



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!



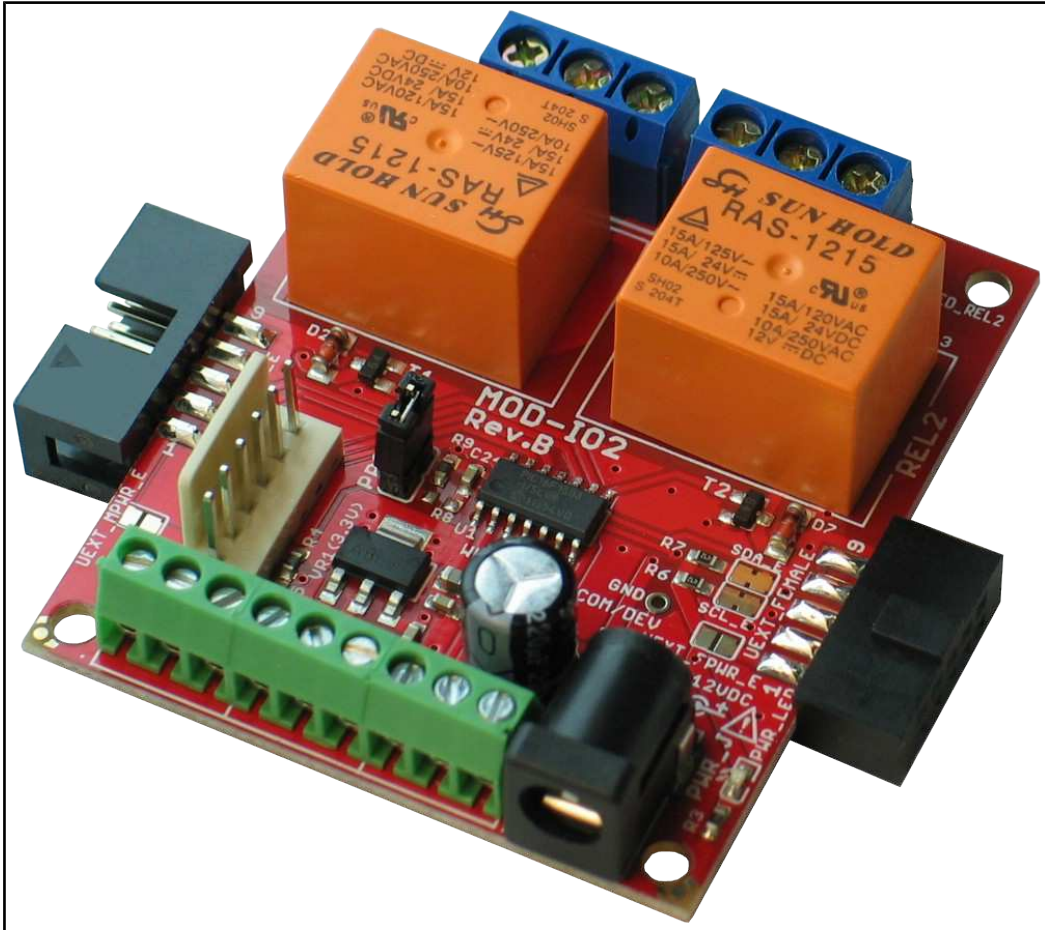
## Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China





## MOD-IO2 extension board USER'S MANUAL

Revision D, May 2015  
Designed by OLIMEX Ltd, 2012



All boards produced by Olimex LTD are ROHS compliant

## DISCLAIMER

---

© 2015 Olimex Ltd. Olimex®, logo and combinations thereof, are registered trademarks of Olimex Ltd. Other product names may be trademarks of others and the rights belong to their respective owners.

**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.**

This work is licensed under the Creative Commons Attribution-ShareAlike 3.0 Unported License. To view a copy of this license, visit <http://www.creativecommons.org/licenses/by-sa/3.0/>.



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

The software is released under GPL.

It is possible that the pictures in this manual differ from the latest revision of the board.

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.

This evaluation board/kit is intended for use for engineering development, demonstration, or evaluation purposes only and is not considered by OLIMEX to be a finished end-product fit for general consumer use. Persons handling the product must have electronics training and observe good engineering practice standards. As such, the goods being provided are not intended to be complete in terms of required design-, marketing-, and/or manufacturing-related protective considerations, including product safety and environmental measures typically found in end products that incorporate such semiconductor components or circuit boards.

Olimex currently deals with a variety of customers for products, and therefore our arrangement with the user is not exclusive. Olimex assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein.

**THERE IS NO WARRANTY FOR THE DESIGN MATERIALS AND THE COMPONENTS USED TO CREATE MOD-IO2. THEY ARE CONSIDERED SUITABLE ONLY FOR MOD-IO2.**

