



---

# USB-IO

## Reference Manual

document code: USB-IO-Eng.doc

release: 1.3 -26/04/2006



---

# Summary

---

<b>Summary .....</b>	<b>2</b>
<b>Revisions .....</b>	<b>3</b>
<b>1. Functional Description.....</b>	<b>4</b>
<b>1.1 Main specifications.....</b>	<b>4</b>
<b>1.2 Technical specifications .....</b>	<b>5</b>
<b>1.3 Software compatibility.....</b>	<b>5</b>
<b>2. Driver installation.....</b>	<b>6</b>
<b>2.1 Microsoft® Windows.....</b>	<b>6</b>
2.1.1 Windows installation test.....	10
<b>2.2 Linux.....</b>	<b>10</b>
<b>3. Software examples .....</b>	<b>11</b>
<b>3.1 TEST_IO.C .....</b>	<b>11</b>
<b>3.2 TUSB.C.....</b>	<b>11</b>
<b>3.2.1 Callable functions list.....</b>	<b>12</b>
<b>4. Installation .....</b>	<b>14</b>
<b>4.1 Wiring instructions.....</b>	<b>14</b>
4.1.1 Power supply.....	15
4.1.2 USB ports.....	15
4.1.3 Input connections.....	16
4.1.3.1 Sample input diagrams.....	17
4.1.4 Output connections.....	18
4.1.4.1 Sample output diagrams.....	20
<b>4.2 Status indicators .....</b>	<b>21</b>
<b>4.4 Mounting instructions .....</b>	<b>24</b>
<b>Dichiarazione di conformità.....</b>	<b>25</b>



---

# Revisions

---

Rev	Date	Description	Written	Verified	Approved
1.0	12-Apr-2004	initial release	Prandini	Lazzari	Prandini
1.1	10-Oct-2004	electrical connections	Botturi	Lazzari	Prandini
1.2	13-Sep-2005	software examples	Botturi	Lazzari	Prandini
1.3	26-Apr-2006	LED numbers fixed	Prandini	Lazzari	Prandini

---

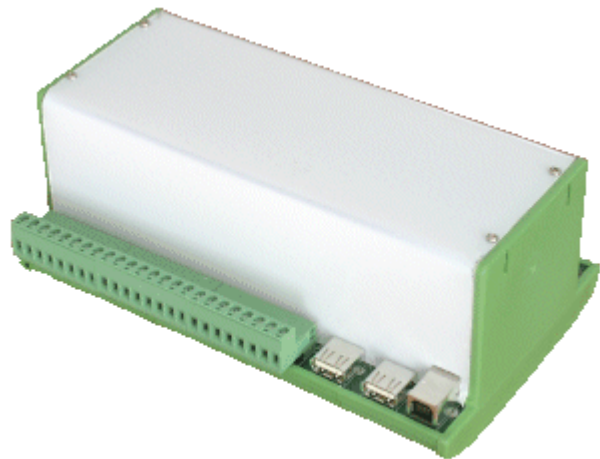
# 1. Functional Description

---

## 1.1 Main specifications

The capability to read digital signals and drive power outputs from a PC in an industrial environment is really useful in many cases ; therefore such a product needs to satisfy more and more demanding specifications in terms of ease of use and reliability.

USB-IO is surely the answer to the needs of such demanding environments ; the integrated USB hub, the snap-in contacts, the simple and easy to use software libraries that don't need special drivers, are surely useful features that greatly reduce the time needed to install and operate our device.



A special attention has been paid in the safety features, like in the emergency signal handling, both with software and hardware controls, or in the power supply where the galvanic isolation greatly improves the protection from ground loops and electromagnetic disturbances.

An electrical interlock assures that in every operating condition (even with a board failure) all the power outputs are disabled when the emergency signal is activated; this feature greatly improves the safety of the system over competitive products.

Through the embedded USB interface all board functions can be easily controlled; this choice guarantees high speed, host platform independence, easy connection of multiple boards for system expansion. Up to 256 boards can be daisy chained for a maximum of 4096 inputs and 4096 outputs.

USB-IO is wholly manufactured and engineered in Italy, this assures an high quality level and durability, but also availability over the years of the product itself.

USB-IO is available in 5 versions:

- USB-IO-I with 16 inputs only
- USB-IO-U16 with 16 outputs, s.p.s.t. normally open contacts
- USB-IO-U32 with 16 outputs, s.p.d.t. contacts
- USB-IO-16 with 16 outputs, s.p.s.t. normally open contacts and 16 inputs
- USB-IO-32 with 16 outputs, s.p.d.t. contacts and 16 inputs

---

## 1.2 Technical specifications

Size:	188 x 130 x 90 mm
Power supply:	22-27 V DC
Power consumption:	8 W
Operating temperature:	0 to +60°C
Storage temperature:	-40° to +85°C
Inputs count:	17
Input configuration:	optocoupler with max. operating isolation voltage 565 V peak to peak minimum isolation voltage: between inputs and +24 V supply 1000 V DC
Input connectors:	multipolar plug connectors nominal current: 12 A, rated voltage: 250 V, pitch: 5.0 mm
Input voltage:	24 V DC
Outputs count:	16
Output type:	relay contact s.p.s.t. n.o. 2 A 250 V AC/DC optionally s.p.d.t. n.o. and n.c. minimum isolation voltage: between contacts 1000 V DC, between coil and contacts 6000 V DC
Output connectors:	multipolar plug connectors nominal current: 12 A, rated voltage: 250 V, pitch: 5.0 mm
Serial port:	EIA/TIA 561, 9600 bps 8 bit no parity 1 stop bit
USB ports:	1 Universal Serial Bus receptacle (Series B), right angle with rear shell 2 Universal Serial Bus receptacle (Series A), right angle without rear shell integrated USB 1.1 hub fully compliant with the USB release 1.1 specification supports 1.5 and 12 Mbits/s USB data rate

## 1.3 Software compatibility

All Microsoft® operating systems starting from Windows 98/ME, except for Windows NT, have built-in support for USB devices ; USB-IO is therefore compatible with Microsoft® Windows XP Home, Microsoft® Windows XP Professional, Microsoft® Windows 2003 Server, Microsoft® Windows 2000 Server and Professional, Microsoft® Windows 98/ME. Linux kernel releases with USB support compiled in or as a module can easily be connected to USB-IO as well.

