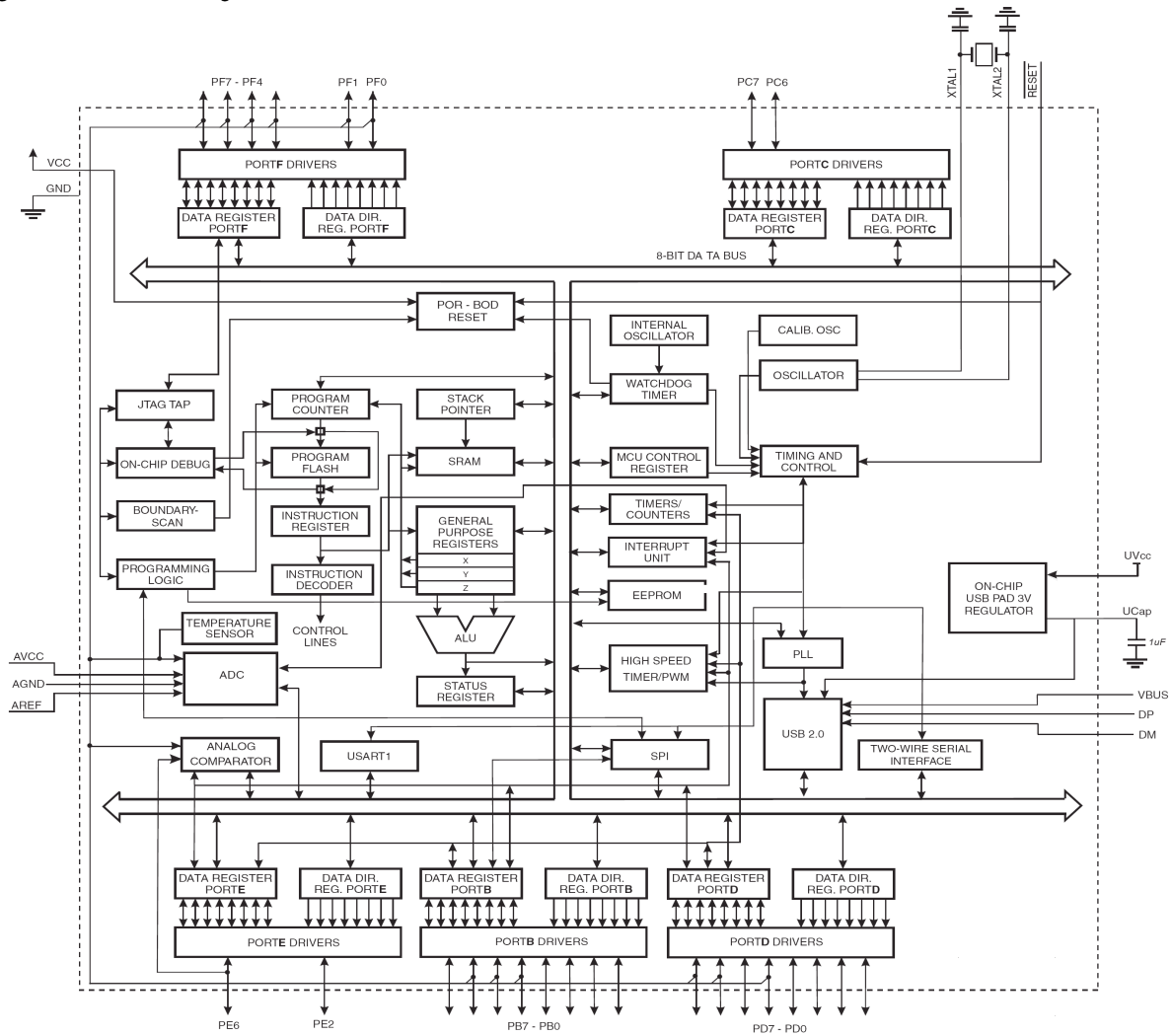
8.1 Memory organization

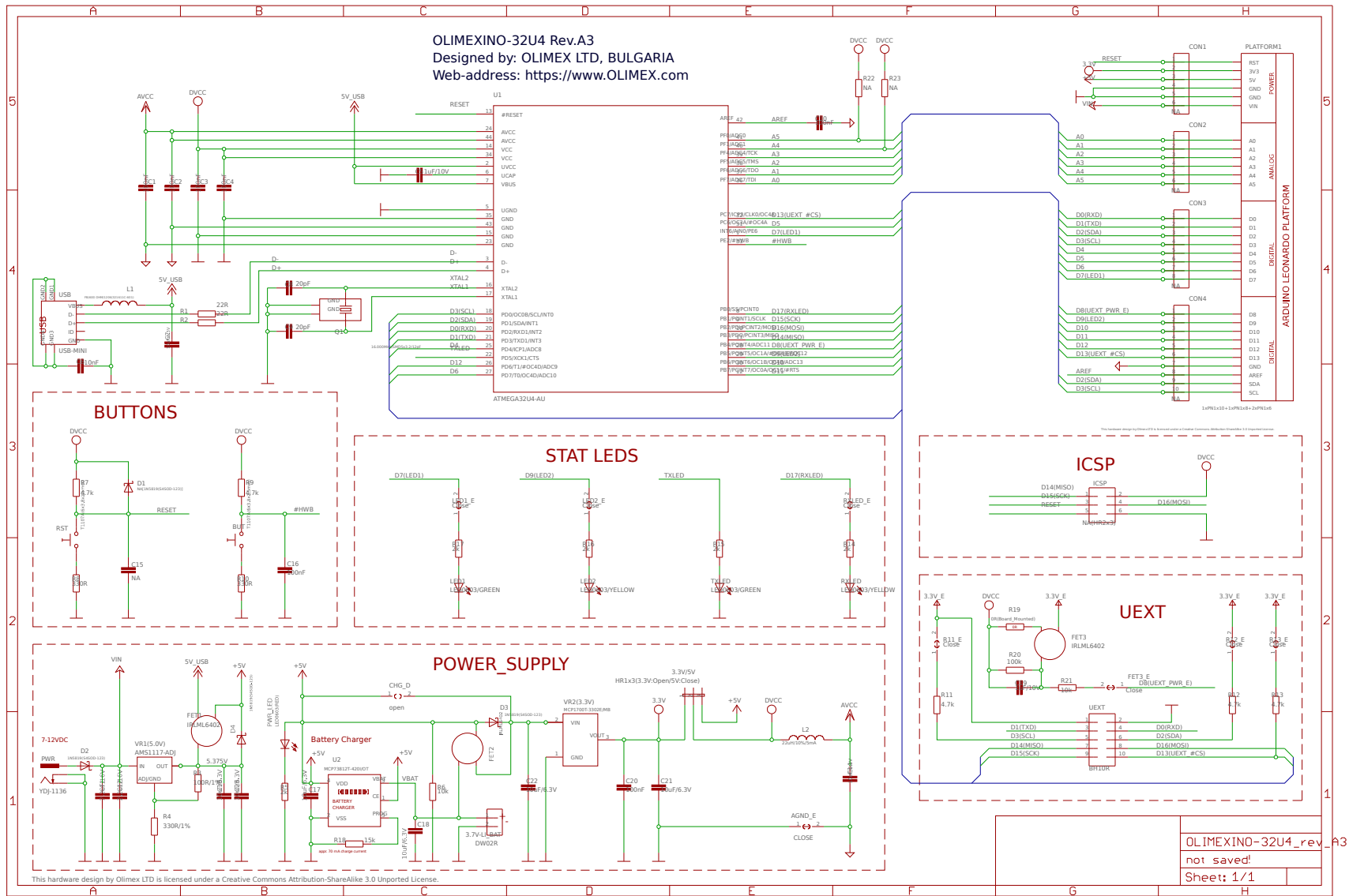| Table 5-1.	Memory Mapping.		
Memory		**Mnemonic**	**ATmega32U4**
Flash	Size	Flash size	32K bytes
	Start Address	- 0x0000	
	End Address	Flash end	0x7FFF[1] 0x3FFF[2]
32 Registers	Size	-	32 bytes
	Start Address	-	0x0000
	End Address	-	0x001F
I/O Registers	Size	-	64 bytes
	Start Address	-	0x0020
	End Address	-	0x005F
Ext I/O Registers	Size	-	160 bytes
	Start Address	-	0x0060
	End Address	-	0x00FF
Internal SRAM	Size	ISRAM size	2,5K bytes
	Start Address	ISRAM start	0x100
	End Address	ISRAM end	0x0AFF
External Memory	Not Present.		
EEPROM	Size	E2 size	1K bytes
	End Address	E2 end	0x03FF
Notes: 1. Byte address.			
2. Word (16-bit) address.			

CHAPTER 9 SCHEMATICS

9. Introduction to the chapter

In this chapter are located the schematics describing logically and physically OLIMEXINO-32U4.

9.1 Eagle schematic

OLIMEXINO-32U4 schematic is visible for reference here. You should zoom the pdf for a better view and also string search is available. You can also find a stand-alone pdf on the web page for OLIMEXINO-32U4 at our site: https://www.olimex.com/Products/Duino/AVR/OLIMEXINO-32U4/. They are located in HARDWARE section.

This work is licensed under the Creative Commons Attribution-ShareAlike 3.0 Unported License. To view a copy of this license, visit http://creativecommons.org/licenses/by-sa/3.0/. The EAGLE schematic is situated below for quicker reference.

