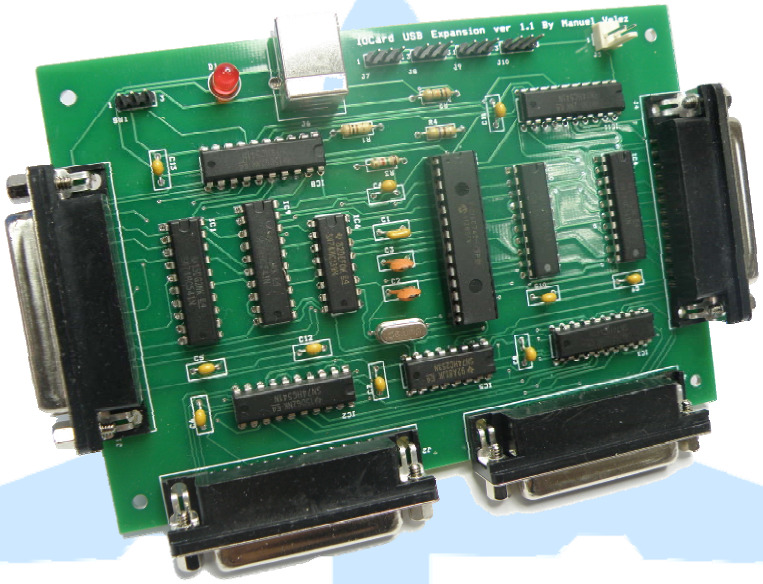
Date:15/11/13
Rev.:2.1

Opencockpits



IOCards USB Expansion & Master Manual

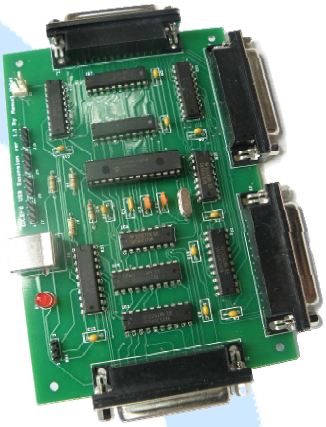
Index:

IOCARDS USB EXPANSION & MASTER MANUAL 1
INDEX: 2
INTRODUCTION: 3
USBEXPANSION: 3
Outline and component: *3*
Description of the connectors: *4*
MASTER: 4
Outline and component: *5*
Description of the connectors: *5*
INPUTS: 6
Connector J3: *6*
Connector J4: *6*
OUTPUTS: 7
Connector J2: *7*
Connector P2: *8*
DISPLAYS: 9
STARTING OF THE CARDS: 9
INSTALLATION OF SOFTWARE AND CONFIGURATION: 10
Proof of functioning of the cards: *10*
LINKS OF INTEREST: 16

Introduction:

The USBExpansion was designed specifically to connect to the USB port and so it improves the facility of configuration and expansion of the group of cards that compose an installation with Opencockpits IOCards, due to that the modern computers not include parallel port, the Master IOCard cannot be connected directly to the PC, therefore in this manual We explain to You jointly the USBExpansion and the Master cards.

UsbExpansion:



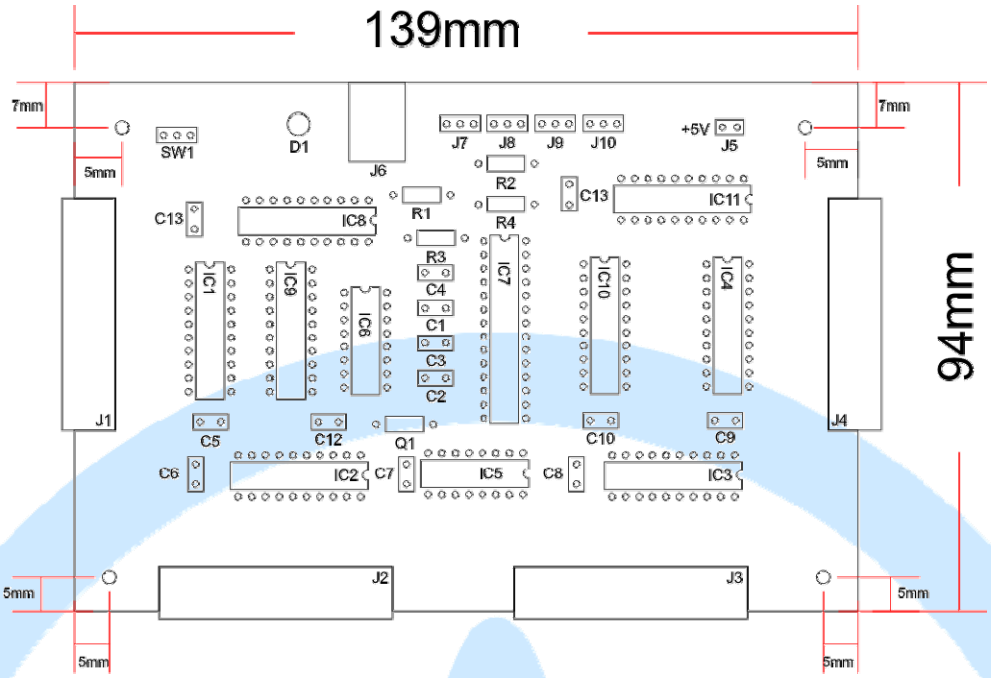
This card permits the connection to a USB port of the computer until 4 Master IOCards. It is possible it moreovers to connect several UsbExpansion cards in a same computer with which the power is multiplied for each USB port available, is ideal to promote big panels that need several Master IOCards.

Moreover with your 4 analogical entrances it is possible to promote 4 axises as can be those of throttle, column stick or the pedals.

If you want to move away the computer of the panels area, it is possible to take a long USB cable until the UsbExpansion and from this distribute all the cables to bring near the Master IOCards to the different systems, by obtaining a better distributed cables, for the whole cabin is less prone to electrical "noises". Moreover it is possible use USB hubs with which can perfectly distribute the whole electronic thing in the cabin.

Outline and component:

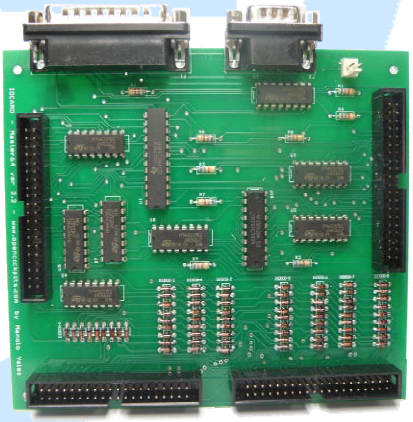
- J1, J2, J3 Y J4 = CONNECTOR DB25 FEMALE
- SW1, J7, J8, J9 Y J10 = REGLET 3 PINS MALE
- D1 = RED LED
- J6 = CONNECTOR FEMALE USB (SQUARE)
- J5 = REGLET 2 PINS 5V+ POWER OUTPUT
- C1 = CERAMIC CONDENSER 220nF
- C2 Y C3 = CERAMIC CONDENSER 22pF
- C4 ..C13 = CONDENSER 0,1mF
- Q1 = QUARTZ CRYSTAL 6 Mhz
- R1 = 100R
- R2 = 10K
- R3 = 1K5
- R4 = 470R
- IC1, IC2, IC3, IC4, IC8, IC9, IC10, IC11 = 74HC541
- IC5 = 74HC253
- IC6 = 74HC139
- IC7 = 16C745



Description of the connectors:

• J6 = USB connector towars the computer
• J1, J2, J3, J4 = connectors for the different Master cards, being the order of connection of the Master cards the same order of the connectors
• J7, J8, J9, J10 = connectors for analogical entrances, in them We can connect potentiometers.
• J5 = power connector, it is a 5V output, although it is not necessary to feed the USB Expansion, the USB Expansion takes power from USB port.
• SW1 = Microchip reset (not used habitually).

