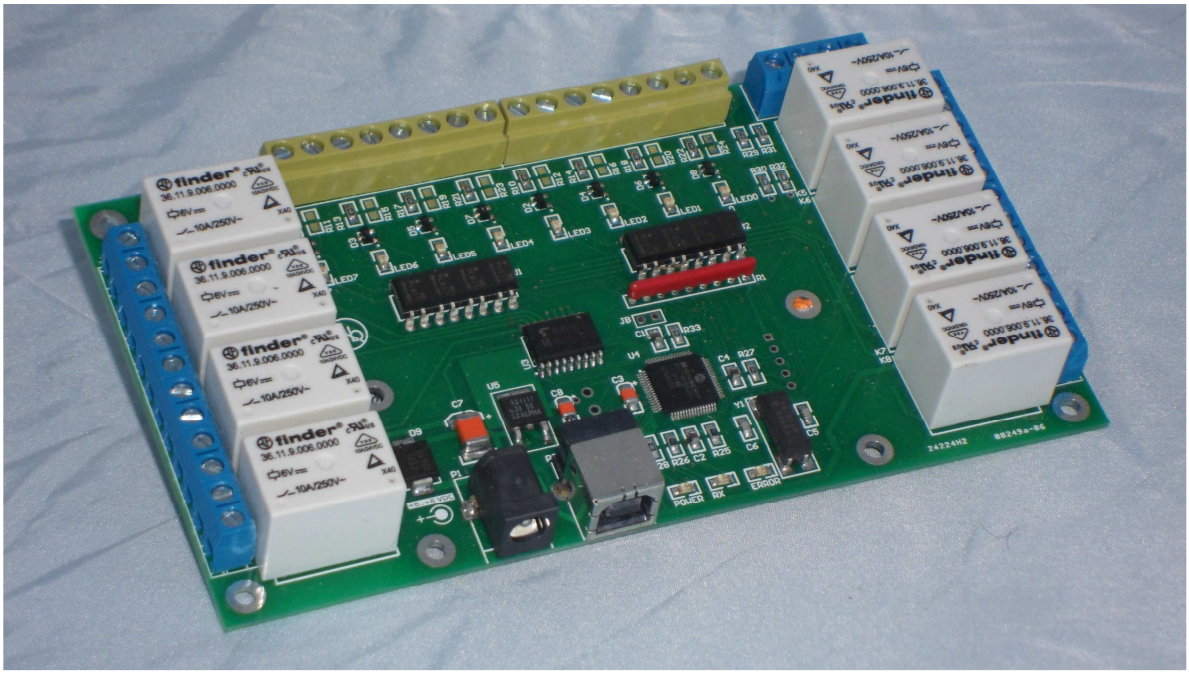
USB-IO

USB board to control up to
- **8 digital optically-isolated inputs**
- **8 digital relay outputs**
- **2 analog inputs**

Programming and User Manual



ATTENTION: USB-IO board are delicate electronic devices, handle with care avoid electrostatic discharge. Hold the board by the edges to avoid touching the integrated circuits on it.

Version 1.1 - December 2013

USB-IO_Concept

USB-IO interface adapter allows connecting various hardware devices to your PC. Easy installation and plug-and-play support makes USB-IO board an ideal solution to add USB interface to your HW.

The USB-IO has 8 SPDT relays outputs, 8 optically isolated inputs and 2 analog inputs. This device has many applications and we couldn't possibly list them all, but here are a few examples : The relay can be used to turn on and off a sprinkler system, automating a hot tub, or low-voltage lights. The inputs can be used to monitor magnetic window switches or garage door position. The analog inputs can be used to monitor some analog levels.

Helpful the provided demonstration software, connect the USB-IO to a USB host port and control IO's is really simple. Or with the Helpful provided Class file for Visual Studio, develop your own application.