## Table of Contents

<b><u>DISCLAIMER.....</u></b>	<b><u>2</u></b>
<b><u>CHAPTER 1 OVERVIEW.....</u></b>	<b><u>5</u></b>
<b><u>1. Introduction to the chapter.....</u></b>	<b><u>5</u></b>
<b><u>1.1 Features.....</u></b>	<b><u>5</u></b>
<b><u>1.2 MOD-IO vs MOD-IO2.....</u></b>	<b><u>5</u></b>
<b><u>1.3 Target market and purpose of the board.....</u></b>	<b><u>6</u></b>
<b><u>1.4 Organization.....</u></b>	<b><u>6</u></b>
<b><u>CHAPTER 2 SETTING UP THE MOD-IO2 BOARD.....</u></b>	<b><u>7</u></b>
<b><u>2. Introduction to the chapter.....</u></b>	<b><u>7</u></b>
<b><u>2.1 Electrostatic warning.....</u></b>	<b><u>7</u></b>
<b><u>2.2 Requirements.....</u></b>	<b><u>7</u></b>
<b><u>2.3 Powering the board.....</u></b>	<b><u>7</u></b>
<b><u>2.4 Firmware description and basic usage under Linux.....</u></b>	<b><u>8</u></b>
<b><u>2.4.1 Custom software tool for controlling MOD-IO2 under Linux.....</u></b>	<b><u>8</u></b>
<b><u>2.4.2 I2C Tools for controlling MOD-IO2 under Linux.....</u></b>	<b><u>11</u></b>
<b><u>CHAPTER 3 MOD-IO2 BOARD DESCRIPTION.....</u></b>	<b><u>14</u></b>
<b><u>3. Introduction to the chapter.....</u></b>	<b><u>14</u></b>
<b><u>3.1 Layout (top view).....</u></b>	<b><u>14</u></b>
<b><u>CHAPTER 4 THE PIC16F1503 MICROCONTROLLER.....</u></b>	<b><u>15</u></b>
<b><u>4. Introduction to the chapter.....</u></b>	<b><u>15</u></b>
<b><u>4.1 The PIC16F1503 features.....</u></b>	<b><u>15</u></b>
<b><u>CHAPTER 5 CONNECTORS AND PINOUT.....</u></b>	<b><u>16</u></b>
<b><u>5. Introduction to the chapter.....</u></b>	<b><u>16</u></b>
<b><u>5.1 ICSP.....</u></b>	<b><u>16</u></b>
<b><u>5.2 UEXT modules.....</u></b>	<b><u>16</u></b>
<b><u>5.2.1 Female connector.....</u></b>	<b><u>16</u></b>
<b><u>5.2.2 Male connector.....</u></b>	<b><u>17</u></b>
<b><u>5.3 Relay output connectors.....</u></b>	<b><u>17</u></b>
<b><u>5.4 GPIO connectors.....</u></b>	<b><u>18</u></b>
<b><u>5.5 PWR Jack.....</u></b>	<b><u>18</u></b>
<b><u>5.6 Jumper description.....</u></b>	<b><u>19</u></b>
<b><u>5.6.1 PROG.....</u></b>	<b><u>19</u></b>
<b><u>5.6.2 SDA_E/SCL_E.....</u></b>	<b><u>19</u></b>
<b><u>5.6.3 UEXT_FPWR_E.....</u></b>	<b><u>19</u></b>
<b><u>5.6.4 UEXT_MPWR_E.....</u></b>	<b><u>19</u></b>
<b><u>5.7 Additional hardware components.....</u></b>	<b><u>19</u></b>
<b><u>CHAPTER 6 BLOCK DIAGRAM AND MEMORY.....</u></b>	<b><u>20</u></b>
<b><u>6. Introduction to the chapter.....</u></b>	<b><u>20</u></b>
<b><u>6.1 Processor block diagram.....</u></b>	<b><u>20</u></b>

---

<b>6.2 Physical memory map.....</b>	<b>21</b>
<b>CHAPTER 7 SCHEMATICS.....</b>	<b>22</b>
<b>7. Introduction to the chapter.....</b>	<b>22</b>
<b>7.1 Eagle schematic.....</b>	<b>22</b>
<b>7.2 Physical dimensions.....</b>	<b>24</b>
<b>CHAPTER 8 REVISION HISTORY AND SUPPORT.....</b>	<b>25</b>
<b>8. Introduction to the chapter.....</b>	<b>25</b>
<b>8.1 Document revision.....</b>	<b>25</b>
<b>8.2 Board's revision.....</b>	<b>26</b>
<b>8.3 Useful web links and purchase codes.....</b>	<b>27</b>
<b>8.4 Product support.....</b>	<b>28</b>

## CHAPTER 1 OVERVIEW

---

### 1. Introduction to the chapter

---

Thank you for choosing the MOD-IO2 single board computer from Olimex! This document provides a user's guide for the Olimex MOD-IO2 board. As an overview, this chapter gives the scope of this document and lists the board's features. The differences between the members of the MOD-IO2 and MOD-IO boards are mentioned. The document's organization is then detailed. The MOD-IO2 development board enables code development of applications running on the microcontroller PIC16F1503, manufactured by Microchip.

#### 1.1 Features

---

- ◆ PIC16F1503 microcontroller
- ◆ ICSP 6 pin connector for in-circuit programming with PIC-ICD2-POCKET and PIC-KIT3 or other compatible programmer/debugger.
- ◆ 9-pin terminal screw connector for 7 GPIOs 3.3V and GND
- ◆ PWR jack for 12VDC
- ◆ 7 GPIOs which could implement different functionality as PWM, SPI, I2C etc.
- ◆ 2 relay outputs with 15A/250VAC contacts with screw terminals
- ◆ RELAY output status LEDs
- ◆ Both male and female UEXT connectors
- ◆ Four mounting holes 3.3mm ~ (0.13)"
- ◆ UEXT female-female cable included
- ◆ FR-4, 1.5 mm (0.062"), soldermask, white silkscreen component print
- ◆ dimensions 61×52mm ~ (2.4×2.05)"

#### 1.2 MOD-IO vs MOD-IO2

---

MOD-IO2 is a smaller input output extension module compared to MOD-IO both in terms of size and in terms of functionality, however in a lot of situations MOD-IO2 might provide the better choice. Designs which need optocouplers should consider MOD-IO. In a lot of the other cases MOD-IO2 might be the better choice since it is possible to stack multiple MOD-IO2's over each other.

MOD-IO has a better power supply with the option to be supplied in the range of 8-30VDC.

## 1.3 Target market and purpose of the board

---

MOD-IO2 is an extension development board which can interface other Olimex boards via UEXT connector it adds add RELAYs and GPIOs. Multiple MOD-IO2s are stackable and addressable. The firmware allows you to interact with the board using simple commands and yet if you wish you can modify the firmware for your needs.

If you work with any of our development boards with UEXT connector and you need more GPIOs and RELAY outputs you can add these by connecting MOD-IO2 to your development board. This board allows easy interfacing to 2 relays and 7 GPIOs. MOD-IO2 is stackable and addressable – these boards can be plugged together and you can add as many inputs and outputs as you want! 2-4-6-8 etc! MOD-IO2 has PIC16F1503 microcontroller and the firmware is open source and available for modification.

The board is a very good addition to most of the Olimex boards if you need analog GPIOs and relays.

## 1.4 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 provides a guide for quickly setting up the board
- Chapter 3 contains the general board diagram and layout
- Chapter 4 describes the component that is the heart of the board: the PIC16F1503
- Chapter 5 covers the connector pinout, peripherals and jumper description
- Chapter 6 shows the memory map
- Chapter 7 provides the schematics
- Chapter 8 contains the revision history, useful links and support information

## CHAPTER 2 SETTING UP THE MOD-IO2 BOARD

---

### 2. Introduction to the chapter

---

This section helps you set up the MOD-IO2 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.