Master:



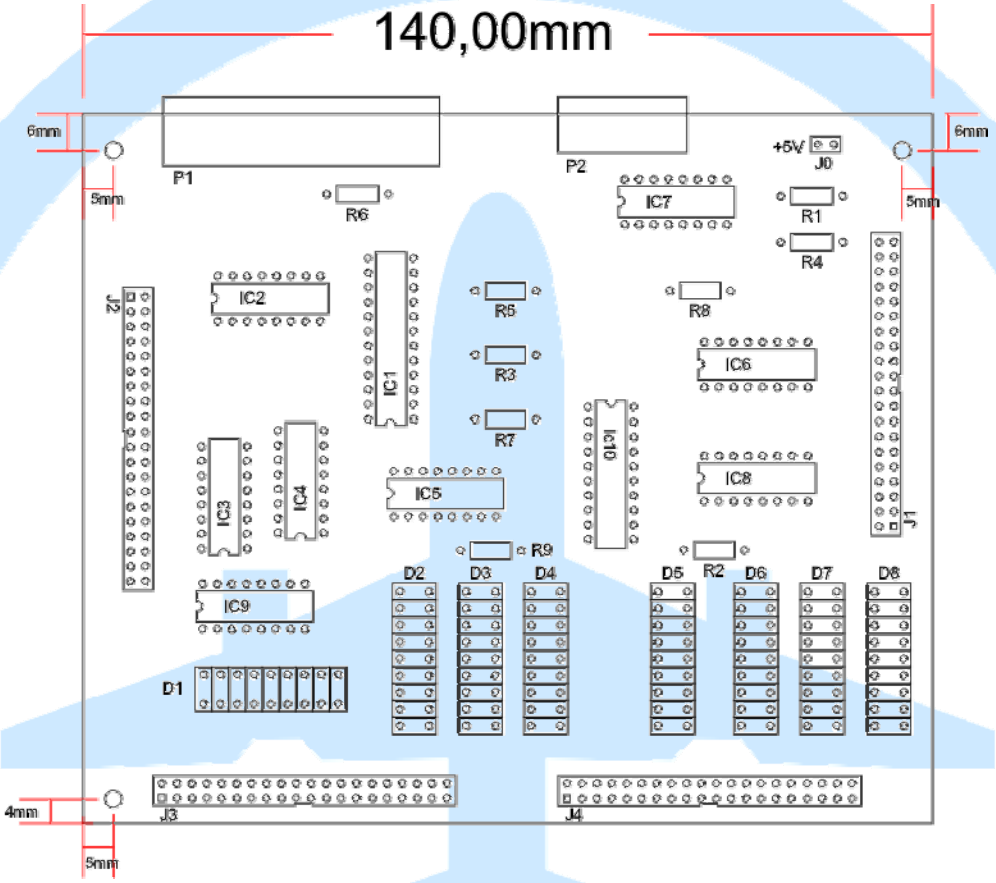
This card is in charge to gather the changes that come of the keys, switches, encoders, etc. and it activates the exits of the data of the simulator in the leds, lcd´s screens, displays of 7 segments, etc.

The card has several connectors with a total of 64 outputs and 72 inputs.

If you has bought the kit and hasn´t experience by soldering plates can see a tutorial in the Opencockpits site.

Outline and component:

- P1 = CONNECTOR DB25 MALE
- P2 = CONNECTOR DB9 MALE
- J0 = REGLET 2 PINS +5V INPUT POWER SUPPLY
- J1, J2, J3 Y J4 = CONNECTORS 40 PINS BT224
- DIODOS 1..8 = 8 DIODE GROUP 1N4148 (72 TOTAL)
- R1..R9 = RESISTORS 6K8 0,33W
- IC1 = 74HC154N
- IC2..IC9 = 74H259
- IC10 = 74HC541



Description of the connectors:

• P1 = connector towards the USBExpansion, permits us the connection with the computer.
• P2 = connector for seven auxiliary outputs.
• J0 = connector for feeding, the Master must be feeded always for this connector, being the more near pin to the P2 connector the positive.
• J1 = bus of communications for the DISPLAYS II cards, permits connect us until 4 cards in series, it permits manage us a total of 64 digits of 7 segments.
• J2 = connector of exits, permits connect and manage leds or relays. The Pin 1 is the positive. The Pin 2 is GND, being common for all exits. We can manage until 38 exits. (see explanatory drawing).
• J3 = connector of inputs, can connect switches, pulsh bottons, etcetera, is formed for four groups of 9 pins, it gives us a total of 36 inputs. (see explanatory drawing).
• J4 = just as the J3, it is a connector for 36 inputs.