USB-IO_Caracteristics

- Full HID USB2.0 compliant; driver natively include under Windows or Linux OS
- High-Speed USB 2.0 device, USB 1.1 compatible
- 8 Relay output's 10A / 250VAC
- 8 optically isolated input's 0/10VDC – User modifiable*
- 2 x 10 bits analog input's 0/10VDC – User modifiable*
- 1 Kbytes of user Flash memory
- Demonstration software and DLL are available FREE OF CHARGES (Windows 2000 / XP / VISTA / Seven) – Visual Studio - Framework .NET 4.0 © MICROSOFT
- PCB dimensions : 150x83 mm - RoHS compliant
- Weight : 181 grams

* need solders tools to change SMD components

Support

Commercial department :
commercial@eko-fpga.com

WEB Site :
http://www.eko-fpga.com

Technical department :
technique@eko-fpga.com

The board is adaptable according to your specifications.
If the volume of your application requires it, do not hesitate to contact us.

If you experience any problems with this manual or just want to give us some feedback, please email us at: ***technique@eko-fpga.com***. Please detail any errors you find and include your mailing address so that we can send you any manual updates.

TABLE OF CONTENTS

GLOSSARY

USB-IO : USB IO controller board
$xx : HEXADECIMAL notation
%xxxxxxxx : BINARY notation

DOCUMENT REVISION HISTORY

Paragraphs	Rev.	Date	Modifications
	1.0	20.06.11	First edition of the document.

NOTA: all the pages carry the index corresponding to their edition even if their contents were not modified.

1 Preface

USB-IO differs from most of other low-cost adapter by the event-driven interface. It means that the USB-IO adapter can notify your PC software immediately after the input value has changed. The event-driven interface is very flexible. The conditions of an event generation can be configured for every USB-IO module independently. For example, ADC module can send events when the voltage on the analog input descends below predefined value or rises above any other else.

Would you like to analyze changes over time? No problem, USB-IO adapter can send events periodically with the predefined repeat interval. The polling interface is supported by the USB-IO adapter.

This is a fully assembled and tested unit. The board requires external power of 6-8VDC / 500mA.

The relays can control loads of up to 10 amps and 240VAC. Please see below for more detailed specifications.

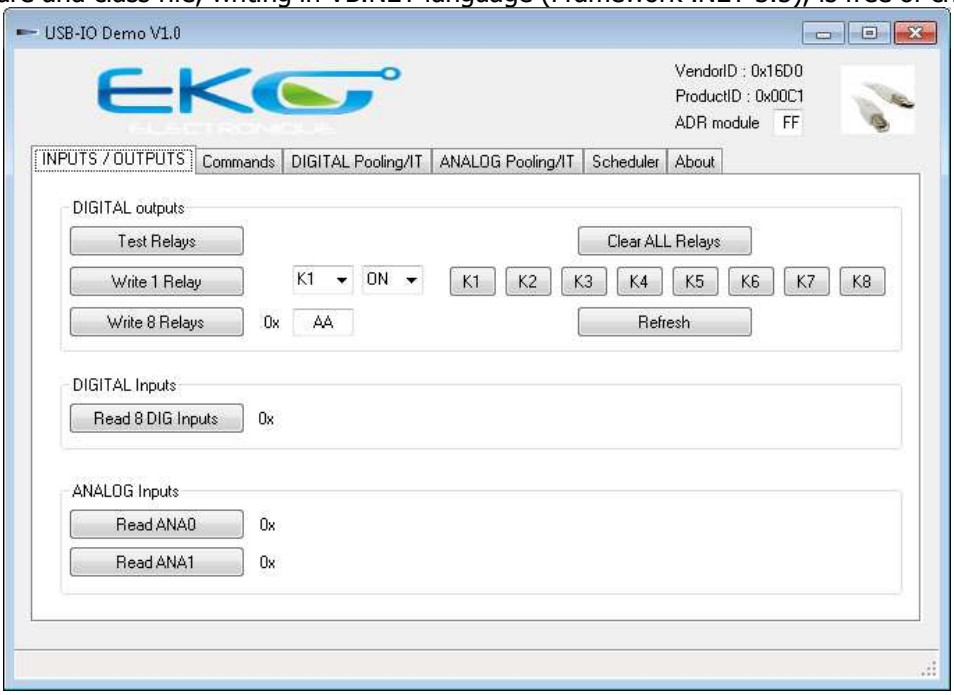
2 USB HID device

USB-IO uses the Human Interface Device (HID) USB profile. It does not require USB drivers and so is immediately Plug-and-Play with present and future Windows©, Linux© and Mac© operating systems.