#### 2.1 Electrostatic warning

---

MOD-IO2 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.

#### 2.2 Requirements

---

In order to set up the MOD-IO2 optimally, the following items are required:

- Any OLIMEX board that has an UEXT connector
- 12V source of power

If you wish to reprogram the board or modify the firmware you will also need:

- PIC ICSP compatible programmer – not that the connector for the ICSP programming is the standard 0.1" one. We have a cheap compatible PIC16F1503 programmer based on Microchip's PIC-KIT3.

Some of the suggested items can be purchased by Olimex, for instance:

**PIC-KIT3** – Olimex programmer capable of programming PIC16F1503  
**SY0612E** – power supply adapter 12V/0.5A for European customers

#### 2.3 Powering the board

---

The board is powered by the power jack. You should provide 12V DC.

For the European customers we sell an affordable power supply adapter 12V/0.5A – **SY0612E**.

If you power the board correctly PWR\_LED would turn on.



## 2.4 Firmware description and basic usage under Linux

---

There is firmware loaded on the PIC of the board that allows easier use of MOD-IO2 via I2C protocol. The firmware of MOD-IO2 has gone through several iterations. The latest firmware revision is revision 3.02.

To use the firmware with not-Linux enabled host boards please refer to the README.txt in the archive that contains the firmware sources.

Firmware revisions 1, 2, and 3 are compatible. These firmware revisions define different MOD-IO2 board addresses and different command set. Firmware revisions 3, 3.1, and 3.02 (3.xx) are compatible.

Please note that the custom firmware does NOT support all of the hardware capabilities of MOD-IO2. In some cases you might need to adapt the firmware to use the hardware of MOD-IO2 to its full potential!

### 2.4.1 Custom software tool for controlling MOD-IO2 under Linux

---

To make the things even simpler we have written a software tool for controlling MOD-IO2 under Linux. You might find it here:

<https://github.com/OLIMEX/OLINUXINO/tree/master/SOFTWARE/UEXT%20MODULES/MOD-IO2/linux-access-tool>

This software tool requires a Linux-enabled board. The tool works with MOD-IO2 units loaded with firmware revision 3 or newer. For full compatibility with the custom software tool, your MOD-IO2 board needs to use firmware revision 3.02 or newer. To use the tool simply place file "modio2tool" on your board. Navigate to the folder where you placed it and type "./modio2tool -h" to get help on all of the available commands.

Most of the commands require the hardware I2C number as defined in your Linux distribution with parameter -B X, where X is the number of the I2C interface. Note that by default the software is set for use with hardware I2C interface #2 and board ID 0x21 – if your setup has different you would need to specify every time by using -B X (X is the hardware I2C number) and -A 0xXX (XX is the I2C address of the module).

#### Some examples for the usage of modio2tool and MOD-IO2 in Linux:

##### - Bringing up the help menu:

```
./modio2tool -h
```

,where

```
./modio2tool – executes the binary  
-h – parameter used to request the help information
```

Expected result: the format of commands would be shown and a list of commands would be printed.

---

**- Switching on both relays:**

```
./modio2tool -B 0 -s 3
```

,where

- B 0 – sets the board to use its hardware I2C #0 (typically either “0”, “1”, or “2”)
- s 3 – “s” is used to turn on the relays; “3” specifies to turn on both relays (use “1” or “2” for only the first or only the second relay)

Expected result: a specific sound would occur and relay LEDs would turn on.

**- Switching off both relays:**

```
./modio2tool -B 0 -c 3
```

,where

- B 0 – sets the board to use its hardware I2C #0 (typically either “0”, “1”, or “2”)
- c 3 – “c” is used to switch off the state relays; “3” specifies to turn off both relays (use “1” or “2” for only the first or only the second relay)

Expected result: a specific sound would occur and the relay LEDs would turn off.

**- Reading the status of the relays (available since MOD-IO2's firmware revision 3.02):**

```
./modio2tool -B 0 -r
```

,where

- B 0 – sets the board to use its hardware I2C #0 (typically either “0”, “1”, or “2”)
- r – “r” is used to read the relays;

Expected result: the state of the relays would be printed. 0x03 means that both relays are on (equivalent of the binary 0x011).

**- Reading analog inputs:**

```
./modio2tool -B 0 -A 1
```

,where

- B 0 – sets the board to use its hardware I2C #0 (typically either “0”, “1”, or “2”)
- A 1 – “A” is used to read the analog input; “1” is the analog input that is read – you can use “1”, “2”, “3” or “5” since not all AN signals are available.

Expected result: the Voltage of the AN would be printed. If there is nothing connected it can be anything like “ADC1: 2.311V”.

- Changing the I2C address – if you use more than one MOD-IO2 (available since MOD-IO2's firmware revision 3.02)

```
./modio2tool -B 0 -x 15
```

,where

-B 0 – sets the board to use its hardware I2C #0 (typically either “0”, “1”, or “2”)  
-x 15 – “x” is used to change the I2C address of the board; “15” is the desired number – it is different from the default “0x21”.

Expected result: the board would have new I2C address and you would need to specify it with -A 0xXX if you wish to use the modio2tools in future.

For more information refer to the help returned by modio2tools or to modio2tools's source code.

---

## 2.4.2 I2CTools for controlling MOD-IO2 under Linux

---

Instead of the custom program mentioned in 2.4.1 you might use the popular I2CTools (here: <http://www.lm-sensors.org/wiki/I2CTools>). MOD-IO2 is compatible with I2CTools since the release of its firmware 3.

In that case the commands are the most popular ones from the i2c-tools – [i2cdetect](#), [i2cdump](#), [i2cget](#), [i2cset](#). More information might be found here: <http://www.lm-sensors.org/wiki/i2cToolsDocumentation>.