Inputs:

The inputs in the Master card are formed for groups of 9 + 1 GND common, that means that we dispose of 36 entrances for connector (J3 and J4), that is, for each Master card connected to the USBExpansion are 72 inputs, that multiplied for the four cards that we can connect to each USBExpansion, gives us a total of 288 digital inputs, numbered according to the following table:

Master card number	J3 inputs	J4 inputs
1	0-35	36-71
2	72-107	108-143
3	144-179	180-215
4	216-251	252-287

The virtual/logical entrances it is not corresponded with the physical number of pin, being the the Master numeric virtual distribution as follows in the following outline:

Connector J3:

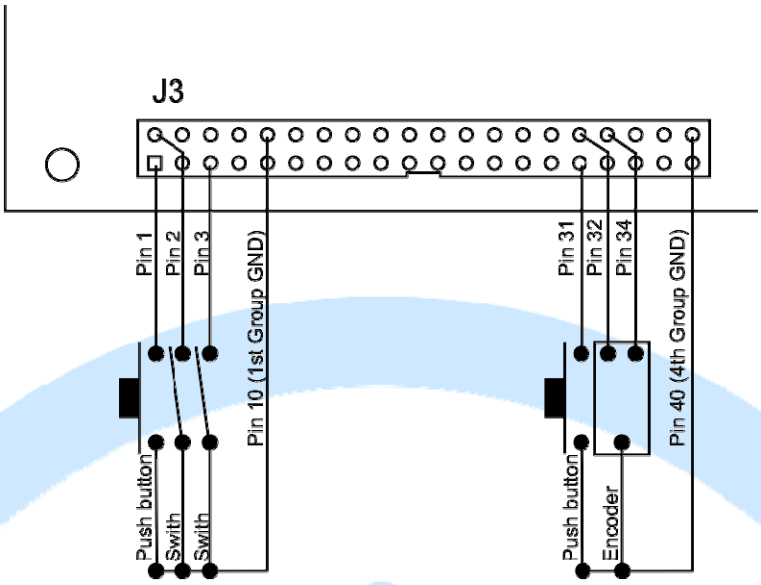
INPUTS	002	003	007	006	GND	011	012	016	015	GND	020	021	025	024	GND	029	030	034	033	GND
PINS	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40
PINS	1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31	33	35	37	39
INPUTS	001	004	008	000	005	010	013	017	009	014	019	022	026	018	023	028	031	035	027	032

Connector J4:

INPUTS	038	039	043	042	GND	047	048	052	051	GND	056	057	061	060	GND	065	066	070	069	GND
PINS	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40
PINS	1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31	33	35	37	39
INPUTS	037	040	044	036	041	046	049	053	045	050	055	058	062	054	059	064	067	071	063	068

As it sees in the diagrams, the entrances goes for groups of 9, at total the J3 and J4 connectors provides 72 inputs.
How we see the different groups of connections in everyone of them we have a common GND not interchangeable, that is to say, the entrances of a group can´t be connected with the GND of any other group, but that always must use the GND of his same group.

In the following image we can see an example of connection of different elements to the J3 connector. In it distinguish clearly the connected switches to the pins 2 and 3, also sees the connection of the push button in the pins 1 and 31, finally sees a connected encoder to the pins 32 and 34 (that correspond to the “logical inputs” 29 and 30), due to that the Gray ´s encoders type can go connected directly to the Master plate in two consecutive logical inputs.



As we see also, the common in all cases always belongs to the same group where it connects the elements.