Many links are available on the Web. You can start by:

- http://www.lvr.com/hidpage.htm
- http://labs.mike-obrien.net/Document.aspx?id=hidlibrary

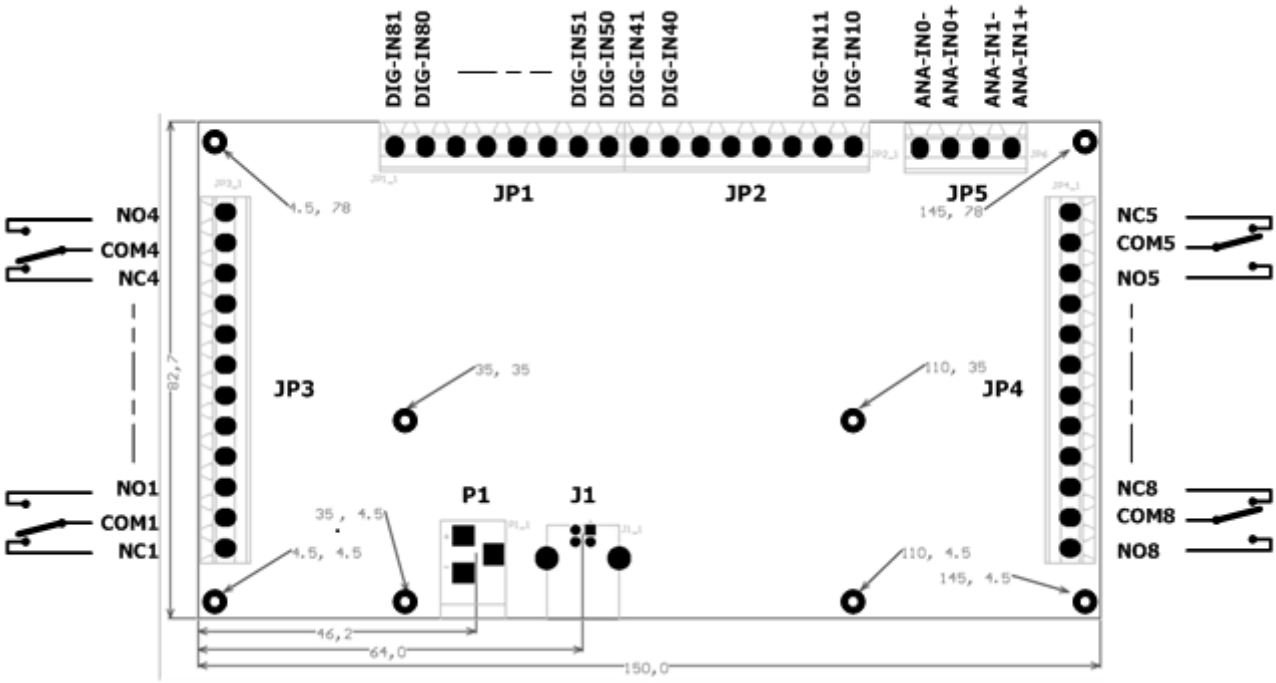
Control software and class file, writing in VB.NET language (Framework .NET 3.5), is free of charge:



You can download the demonstration software on http://www.eko-fgpa.com

3 Hardware Information's

3.1 Board dimensions & Connectors

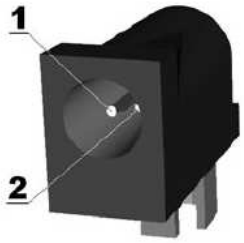


The board provides 8 fixation holes of 3.2mm diameters.

3.1.1 POWER Supply connector P1

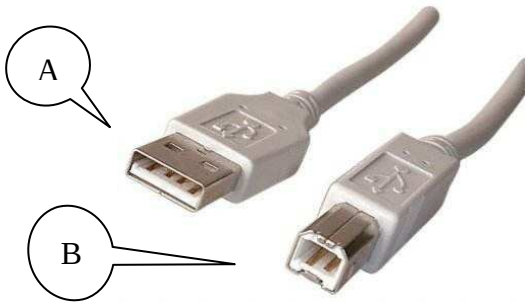
P1 is the power connector. A +6 to +8VDC /500mA power supply need to be provided.