Use the above commands and the information about the firmware to send (i2cset) and receive (i2cget) different data. The information about the firmware is located in a README.txt file in the archive of the firmware; the archive containing the latest firmware (3.02) might be found here: [https://www.olimex.com/Products/Modules/IO/MOD-IO2/resources/MOD-IO2\\_firmware\\_v\\_3\\_02.zip](https://www.olimex.com/Products/Modules/IO/MOD-IO2/resources/MOD-IO2_firmware_v_3_02.zip)

### Some examples for setting/reading MOD-IO2's peripherals in Linux using i2c-tools

#### - Turning on the relays:

```
i2cset -y 2 0x21 0x40 0x03
```

,where

i2cset – command for sending data;  
-y – to skip the y/n confirmation prompt;  
2 – board's hardware I2C number (typically 0 or 1 or 2);  
0x21 – board address (0x21 should be used for writing);  
0x40 – relay operations (as seen in the firmware README.txt);  
0x03 – should be interpreted as binary 011 – turns on both relays (0x02 would turn only second relay, 0x01 only the first, 0x00 would turn both off – 0x03 again would turn them off also);

Expected result: a specific sound would occur and relay lights would turn on.

**- Reading the status of the relays (available since MOD-IO2's firmware revision 3.02):**

```
i2cset -y 2 0x21 0x43
```

and then the read command

```
i2cget -y 2 0x21
```

,where

i2cset – command for sending data;

-y – to skip the y/n confirmation prompt;

2 – I2C number (usually 0, 1, or 2);

0x21 – board address (0x21 should be used for writing);

0x43 – read relay operations (as seen in the firmware README.txt);

Expected results: 0x00 – meaning both relays are off; 0x03 – should be interpreted as binary 011, e.g. both relays are on; etc.

**- Reading analog inputs/outputs:**

```
i2cset -y 2 0x21 0x10
```

and then the read command

```
i2cget -y 2 0x21
```

,where

0x10 – the first analog IO;

The big thing here is that to read you actually have to write (“that you would read”). Read is a combination of i2cset and i2cget!

Expected results: on the terminal you would receive random and changing number or 0x00 or 0x08 or 0xFF whether you have the GPIO floating or set to 0V or set to 3.3V.

**- Setting all analog IOs at high level:**

```
i2cset -y 2 0x21 0x01 0x01
```

,where

0x21 – the I2C address of the MOD-IO2

0x01 – according to the README.txt is SET\_TRIS is used to define port directions;

0x01 – the high level (for low level use 0x00).

**- Reading all analog IOs:**

```
i2cset -y 2 0x21 0x01  
i2cget -y 2 0x21
```

Detailed explanation of the preloaded software may be found in the demo package available at our web-page.

**- Changing I2C device address – if you use more than one MOD-IO2 (available since MOD-IO2's firmware revision 3.02)**

```
i2cset 2 0x21 0xHH
```

where

HH is new address in hexadecimal format

Note that PROG jumper must be closed to be able to change the address.

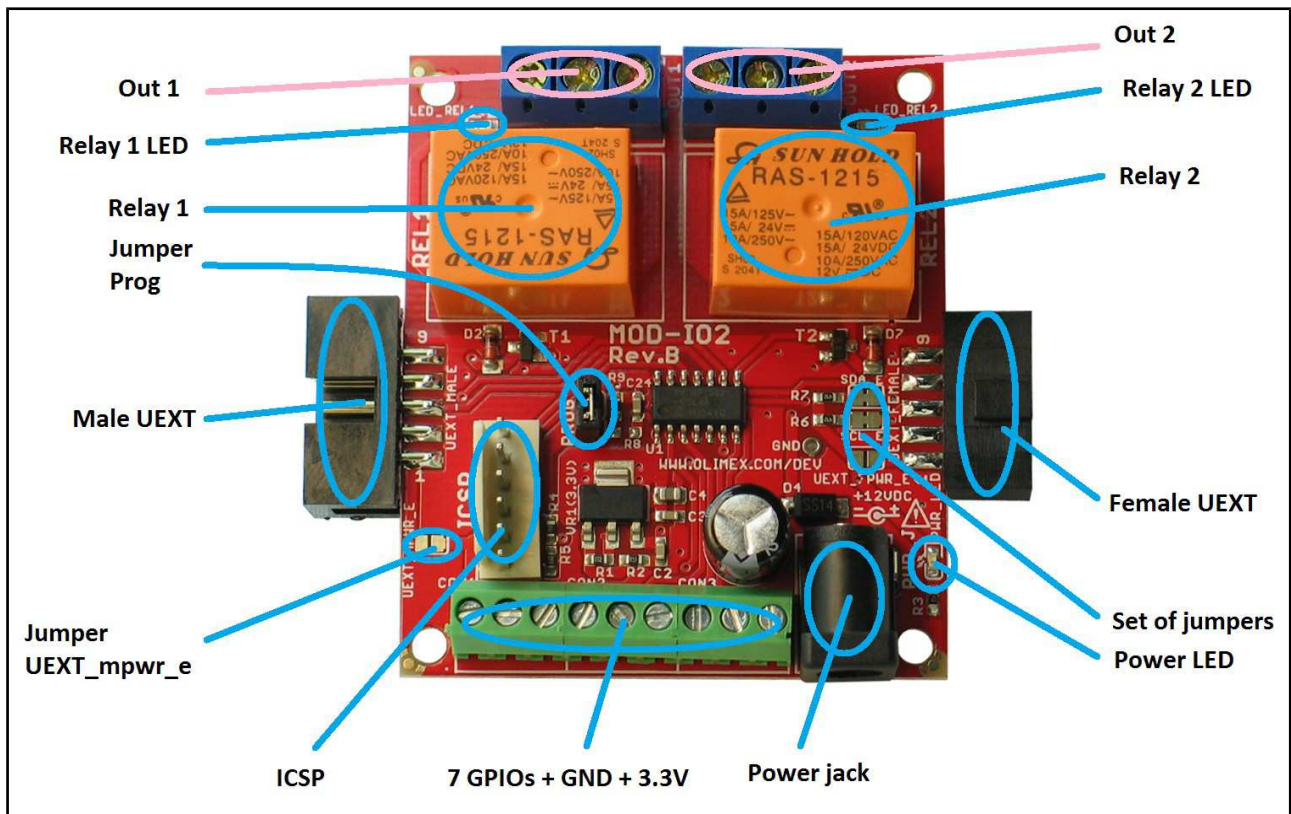
If you forget the number of the address you can use the modio2tool to find the address, the command and parameter would be "modio2tool -I". You can also reset the default address (0x21) with the command and parameter "modio2tool -X".