OLIMEXINO-32U4 Rev.A3
Designed by: OLIMEX LTD, BULGARIA
Web-address: https://www.OLIMEX.com

BUTTONS

STAT LEDS

ICSP

POWER_SUPPLY

UEXT

OLIMEXINO-32U4_rev_A3
not saved!
Sheet: 1/1

This hardware design by Olimex LTD is licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported License.

9.2 Physical dimensions

Note that all dimensions are in inches.

CHAPTER 10 REVISION HISTORY

10. Introduction to the chapter

In this chapter you will find the current and the previous version of the document you are reading. Also the web-page for your device is listed. Be sure to check it after a purchase for the latest available updates and examples.

10.1 Document revision

Revision	Changes	Modified Pages
Initial	Initial release	All
B	Manual adjusted for Arduino IDE version 1.0.1 Fixed links	8, 9, 15
C	Adjusted formatting, fixed some mistakes, updated links	All
D	Added info about the I2C Updated disclaimer and warranty pages ICSP and UEXT connector changes due to new schematics Adjusted formatting, fixed some mistakes	2, 15, 16, 17, 20, 22, 26, All
E	Fixed - "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" - not true for RST button Fixed - "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" - not true with OLIMEXINO-32u4 (only on component side) Fixed - "Proto area is shown." - there is no proto are Fixed -"programming/debugging outside of the PINGUINO IDE" - should be ARDUINO IDE Fixed "Also note that the board comes with no	5, 14, 15, 17, 25