PIN	Signal Name
1 (center)	Positive
2	Ground



3.1.2 USB connector J1

Connect the module directly to a free USB port from your computer: the connection between the board and the host port will be done using a cable with a USB type A and a USB type B connectors.



3.1.3 DIGITAL OUTPUTS connectors JP3 & JP4

The 8 digitals outputs are relay type 1C-form. Form C relays are SPDT (single-pole double-throw) and break the connection with one throw before making contact with the other (break-before-make).

You can access individually at each relay or simultaneous to the 8 relays.

3.1.4 DIGITAL INPUTS connectors JP1 & JP2

With USB-IO event-driven interface your application can be notified when the digital input value changes.

You can configure the USB-IO adapter to send events only after changes or to send periodically the 8 inputs values. The period is configurable from 1ms to 16777216ms (24bits period counter).

3.1.5 ANALOG INPUTS connectors JP5

The 2 analog inputs are connected to 10-bit analog-to-digital converter (ADC). The ADC module is most commonly used to detect variations in voltage, light, temperature or pressure. It can also be used to read switches, potentiometers and other analog devices.

At any time you can read the voltage on the analog inputs.

Besides, the USB-IO adapter may send events containing the analog input values. Events are sending either periodically with the predefined repeat interval (from 1ms to 16777216ms) or when the voltage reaches the user defined limits. These limits are specified as low and high threshold values. They are compared each 1ms with the input voltage. The ADC module generates one event when the input voltage descends below the low threshold value or rises above the high threshold value.

3.2 LED information's

The LEDs on the board are used to indicate state of digital inputs, power, data and error transmissions.

LED1 to LED8 : digital inputs	Illuminating when input activate
LED9 : POWER	Illuminating when power on
LED10 : RX	Flashing when a frame is received with the right address of the board.
LED11 : ERROR	Illuminating when no USB communication

3.3 Electricals characteristics

3.3.1 Digital Inputs

These inputs may be driven by DC sources of 0 to 12 volts. Two Resistors forms a voltage divider. The input voltage range may be adjusted to your needs.

Original Factory settings	*Min*	*Max*
VIH	3.5V	12V
VIL	0	3V

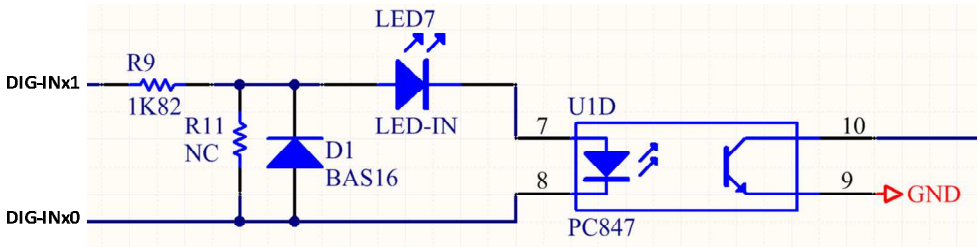


Figure 1 : Example of one input digital circuit

- Number of inputs: 8 optically isolated from each other and from the computer
- Voltage Range: 0 to 35 DC (user adjustable)
- Input Resistance: 1.8K ohms in series with opto coupler (factory setting)
- Response Times : Rise Time = 4 uS / Fall Time = 3 uS (typical)
- Short time inversion polarity is not destructive.

Consult with factory for available modified input ranges or have a look to
http://en.wikipedia.org/wiki/Voltage_divider

Examples for digital input 1	**VIL min/ max**	**VIH min/ max**	
R9 = 1K8 R11 = Not Connected	0 / 3	3.5 / 12	**Default configuration**
R9 = 22K1 R11 = Not Connected	0 / 12	13 / 20	
R9 = 22K1 R11 = 15K	0 / 15.5	16 / 35	

The resistors values are 0805 SMD size and 1% tolerance.