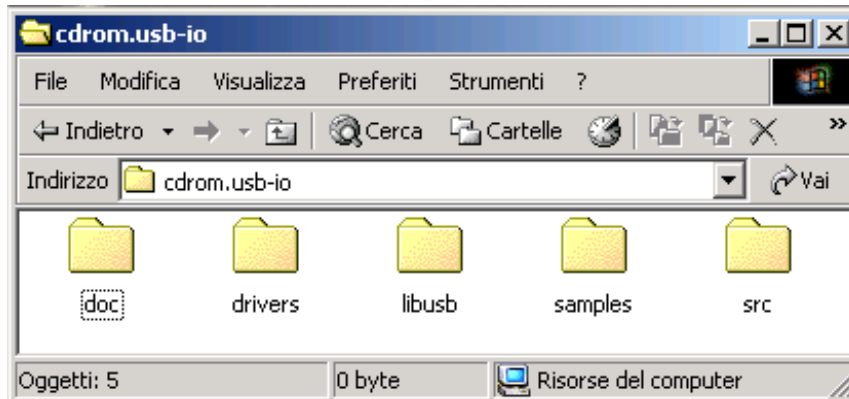
---

## 2. Driver installation

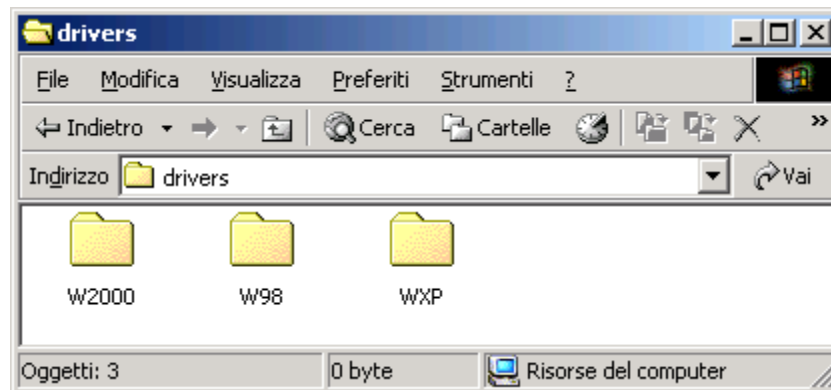
---

### 2.1 Microsoft® Windows

To operate the USB-IO board with the Microsoft Windows operating system it is mandatory to install the software drivers that are delivered in the installation CD. Each supported operating system version has its own directory, as shown below:



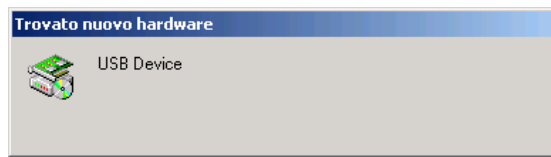
Now entering the *drivers* directory:



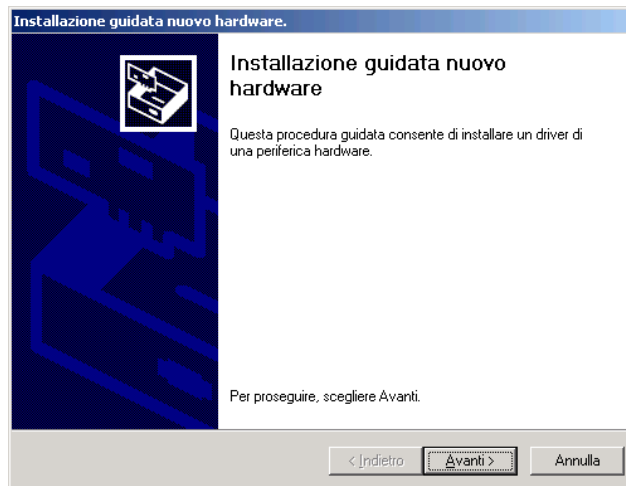
The W2000 directory must be used for the Windows 2000 versions ( Windows 2000 Professional, Windows 2000 Server, Windows 2003 Server ), the W98 directory for Windows 98 and ME, the WXP directory for the Windows XP versions ( Home and Professional).

The installation is really very simple, it is enough to follow in the exact order the following steps:

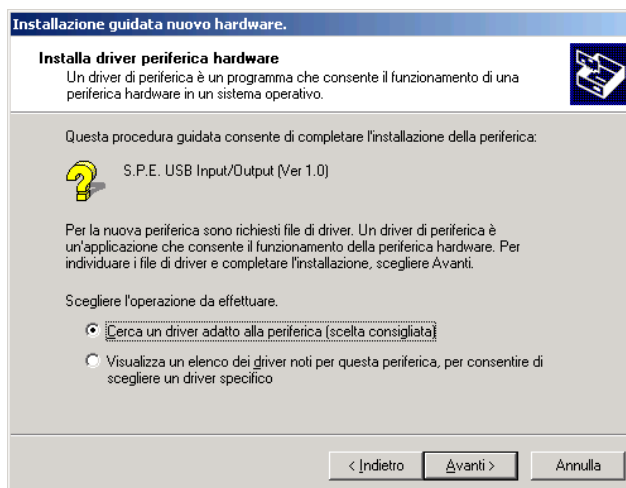
- 1) Connect the unpowered USB-IO board using the cable supplied with the board to an USB host port.
- 2) Power up the board.
- 3) Wait for the automatic operating system detection of new hardware:



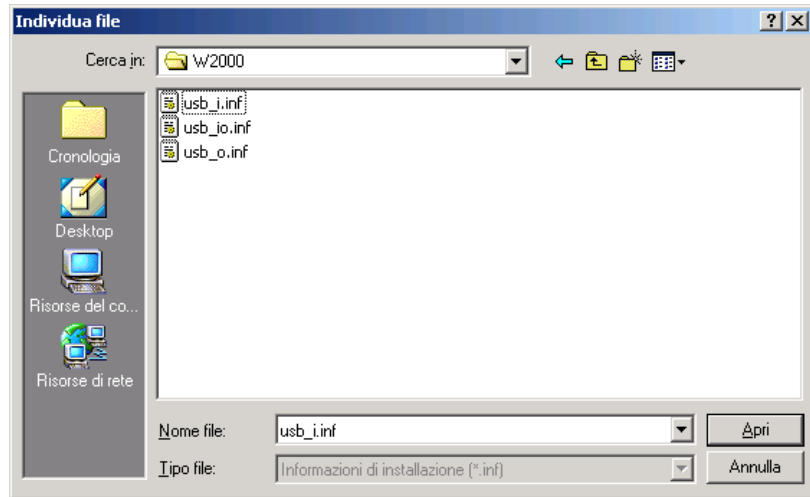
4) When asked for installation of the driver for a new hardware peripheral, confirm the default choice:



5) Now the board type is shown and Windows asks for the driver location:



6) According to the USB-IO board type select the right .inf file from the operating system specific directory in the installation CD. scegliere l'opportuno file .inf dalla directory drivers del CD di installazione.



7) The right installation file is selected according to the following table:

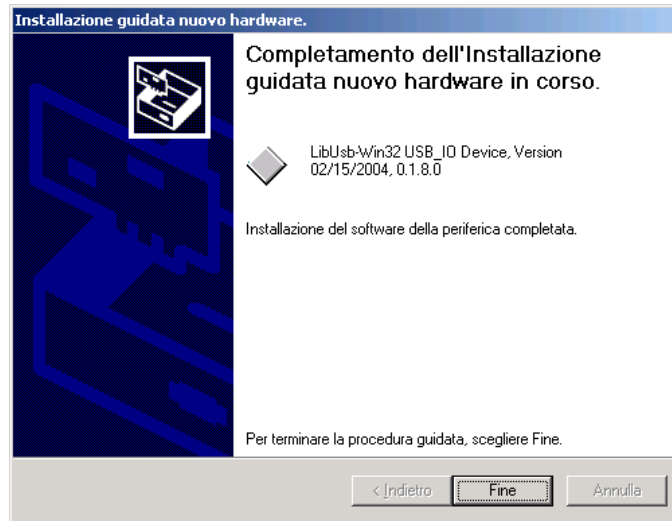
Board type	Installation information file
USB-IO-I	usb_i.inf
USB-IO-U16	usb_o.inf
USB-IO-U32	usb_o.inf
USB-IO-16	usb_io.inf
USB-IO-32	usb_io.inf

8) Confirm the choice, Windows will prompt for final installation, please select the default choice for completion.



9) After a while Windows will confirm the setup; the board is then recognized and usable by the application programs.

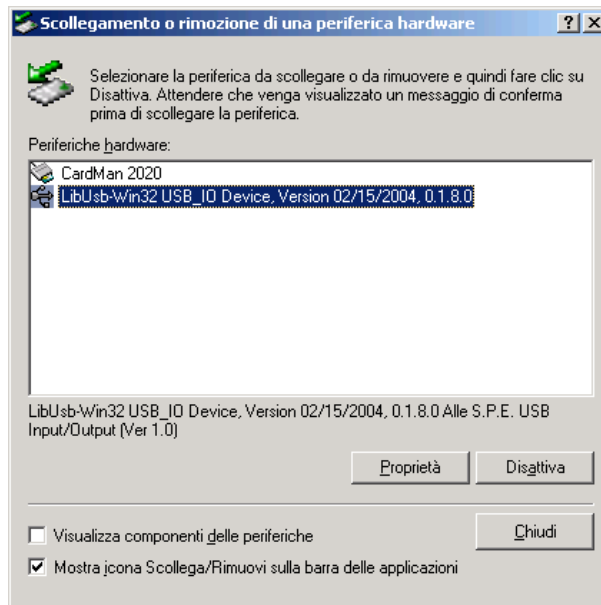




Please remember that when disconnecting the USB-IO board the Windows procedure must be followed to avoid reconnection problems; simply double click the icon shown on the toolbar as follows:



The disconnection menu will appear:

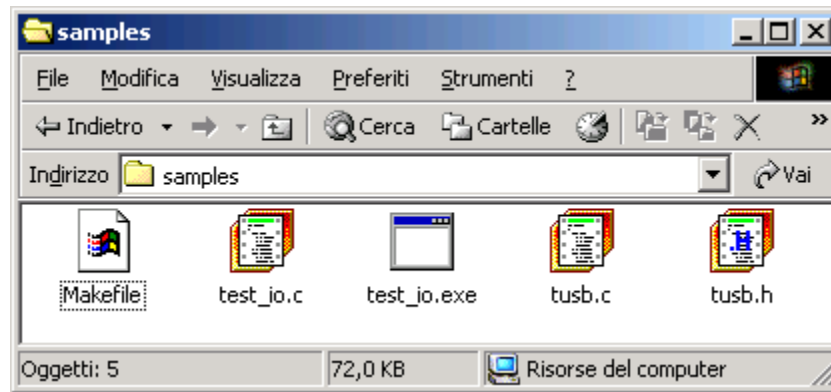


Select the LibUsb-Win32 USB\_IO Device as shown on the picture above and then press the Deactivate button, confirming the choice on the following window. The Windows operating system will then show a further message when the USB-IO board can be effectively disconnected.

---

## 2.1.1 Windows installation test

In the *samples* directory on the installation CD are stored both the source code and the executable code of TEST\_IO program that can be used for testing the correct operation of the board and to have a complete working example.



The program operation is quite simple, when it is running every output from 1 to 16 is switched on or off following the state of the input with the same number ( output 0 copies the state of input 0 and so on).

## 2.2 Linux

To be completed.