## CHAPTER 3 MOD-IO2 BOARD DESCRIPTION

### 3. 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 MOD-IO2 board itself.

### 3.1 Layout (top view)



---

## CHAPTER 4 THE PIC16F1503 MICROCONTROLLER

---

### 4. Introduction to the chapter

---

In this chapter is located the information about the heart of MOD-IO2 – its PIC16 microcontroller. The information below is a modified version of the datasheet provided by its manufacturers from Microchip.

#### 4.1 The PIC16F1503 features

---

- Enhanced Mid-range Core with 49 Instruction, 16 Stack Levels
- Flash Program Memory with self read/write capability
- Internal 16MHz oscillator
- 4x Standalone PWM Modules
- Complementary Waveform Generator (CWG) Module
- Numerically Controlled Oscillator (NCO) Module
- 2x Configurable Logic Cell (CLC) Modules
- Integrated Temperature Indicator Module
- 8 Channel 10-bit ADC with Voltage Reference
- 5-bit Digital to Analog Converter (DAC)
- I<sup>2</sup>C, SPI
- 25mA Source/Sink current I/O
- 2x 8-bit Timers (TMR0/TMR2)
- 1x 16-bit Timer (TMR1)
- Extended Watchdog Timer (WDT)
- Enhanced Power-On/Off-Reset
- Low-Power Brown-Out Reset (LPBOR)
- Programmable Brown-Out Reset (BOR)
- In Circuit Serial Programming (ICSP)
- In-Circuit Debug using a Debug Header
- PIC16LF1503 (1.8V – 3.6V)
- PIC16F1503 (2.3V – 5.5V)

For comprehensive information on the microcontroller visit the Microchip's web page for a datasheet.

At the moment of writing the microcontroller datasheet can be found at the following link:  
<http://ww1.microchip.com/downloads/en/DeviceDoc/41607A.pdf>.



## CHAPTER 5 CONNECTORS AND PINOUT

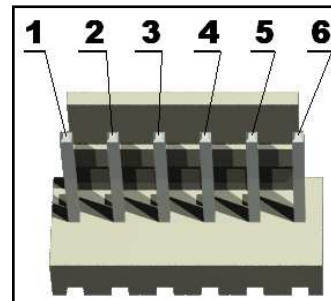
### 5. Introduction to the chapter

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

#### 5.1 ICSP

The board can be programmed and debugged from the 6-pin ICSP. Below is the table of the JTAG. This interface can be used with the Olimex's PIC-KIT3 debuggers.

ICSP			
Pin #	Signal Name	Pin #	Signal Name
1	MCLR	4	GPII00_ICSPDAT
2	+3.3V	5	GPII00_ICSPCLK
3	GND	6	Not connected



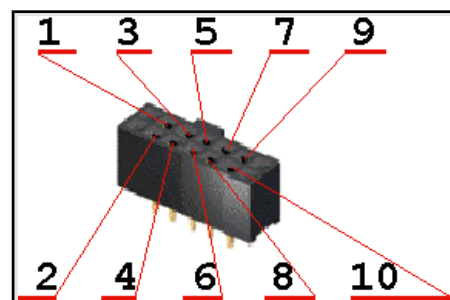
#### 5.2 UEXT modules

MOD-IO2 board has two UEXT connectors (male and female) and can interface Olimex's UEXT boards. For more information on UEXT please visit: <https://www.olimex.com/Products/Modules/UEXT/>

##### 5.2.1 Female connector

The female connector is used either to connect to a board directly (without using the female-female cable) or to connect the module to another MOD-IO2 – to create a stackable module that can be addressed via the I2C. Remember to change the I2C address of each board when using multiple boards. By default the I2C address is 0x21.

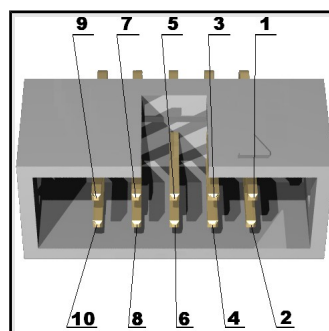
Female UEXT			
Pin #	Signal name	Pin #	Signal name
1	+3.3V	6	SDA
2	GND	7	Not connected
3	Not connected	8	Not connected
4	Not connected	9	Not connected
5	SCL	10	Not connected



### 5.2.2 Male connector

The male connector is used with the ribbon cable in the package to connect to another male UEXT or to connected to another MOD-IO2

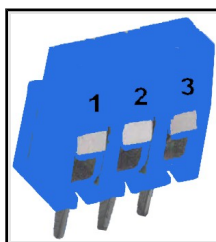
Male UEXT			
Pin #	Signal name	Pin #	Signal name
1	+3.3V	6	SDA
2	GND	7	Not connected
3	Not connected	8	Not connected
4	Not connected	9	Not connected
5	SCL	10	Not connected



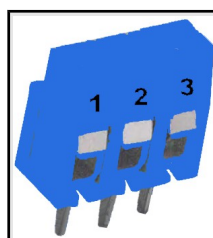
### 5.3 Relay output connectors

There are two relays in MOD-IO. Their output signals are the standard Normal Closed (NC), Normal Open (NO) and the Common (COM).

REL1 - OUT1	
Pin #	Signal name
1	NO - normal open
2	NC - normal closed
3	COM - common



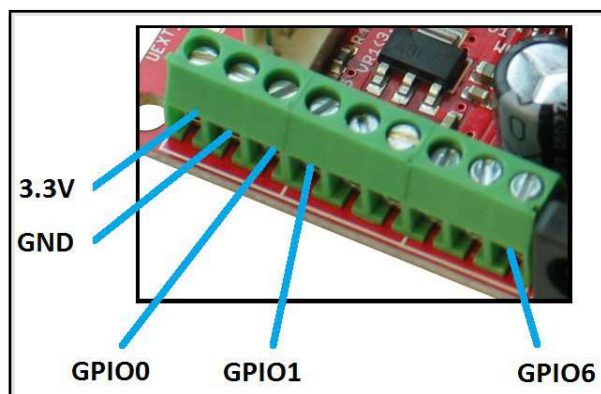
REL2 - OUT2	
Pin #	Signal name
1	COM - common
2	NO - normal open
3	NC - normal closed



## 5.4 GPIO connectors

The GPIO connectors can be used to implement PWM, I2C, SPI, etc. Note that the names of each pin are also printed at the bottom of the board.