The inputs of each channel are comprised of 2 pins: DIG-INx0 and DIG_INx1. The DIG-INx0 is the references voltage.

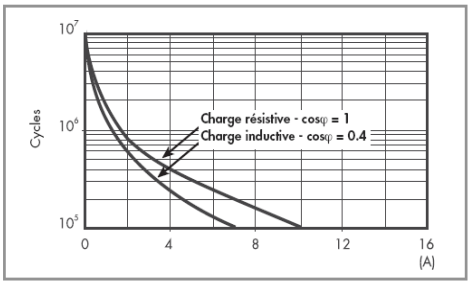
Digital inputs	Resistors Voltage divisor	LED Channel
1	R22 / R24	LED0
2	R18 / R20	LED1
3	R14 / R16	LED2
4	R10 / R12	LED3
5	R21 / R23	LED4
6	R17 / R19	LED5
7	R13 / R15	LED6
8	R9 / R11	LED7

3.3.2 Digital Outputs

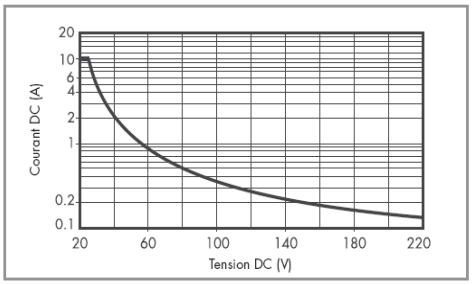
The board's outputs are comprised of 8 FORM C SPDT electro mechanical relays reference 36.11.9.006.0000 from Finder manufacturer (http://www.findernet.com).

- Number of outputs: 8 SPDT form C
- Synchronization between each relay: Activation time :9ms Release time :3ms (relay setting time)
- Contact Type: AgSnO2
- Rated Load AC: 10 A at 250 VAC
- Rated Load DC: 10 A at 30 VDC
- Contact Resistance: 100 mΩ max.
- Contact Life mechanical: 5 million operations min.

F 36 - Durée de vie électrique (AC) en fonction de la charge



H 36 - Pouvoir de coupure maxi pour une charge en DC1



- La durée de vie électrique pour des charges résistives en DC1 ayant des valeurs de tension et de courant sous la courbe est ≥ 100x10³ cycles.
- Pour les charges en DC13, le raccordement d'une diode polarité inverse en parallèle avec la charge permet d'obtenir une durée de vie électrique identique à celle obtenue avec une charge en DC1. Nota: le temps de coupure de la charge sera augmenté.

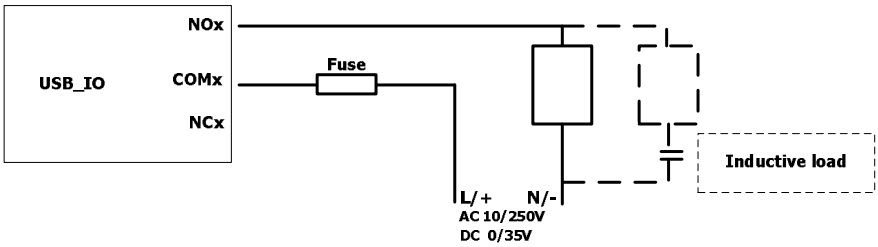


Figure 2 : How to connect inductive loads

If you use an inductive load on the digital outputs relays, you may need to add a snubber circuit: http://en.wikipedia.org/wiki/Snubber

3.3.3 Analog inputs

The board comprised 2 analog inputs ANA0 and ANA1. The Analog to digital convertor performed on 10 bits. With the factory configuration, the full scale input is 0/10Volts. Each bit is equal to 10/1024 = 9.76mV.