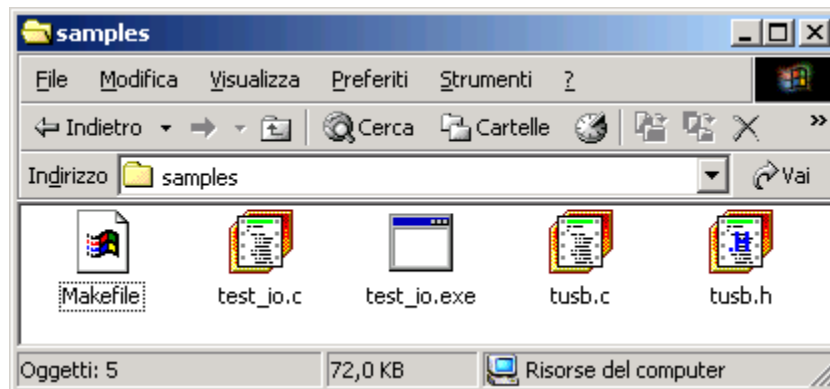
---

## 3. Software examples

---

### 3.1 TEST\_IO.C

In the *samples* directory on the installation CD are stored both the source code and the executable code of TEST\_IO program that can be used for testing the correct operation of the board and to have a complete working example.

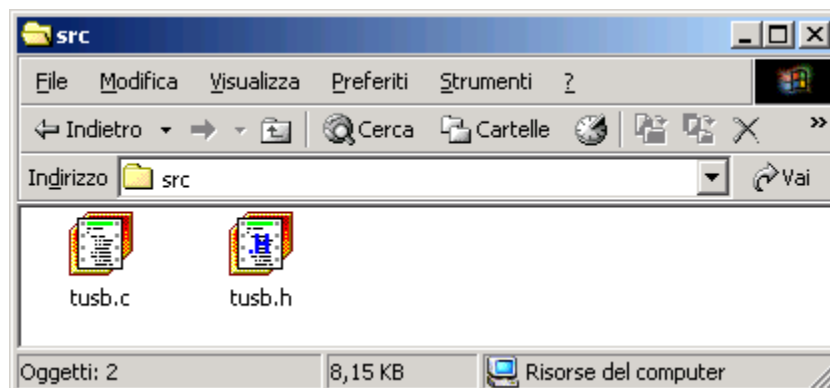


The program operation is quite simple, when it is running every output from 1 to 16 is switched on or off following the state of the input with the same number ( output 0 copies the state of input 0 and so on).

**In the directory there is also a Makefile for rebuilding the application using the Borland C++ BC55 compiler that can be freely downloaded from Borland site.** The C source listing shows clearly how to use all functions call that are delivered by the driver interface library that is implemented in the *TUSB.C* file.

### 3.2 TUSB.C

In the *src* directory on the installation CD are stored the source code and the include file of the TUSB.C interface library that can be used for driving all the card features.



---

## 3.2.1 Callable functions list

- **tusbInitIO(void)**: it finds all connected boards and starts the connection between PC and USB-IO. It returns the number of boards it found. It has to be invoked before all other calls.  
Parameters: none  
Return value: 0 if no connected boards are found, otherwise the boards count.
- **tusbCloseAll(void)**: closes the connection between PC and USB-IO. It has to be invoked at program termination.  
Parameters: none  
Return value: none.
- **getNumBoard(void)**: gives back the connected boards count.  
Parameters: none  
Return value: 0 if no connected boards are found, otherwise the boards count.
- **tusbChk(void)**: checks if any change happened in the connected boards count or in the order of the USB-IO devices on the USB bus and gives back the number of changes found.  
Parameters: none  
Return value: 0 if no changes are found, otherwise the changes count.
- **getUsbLastErr(void)**: gives back a data structure with detailed informations about any errors occurred during the previous library function call.  
Parameters: none  
Return value: an *usbLastErr* struct with the error number, the function call and the MAC of board where the error occurred.
- **getBoardManufacturer(char \*mac, char \*mname, int len)**: stores in the *mname* array the board manufacturer's name.  
Parameters: **mac** = MAC of the board to be polled  
**mname** = array where the manufacturer's name is stored  
**len** = maximum length of *mname* array  
Return value: 0 when successful, otherwise the error code
- **getBoardDescription(char \*mac, char \*dname, int len)**: stores in the *dname* array the board name.  
Parameters: **mac** = MAC of the board to be polled  
**dname** = array where the board name is stored  
**len** = maximum length of *dname* array  
Return value: 0 when successful, otherwise the error code
- **setBoardUsbTimeout(char \*mac, int tout)**: enables the communications watchdog between PC and USB-IO and sets the timeout in 10 millisecond steps. If there are no commands exchanged with the board within the set time all outputs are automatically switched off.  
Parameters: **mac** = MAC of the board to be polled  
**tout** = timeout in 10 milliseconds steps  
Return value: 0 when successful, otherwise the error code



- 
- **setBoardOutputs(char \*mac, BYTE \*bout):** switches the board outputs as defined in the *bout* array.  
Parameters: **mac** = MAC of the board to be polled  
**bout** = three bytes array, as follows  
    **bout[0]**=> activation flag for output refresh ( must be 0x01)  
    **bout[1]**=> values for outputs 0 to 7  
    **bout[2]**=> values for outputs 8 to 15  
Return value: 0 when successful, otherwise the error code
  
  - **getBoardInputs(char \*mac, BYTE \*binp):** reads the board inputs and stores them in the *binp* array.  
Parameters: **mac** = MAC of the board to be polled  
**binp** = eight bytes array, as follows  
    **binp[0]**=>  
        bit 7 : when set indicates this is the first readout after powerup  
        bit 0 : emergency input state  
    **binp[1]**=> inputs 0 to 7 state  
    **binp[2]**=> inputs 8 to 15 state  
    **binp[3]**=> each bit is set when there was a change on the inputs state  
        sinche last poll ( bit 0 for input 0, and so on up to 7 for input 7)  
    **binp[4]**=> each bit is set when there was a change on the inputs state  
        sinche last poll ( bit 0 for input 8, and so on up to 7 for input 15)  
    **binp[5]**=> outputs 0 to 7 state  
    **binp[6]**=> activation flag for output refresh state  
    **binp[7]**=> outputs 8 to 15 state  
Return value: 0 when successful, otherwise the error code