Pin #	Signal name	Analog Input
1	3.3V	-
2	GND	-
3	GPIO0	AN0
4	GPIO1	AN1
5	GPIO2	AN2
6	GPIO3	AN3
7	GPIO4	-
8	GPIO5	AN7
9	GPIO6	PWM

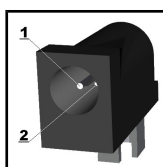


## 5.5 PWR Jack

The DC barrel jack has 2.0mm inner pin and 6.3mm hole. More information about the exact component might be found here: <https://www.olimex.com/wiki/PWRJACK>

For the European customers, we also stock and sell basic power supply adapters compatible with the power jack.

Pin #	Signal name
1	Power Input
2	GND



## 5.6 Jumper description

---

Please note that almost all (except PROG) of the jumpers on the board are SMD type. If you feel insecure in your soldering/cutting technique it is better not to try adjusting SMD jumpers.

Also if you feel incapable of removing the PTH jumper with hands better use tweezers.

### 5.6.1 PROG

---

When closed this jumper allows addressing a single MOD-IO2 (the one with the closed jumper) over all others stacked to it – since initially all connected MOD-IO2 boards have the same address. After you have closed PROG on one of the MOD-IO2s you can communicate with that one via the UEXT using the commands found in the commands list that can be found on the product's web page.

Default position is open.

### 5.6.2 SDA\_E/SCL\_E

---

When you have more than one MOD-IO2 connected you need to keep those two jumpers closed, else the I2C line will be disconnected.

The default positions for both jumpers are closed/soldered.

### 5.6.3 UEXT\_FPWR\_E

---

If closed provides 3.3V at the female UEXT connector. (be careful since if you close that jumper and also you close the male one on the next MOD-IO2 line this might cause electrical burn to the board.

Default position is open/unsoldered.

### 5.6.4 UEXT\_MPWR\_E

---

If closed provides 3.3V at the male UEXT connector. (be careful since if you close that jumper and also you close the female one on the next MOD-IO2 line this might cause electrical burn to the board.

The default position is open/unsoldered.

## 5.7 Additional hardware components

---

The components below are mounted on MOD-IO2 but are not discussed above. They are listed here for completeness:

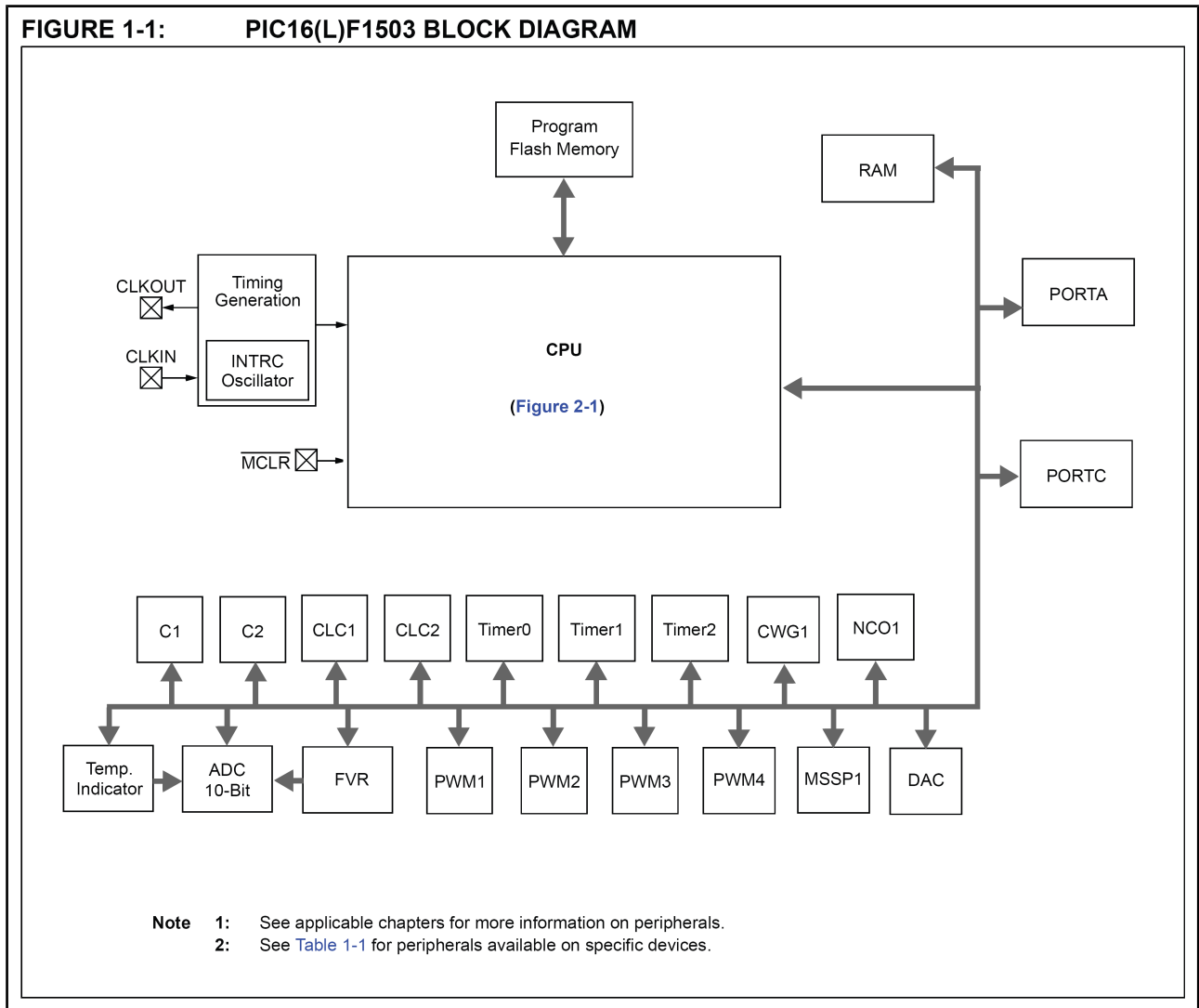
Relay LEDs + Power LED

# CHAPTER 6 BLOCK DIAGRAM AND MEMORY

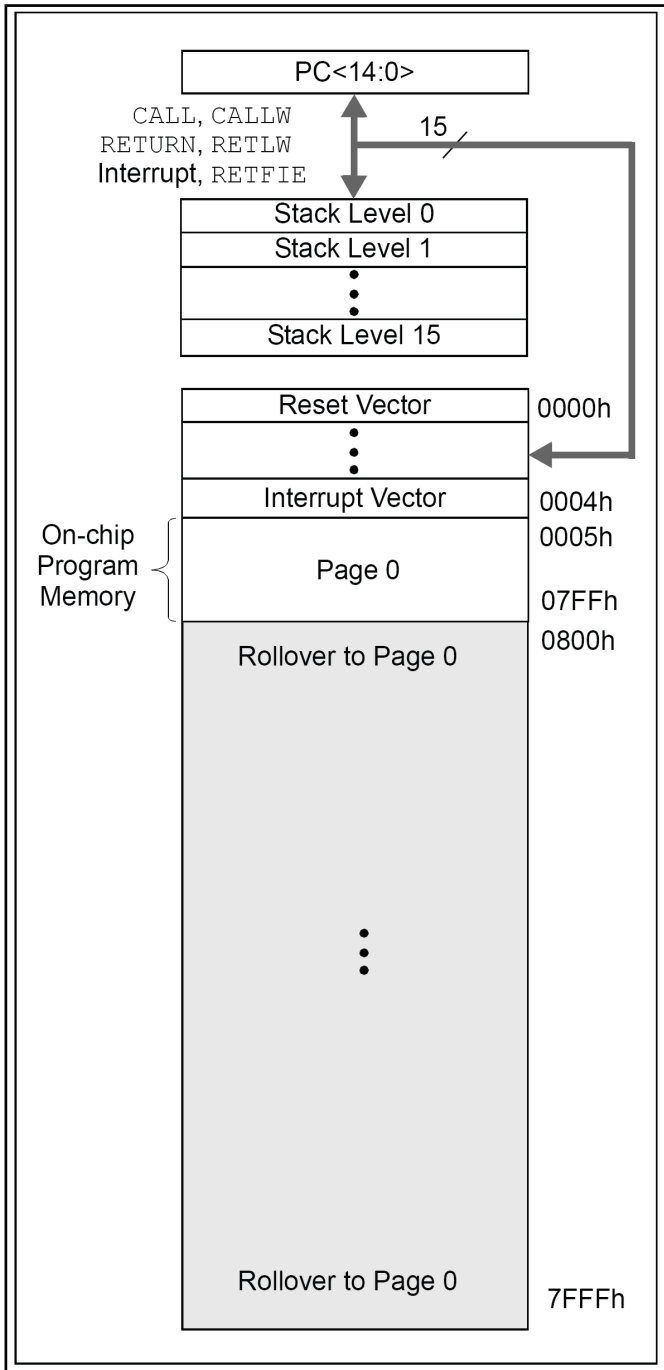
## 6. Introduction to the chapter

Down this page you can find a memory map for this family of processors. It is strongly recommended to refer to the original datasheet released by Microchip for one of higher quality.

### 6.1 Processor block diagram



## 6.2 Physical memory map



## CHAPTER 7 SCHEMATICS

---

### 7. Introduction to the chapter

---

In this chapter are located the schematics describing logically and physically MOD-IO2.