VANx	INPUT_ANAx register value
+10V DC	0x3FF = 0b1111111111
0V DC	0x000 = 0b0000000000

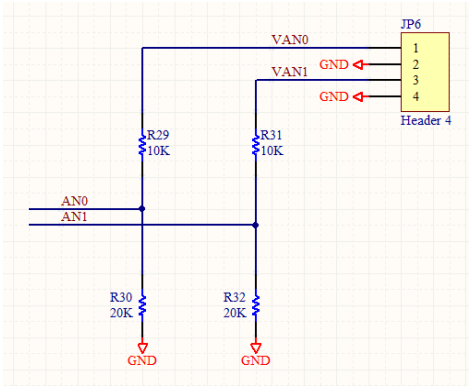


Figure 3 : input analog circuits

Consult with us for available modified input ranges or have a look to http://en.wikipedia.org/wiki/Voltage_divider

The values AN0 and AN1 (CPU pins) do never increase to 3.6V (3.3V max. recommended).

	min	*max*	
ANx	0V	3.6V	**Absolute maximum voltage**

DO NOT REVERSE POLARITY

You may add a low capacity between the VANx pin and GND to filter the input if you need.
If some analog inputs are not used, it is recommended to short wire inputs pins to GND.

Examples for digital analog 0	VAN0 max / AN0 max	
R29 = 100K R30 = 49.9K	10v / 3.32V	**Default configuration**

3.4 Environmental

- Operating Temperature Range: 0° to 70°C
- Storage Temperature Range: -40° to +85°C
- Humidity: Maximum 90% RH, without condensation
- Board Dimension: 150 x83mm
- Weight : about 180 grams

4 Registers

4.1 Mapping Register

@	Name	Access	Factory settings	
0x00	ADDRESS	R/W	0xFF	Address of the USB-IO controller
	reserved			
0x10	CONFIG_DIG	R/W	0x00	Configuration bits for the DIGITALS IO's
	reserved			
0x20	OUTPUTS_REL	R/W	0x00	States of the 8 relay outputs
0x22	POOL_DIG_TIMERH	R/W	0x00	Set the pooling time of digitals inputs MSB byte
0x23	POOL_DIG_TIMERM	R/W	0x00	Set the pooling time of digitals inputs Middle byte
0x24	POOL_DIG_TIMERL	R/W	0x64	Set the pooling time of digitals inputs LSB byte (100 milliseconds)
	reserved			
0x30	CONFIG_ANA	R/W	0x00	Configuration bits for the ANALOGS inputs
0x31	THH_ANA0	R/W	0x02	Threshold High for the analog0 input – MSB
0x32	THL_ANA0	R/W	0x00	Threshold Low for the analog0 input – LSB
0x33	TLH_ANA0	R/W	0x00	Threshold Low for the analog0 input – MSB
0x34	TLL_ANA0	R/W	0x80	Threshold Low for the analog0 input – LSB
0x35	THH_ANA1	R/W	0x01	Threshold High for the analog1 input – MSB
0x36	THL_ANA1	R/W	0x00	Threshold High for the analog1 input – LSB
0x37	TLH_ANA1	R/W	0x00	Threshold Low for the analog1 input – MSB
0x38	TLL_ANA1	R/W	0x30	Threshold Low for the analog1 input – LSB
0x3C	POOL_ANA0_TIMERH	R/W	0x00	Set the pooling time of analog inputs MSB byte
0x3D	POOL_ANA0_TIMERM	R/W	0x03	Set the pooling time of analog inputs Middle byte
0x3E	POOL_ANA0_TIMERL	R/W	0xE8	Set the pooling time of analog inputs LSB byte (1000 milliseconds)
0x3F	POOL_ANA1_TIMERH	R/W	0x00	Set the pooling time of analog inputs MSB byte
0x40	POOL_ANA1_TIMERM	R/W	0x03	Set the pooling time of analog inputs Middle byte
0x41	POOL_ANA1_TIMERL	R/W	0xE8	Set the pooling time of analog inputs LSB byte (1000 milliseconds)
	reserved			
0xFF	*reserved*			

See each register paragraph for full descriptions.

4.1.1 ADDRESS Register

ADRESS	
8 bits	Address of the module

FUTUR release – DO NOT USE, set to 0xFF

4.1.2 CONFIG_DIG bits Register

Bits	Name	Initial State	
0	SET_OUTPUTS_PSU	0	If Set, all relays take previous state when power-up
1	EN_IT_DIG	0	If Set, the DIGITAL INTERUPT message is send if at least a DIGITAL input change
2			
3	EN _POOL_DIG	0	If Set, the pooling timer will send 8 digitals inputs each POOL_DIG_TIMER ms. The down count start on rising edge of this bit.