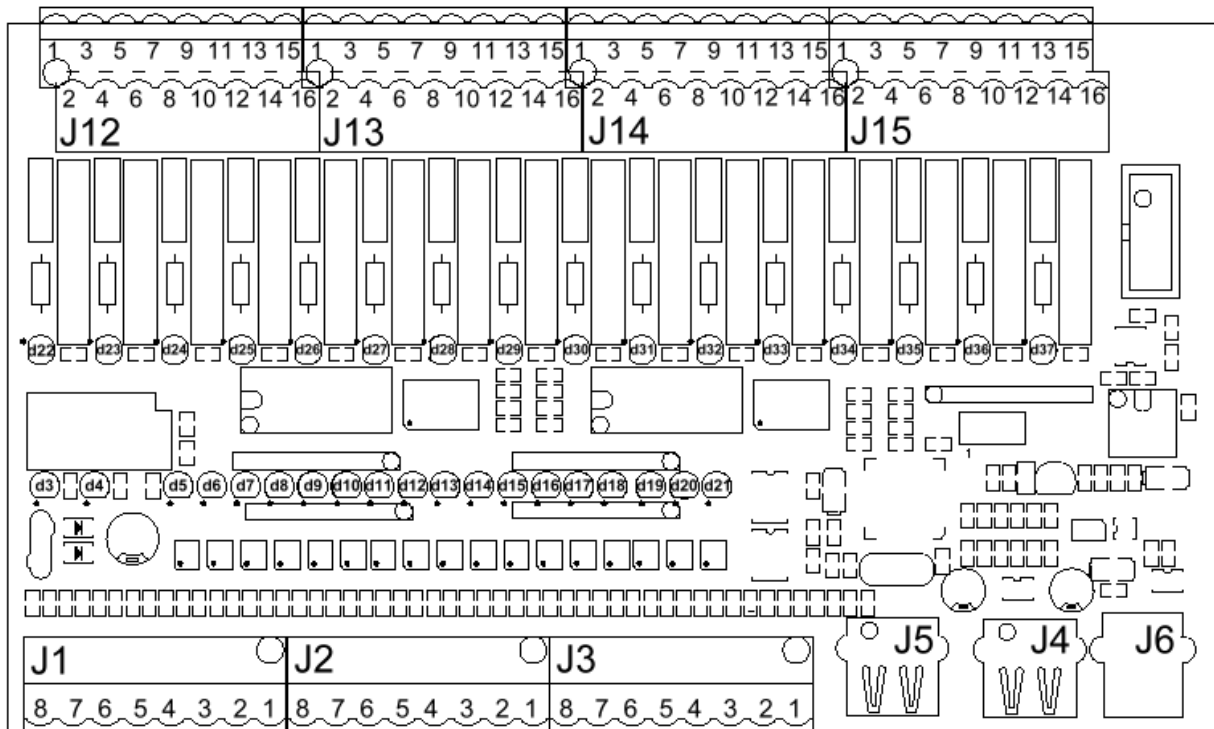
---

# 4. Installation

---

## 4.1 Wiring instructions

USB-IO can have up to 10 connectors, as shown in the picture below.



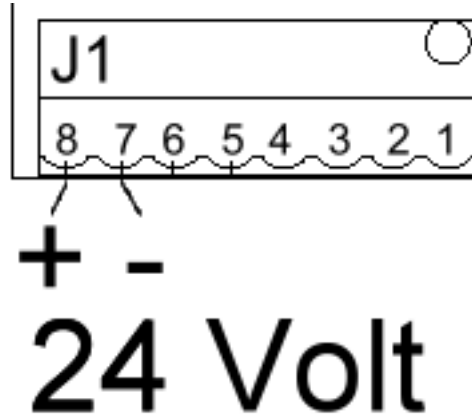
The input-only version USB-IO-I doesn't have the J12..J15 connectors, that are dedicated to outputs; the USB-IO-32 and USB-IO-U32 versions, that have an output s.p.d.t. switch, use both rows on two levels of the J12..J15 connectors, while the s.p.s.t. versions use only the odd numbered pins on the lower level.

Apart from the J4..J6 USB ports, all the other connectors are multipolar plug connectors with nominal current 12 A, rated voltage 250 V and pitch 5.0 mm ; they can be used with wire AWG 24-12 or 0,2-2,5 mm<sup>2</sup> ; it is recommended to use a torque between 0,5 and 0,6 Nm.

---

## 4.1.1 Power supply

The power supply voltage must be between +22 Volt and +27 Volt D.C. , with a maximum power of 8 Watt; it must be connected on the J1 connector as per the following diagram.

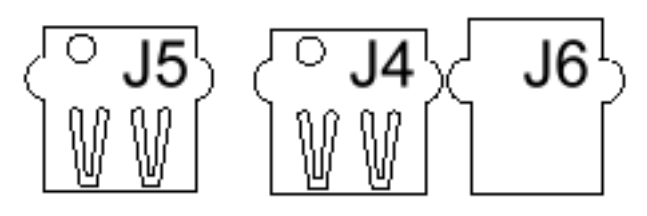


There must be a protection device for current limiting on the power source (e.g. a 500 mA fuse) and a switching device ( e.g. s.p.s.t. level switch ) for disconnecting USB-IO from the power source.

## 4.1.2 USB ports

The USB-IO board includes a self powered USB 1.1 two port hub. No power is drained from the host and no power is delivered by the board itself to the 2 output USB ports; the power supply from the host is simply redirected to the output ports, limiting the output current to a maximum of 500 mA for appropriate protection.

No galvanic isolation is present between the USB ports and between them and the internal circuitry of USB-IO ; but they are isolated from all the other inputs and outputs. The USB ground is connected with the ground of the power supply, but not with the inputs or outputs ground.