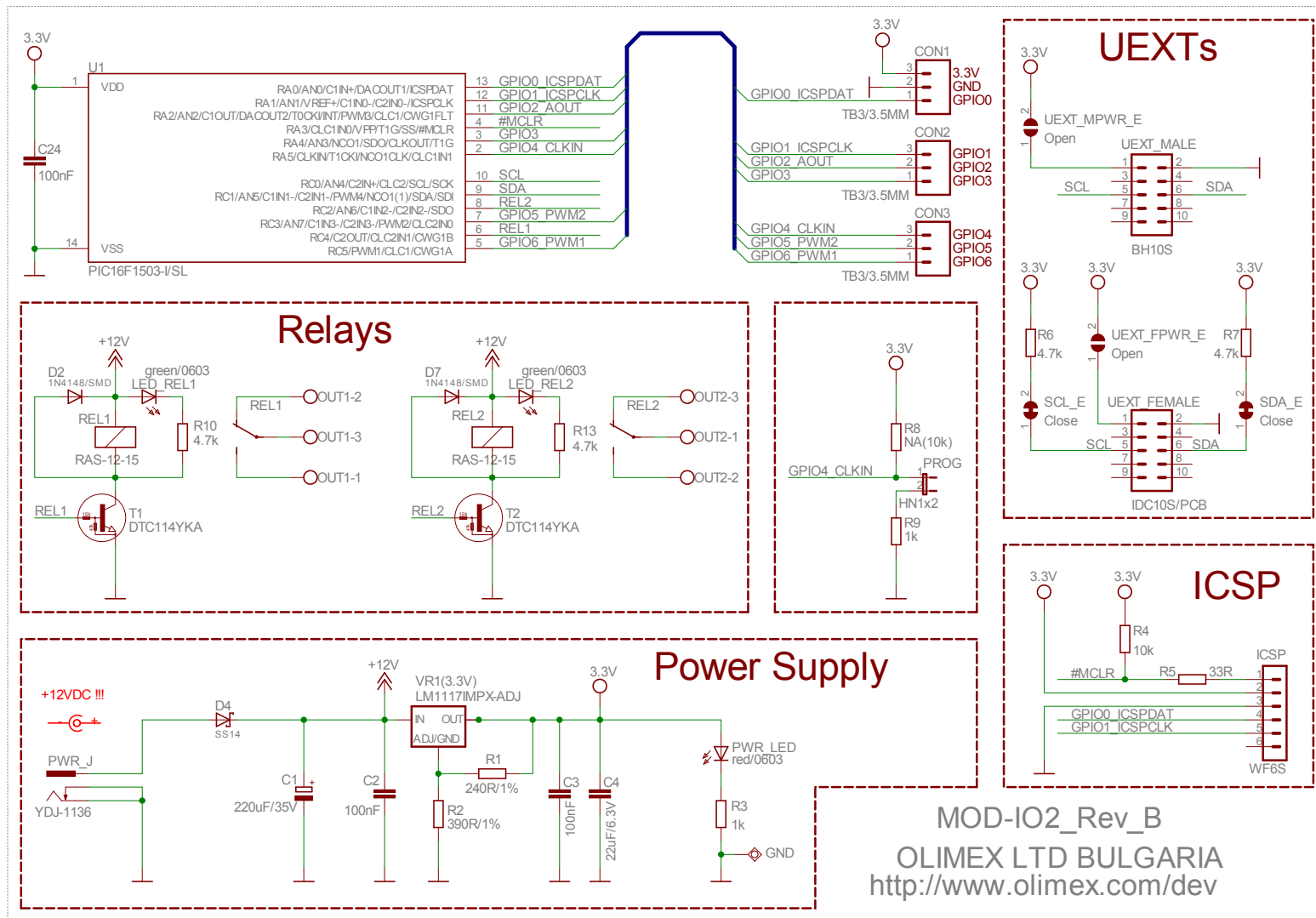
#### 7.1 Eagle schematic

---

MOD-IO2 schematic is visible for reference here. You can also find it on the web page for MOD-IO2 at our site: <https://www.olimex.com/Products/Modules/IO/MOD-IO2/open-source-hardware>

They are located in HARDWARE section.

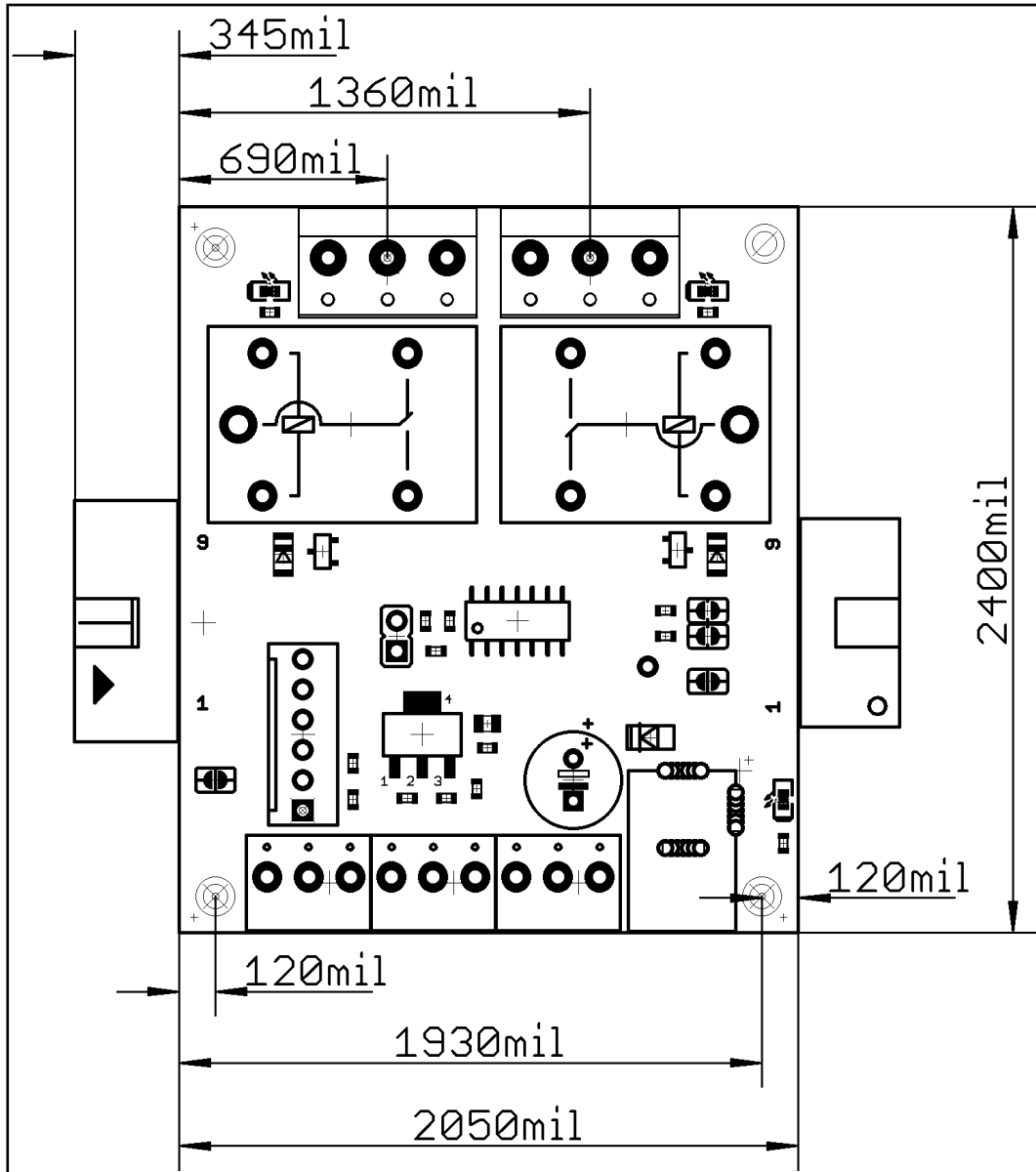
The EAGLE schematic is situated on the next page for quicker reference.





## 7.2 Physical dimensions

Note that all dimensions are in mils.



The three highest elements on the board in order from the tallest to the shortest are: relay T1 – 0.600" (15.25 mm) over the pcb; relay T2 – 0.600" (15.25 mm); ICSP connector – 0.450" (11.43 mm).

Note that the above measures does not include the PCB.

## CHAPTER 8 REVISION HISTORY AND SUPPORT

### 8. 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.

### 8.1 Document revision

Revision	Changes	Modified page#
A, 27.08.12	- Initial creation	All
B, 16.10.12	- Fixed several leftovers from the template which were referencing wrong processors and boards - Updated links	6, 10, 20
C, 24.10.13	- Updated Disclaimer to fit the open-source nature of the board - Added few examples and firmware version 3 explanation - Updated Product support - General formatting improvements	2 7 23 All
D, 27.05.15	- Updated the manual to reflect latest firmware revision 3.02 - Added information about the new Linux tool - modio2tools	7, 8, 9, 10, 11