4.1.3 OUTPUTS Register

OUTPUTS	
8 bits	States of the 8 relay outputs K8 = MSB…. K1 = LSB bit

Turn ON (1) or OFF (0) each output relay.
When turn OFF, the relay are "NO" (Normally Open). When turn ON, the relay are "NC" (Normally Closed).

The register is readable and writable. The register is saved in flash memory and call back on each power up of the board if the bit "SET_OUTPUTS_PSU" is set.

4.1.4 POOL_DIG_TIMERH , M and L Registers

POOL_TIMER	
24 bits	Time in milliseconds between each samples reads of digital inputs

The POOL_DIG_TIMER is only use in combination with the digital inputs. It is a 24 bits register. The LSB bit is equal to 1milliseconde.

4.1.5 CONFIG_ANA bits Register

Bits	Name	Initial State	
0	EN_TH_ANA0	0	If Set, Enable the interrupt on Threshold of ANA0 input
1	EN_TH_ANA1	0	If Set, Enable the interrupt on Threshold of ANA1 input
2	EN _POOL_ANA0	0	If Set, Enable the pooling of ANA0 input
3	EN _POOL_ANA1	0	If Set, Enable the pooling of ANA1 input

4.1.6 POOL_ANA0_TIMERH, M and L Registers

POOL_ANA0_TIMERx	
24 bits	Time in milliseconds between each samples reads of analog inputs

The POOL_ANA0_TIMER is only use in combination with the ANA0 input. It is a 24 bits register. The LSB bit is 1milliseconde.

4.1.7 POOL_ANA1_TIMERH, M and L Registers

POOL_ANA1_TIMERx	
24 bits	Time in milliseconds between each samples reads of analog inputs

Same as POOL_ANA0_TIMER, but for the channel ANA1.

4.1.8 TLH_ANA0 & TLL_ANA0 Registers

TLx_ANA0	
16 bits	Threshold LOW for AN0 channel – only 10 LSB bits are used

4.1.9 THH_ANA0 & THL_ANA0 Registers

THx_ANA0	
16 bits	Threshold HIGH for AN0 channel – only 10 LSB bits are used

4.1.10 TLH_ANA1 & TLL_ANA1 Registers

TLx_ANA0	
16 bits	Threshold LOW for AN1 channel – only 10 LSB bits are used

4.1.11 THH_ANA1 & THL_ANA1 Registers

THx_ANA0	
16 bits	Threshold HIGH for AN1 channel – only 10 LSB bits are used

5 Interrupts, Pooling and Threshold modes

5.1 Interrupts on Digitals inputs

If the interrupt bit (EN_IT_DIG) is set, the USB-IO board will send the state of the 8 inputs if any changes occur on the 8 inputs.
Due to the USB HID driver, the latency of this interrupt may be up to 1 millisecond

The returned frame is as the form of:

Address of the module	*Code Command*	*DIG_IN*
	I	Data byte

5.2 Pooling on Digitals inputs

If the pooling bit (CONFIG_DIG.EN_POOL_DIG) is set, the USB-IO board will send the state of the 8 inputs each "POOLING_DIG_TIMER" millisecond.

The returned frame is as the form of:

Address of the module	*Code Command*	*DIG_IN*
	D	Data byte

The minimal interval period is 1milli second.

5.3 Thresholds on Analogs inputs

If the threshold bit (CONFIG_ANA.EN_TH_ANAx) is set, the USB-IO board will send the value of the analog input, the channel number and the threshold level if the value increase or decrease from the threshold.

The returned frame is as the form of:

Address of the module	*Code Command*	*Channel*	*ANA_IN*	*ANA_IN*	*Threshold*
	J	channel source	MSB Data byte	LSB Data byte	"H" or "L"

The value is coded on the 10 LSB bits.
The Threshold value indicates a High level or Low level. If you don't need one of booth thresholds, simply set it to the extreme value (0x000 or 0x3FF).
If enable, the analog inputs are sampled each 1ms.
To be "armed", the value of the ANAx input need to be between the thresholds at the time you set the EN_TH_ANAx bit.