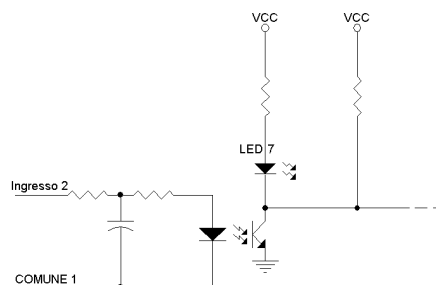
The J6 connector is an Universal Serial Bus receptacle (Series B), right angle with rear shell, for connection to the host, while the two hub output ports are Universal Serial Bus receptacle (Series A), right angle without rear shell.

### 4.1.3 Input connections

On the same side where are the power supply connections there are also the input signals. They can be found on connectors from J1 to J3 and they are described in the following table.

Connector	Pin	LED	Name
J3	1	5	Input 0
	2	6	Input 1
	3	7	Input 2
	4	8	Input 3
	5	9	Input 4
	6	10	Input 5
	7	11	Input 6
	8	12	Input 7
J2	1		Common 1 (inputs 0-7)
	2		Common 1 (inputs 0-7)
	3	13	Input 8
	4	14	Input 9
	5	15	Input 10
	6	16	Input 11
	7	17	Input 12
	8	18	Input 13
J1	1	19	Input 14
	2	20	Input 15
	3		Common 2 (inputs 8-15)
	4		Common 2 (inputs 8-15)
	5	21	IEM (emergency)
	6		Common for IEM
	7		-24 V DC
	8		+24 V DC

The input schematic is shown below; an optocoupler galvanically isolates inputs and internal USB circuitry, protecting therefore the host PC from disturbances and potentially hazardous voltages.

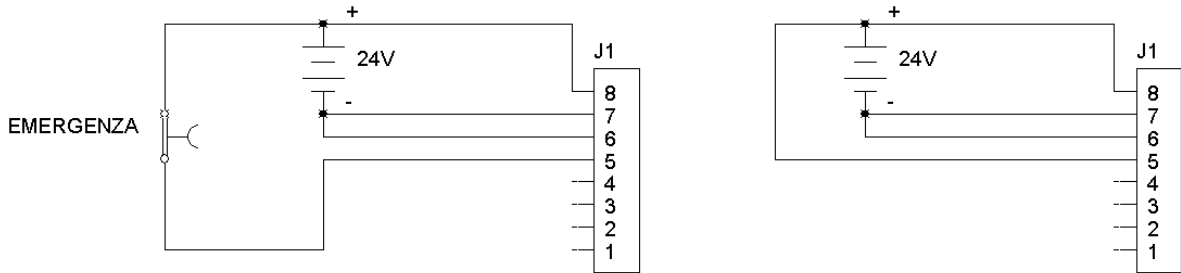




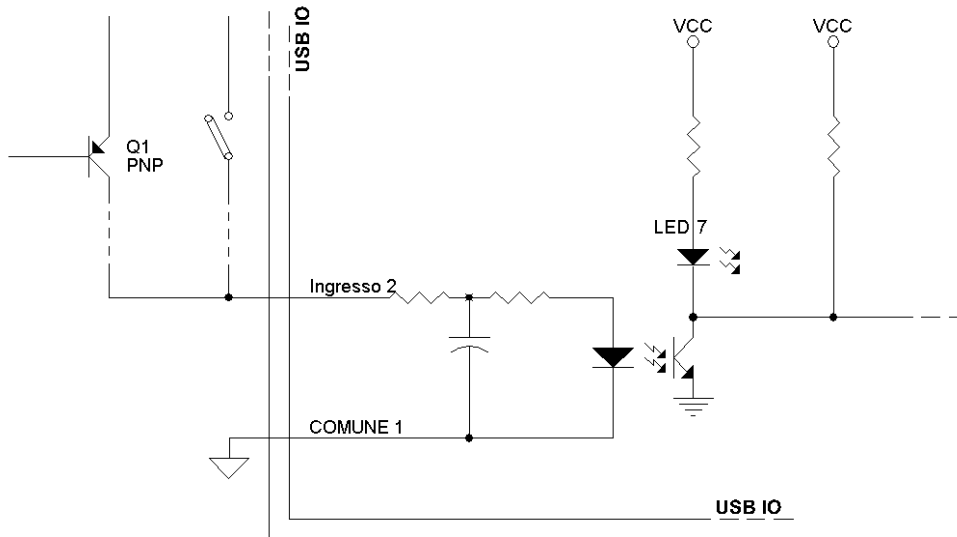
### 4.1.3.1 Sample input diagrams

There are two possible configurations for the input connections, when there is an emergency pushbutton or when there is not.

On the left side of the following picture it is shown the input connection of the IEM signal with an emergency pushbutton ; on the right the IEM signal is fixed.



The other input signals can be connected in many different ways; the more common one is obtained by tying the Common signal to a power ground and driving the input pin with a PNP driver or switch connected to a power positive supply in the operating range, as shown below.



---

## 4.1.4 Output connections

On the opposite side of the power supply connections there are the output signals. They can be found on connectors from J12 to J15 and they are described in the following table for the USB-IO-32 and USB-IO-U32. Each connector has 16 connections on 2 levels of 8 signals ; the odd ones are on the lower level, the even ones are on the higher level.

Connector	Pin	LED	Name
J12	1	22	COM0
	2	22	COM0
	3	22	NO0
	4	22	NC0
	5	23	COM1
	6	23	COM1
	7	23	NO1
	8	23	NC1
	9	24	COM2
	10	24	COM2
	11	24	NO2
	12	24	NC2
	13	25	COM3
	14	25	COM3
	15	25	NO3
	16	25	NC3
J13	1	26	COM4
	2	26	COM4
	3	26	NO4
	4	26	NC4
	5	27	COM5
	6	27	COM5
	7	27	NO5
	8	27	NC5
	9	28	COM6
	10	28	COM6
	11	28	NO6
	12	28	NC6
	13	29	COM7
	14	29	COM7
	15	29	NO7
	16	29	NC7
J14	1	30	COM8
	2	30	COM8
	3	30	NO8
	4	30	NC8
	5	31	COM9
	6	31	COM9
	7	31	NO9
	8	31	NC9
	9	32	COM10
	10	32	COM10
	11	32	NO10
	12	32	NC10



	13	33	COM11
	14	33	COM11
	15	33	NO11
	16	33	NC11
J15	1	34	COM12
	2	34	COM12
	3	34	NO12
	4	34	NC12
	5	35	COM13
	6	35	COM13
	7	35	NO13
	8	35	NC13
	9	36	COM14
	10	36	COM14
	11	36	NO14
	12	36	NC14
	13	37	COM15
	14	37	COM15
	15	37	NO15
	16	37	NC15

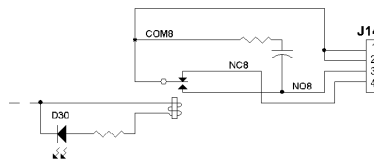
In the above table the COMxx legend indicates the common connection of the single pole dual throw contact of the relay identified by the output number xx ; NOxx is the normally open contact and NCxx is the normally closed contact.

The USB-IO-I device has no outputs and the J12..J15 connectors are simply not mounted on the board; the USB-IO-16 and USB-IO-U16 devices have only the odd numbered pins, as per the following table, because only the normally open contact is available.

Connector	Pin	LED	Name
J12	1	22	COM0
	3	22	NO0
	5	23	COM1
	7	23	NO1
	9	24	COM2
	11	24	NO2
	13	25	COM3
	15	25	NO3
J13	1	26	COM4
	3	26	NO4
	5	27	COM5
	7	27	NO5
	9	28	COM6
	11	28	NO6
	13	29	COM7
	15	29	NO7
J14	1	30	COM8
	3	30	NO8
	5	31	COM9
	7	31	NO9
	9	32	COM10
	11	32	NO10
	13	33	COM11
	15	33	NO11

J15	1	34	COM12
	3	34	NO12
	5	35	COM13
	7	35	NO13
	9	36	COM14
	11	36	NO14
	13	37	COM15
	15	37	NO15

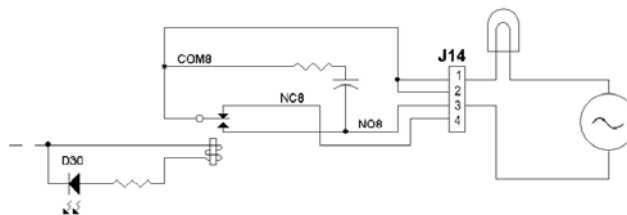
The output schematic is shown below; an output RC snubber connected in parallel to the contact is indicated. This snubber is normally not installed on the board, but can be added on request.



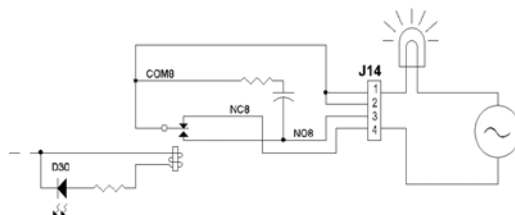
#### 4.1.4.1 Sample output diagrams

There are three possible configurations for the output connections, using the normally open or normally closed s.p.s.t contact or switching between loads with the s.p.d.t. contact.

The output schematic is shown below in a normally open configuration.



The output schematic is shown below in a normally closed configuration.



## 4.2 Status indicators

On the USB-IO there are many LEDs , quite useful for finding the actual board status.  
Some of them when lit show the power supply status:

- d3 external power supply ( +24 V on the input connector ) is present
- d4 internal power supply (+5 V after the DC/DC converter) is present

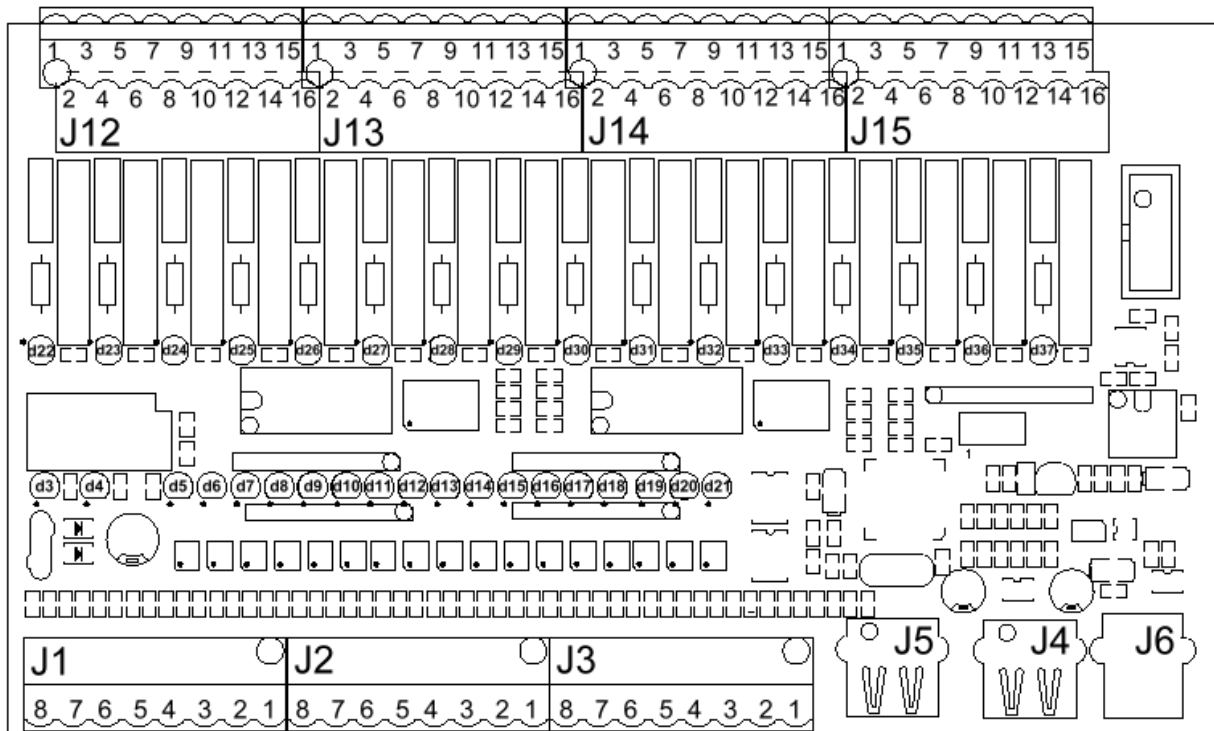
Then there are 17 more for the inputs:

- d5 IEM (emergency ) input active
- d21..d6 input 0..15 is active

And finally 16 LEDs for the outputs:

- d22..d37 output 0..15 is active

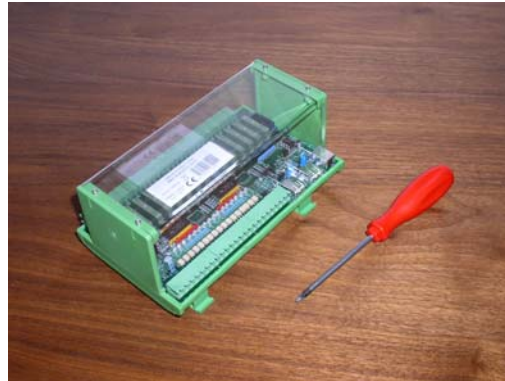
The position and number of the LEDs can easily be found on the following picture:



---

## 4.3 Case opening instructions

The USB-IO board is normally delivered in a wholly enclosed plastic case ; in normal operation is not required to open the case, because there are no user serviceable parts inside. If for any reason it is needed to open the case, switch off any power source, remove all the connectors and then follow the instructions detailed in the pictures below, after removing the USB-IO device from the DIN rail as described in 4.4:



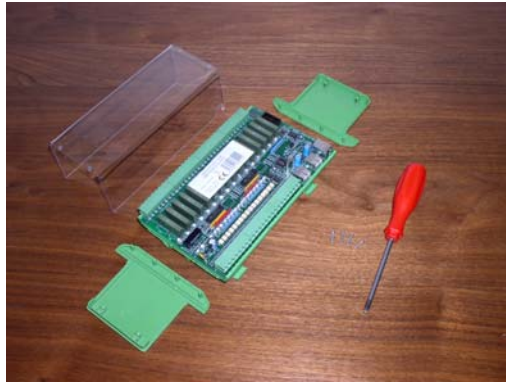
First of all get a Phillips screwdriver with a 4,5 mm tip.



Remove the 4 self threading screws that hold the transparent protective cover.



Now carefully remove the protective cover. Of course be careful not to lose track of the removed screws or of the other disassembled parts!



The lateral walls can easily be removed by gently pulling them free of the holding pins.



Now slide out the board from the plastic mounting rails.

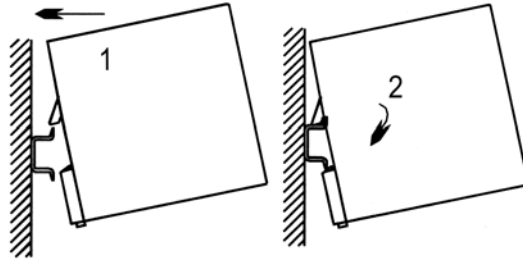


The USB-I/O is fully disassembled in all its parts.

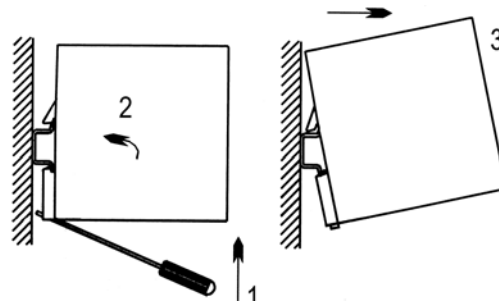
---

## 4.4 Mounting instructions

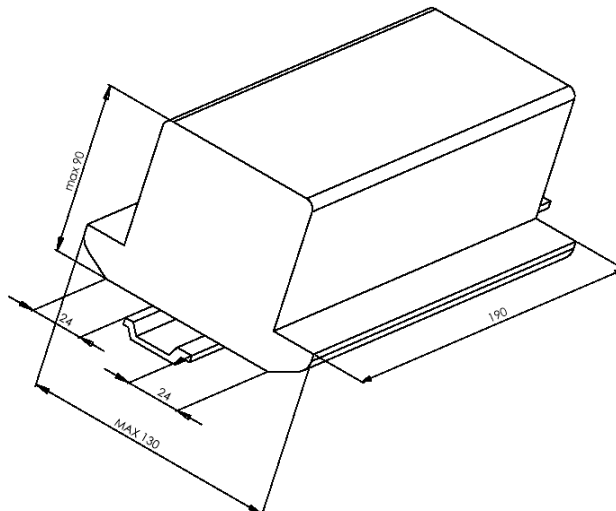
It is quite simple to mount the USB-IO device on a DIN rail: as shown on the following picture, first of all simply engage the top pin over the rail while keeping the unit slightly inclined; then lock the lower movable pin on the bottom of the rail with a swift clockwise movement .



To remove the USB-IO device from the DIN rail please follow the instructions as illustrated in the picture below: using a flat blade screwdriver or similar pull the lower locking level by leveraging on the lower case edge ; then carefully rotate the whole device counterclockwise until the lower retaining pin is unlocked from the rail ; now simply remove the USB-IO from the rail by pulling it.



In the following picture the main measurements are shown in mm, in order to carefully plan enough space around the device and be able to correctly mount it.







CE

**Dichiarazione di conformità**  
secondo EN 45014

Assumendone la piena responsabilità, dichiariamo che il prodotto:

**USBI-IO**

cui è relativa la presente dichiarazione è conforme alle seguenti normative ed ai relativi documenti:

- a) EN 55022 [ CEI 110-5 (88) ]
- b) EN 50082-1 [ CEI 110-8 (92) ]

in base alle prescrizioni delle direttive 89/336/EEC e 93/68/EEC.

Luogo di emissione: ITALIA  
Data di emissione: 12 Aprile 2004

In fede,