5.4 Pooling on Analogs inputs

If the pooling bit (CONFIG_ANA.EN_POOL_ANAx) is set, the USB-IO board will send the value of the ANAx input each "POOLING_ANAx_TIMER" millisecond.

The returned frame is as the form of:

Address of the module	*Code Command*	*Channel*	*ANA_IN*	*ANA_IN*
	A	channel source	MSB Data byte	LSB Data byte

The minimal interval period is 1millisecond.

6 Instructions

Each attributes are 8 bits size.

6.1 "0" : Reset

Adr	Cmd	
ADR	0x30 - "0"	

Execute a RESET of the module.

6.2 "1" : Write Register

Adr	Cmd	Attribut1	Attribut2
ADR	0x31 – "1"	ADR register	Data register

Write the data register value to the specified ADR register. The registers are preserved at the power down.

6.3 "2" : Write bit Register

Adr	Cmd	Attribut1	Attribut2	Attribut3
ADR	0x32 – "2"	ADR register	Data register	Bit value

Write the data bit value to the specified bit register. The registers are preserved at the power down.

6.4 "3" : Read Register

Adr	Cmd	Attribut1
ADR	0x33 – "3"	ADR register

Read the register stored at address ADR

The returned frame is as the form of:

Address of the module	*Code Command*	*adr*	*Value*
	3	Adr register	Data value

6.5 "4" : Write FLASH byte

Adr	Cmd	Attribut1	Attribut2	Attribut3
ADR	0x34 – "4"	adrH MSB Address	adrL LSB Address	D

Write the data D byte to the specified Address. The size of the FLASH memory is 1024 bytes. Only the 10 LSB bits of the address are used. The FLASH memory is preserved at the power down

6.6 "5" : Read FLASH byte

Adr	Cmd	Attribut1	Attribut2
ADR	0x35 – "5"	adrH MSB Address	adrL LSB Address

Read the data stored at address A. The size of the FLASH memory is 1024 bytes. Only the 10 LSB bits of the address are used.

The returned frame is as the form of:

Address of the module	*Code Command*	*adrH*	*adrL*	*Value*
	5	MSB	LSB	Data value

6.7 "6" : Read 8 Digitals inputs

Adr	Cmd
ADR	0x36 – "6"

Read the value of the 8 digital inputs pins.

The returned frame is as the form of:

Address of the module	*Code Command*	*Value*
	6	Data value

6.8 "7" : Read Analog input

Adr	Cmd	Attribut1
ADR	0x37 – "7"	Channel

Read the value of the specified channel (0 or 1).

The returned frame is as the form of:

Address of the module	*Code Command*	*Channel*	*Value*	*Value*
	7	channel source	MSB Data value	LSB Data value

6.9 "9" : Read ID

Adr	Cmd
ADR	0x39 – "9"

Read the identification of the board, the version and date build of the firmware.

The returned frame is as the form of:

Address of the module	*Code Command*	*Char 1*	….	*Char N*
	9			

Example of the returned frame: "USB-IO 1.00 – 18/06/2011 23:57:38"

6.10 "X" : Reset factory setting

Adr	Cmd	Attribut1	Attribut2
ADR	0x58 – "X"	0xAA	0x55

This command reset all registers in the initial factory settings. The new values are available after next power up.
It will not erase the flash data user area.

7 Software installation

These paragraphs are intended to detail the software installation steps as well as describe what is being installed.
The software provided with this board need to be downloading and ***must be installing onto your hard disk prior to use.*** To do this, perform the following steps as appropriate for your operating system.

Win2000/XP/Vista/Seven

a. download the install file from http://www.eko-fpga.com
b. run the installation
c. Follow the on-screen prompts to install the software for this board.

If you don't have the Microsoft .NET4.0 Framework already installed, the installation software will try to download it.

The demonstration software, sources project (VB.NET Framework 4.0) and DLL are provided as free of charges.

END OF DOCUMENT