Outputs:

Each Master card has until 45 exits distributed between the J2 and P2 connectors (38 and 7 respective-mentions), it who gives us a total, for the four Master that we can connect to an USBExpansion, of 180 outputs and are numbered according to the following table:

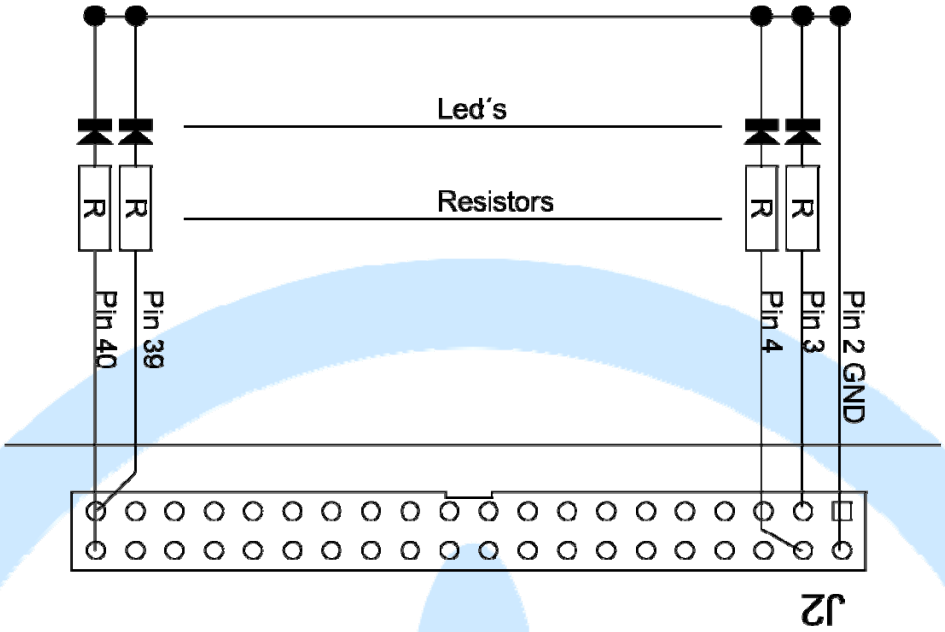
Master card number	J2 Outputs	P2 Outputs
1	11-48	49-55
2	75-112	113-119
3	139-176	177-183
4	203-240	241-247

Each exit provides 5V and about 25 mA, in them we will connect habitually LEDS, for the below amperage that provides, and could not connect elements with a high consumption.

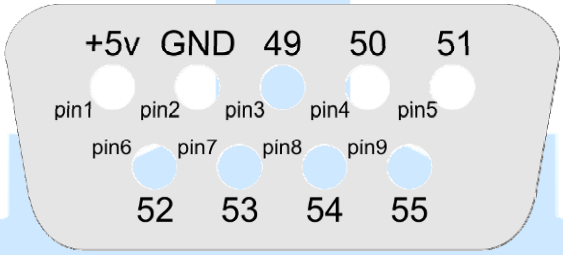
Connector J2:

OUTPUTS	GND	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40	42	44	46	48
PINS	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40
PINS	1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31	33	35	37	39
OUTPUTS	+5V	11	13	15	17	19	21	23	25	27	29	31	33	35	37	39	41	43	45	47

The connection of these LEDS is as we see in the following diagram:

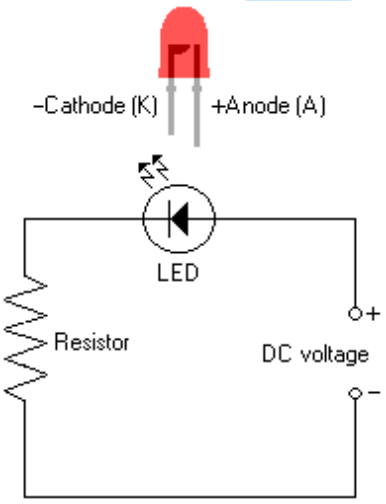


Connector P2:



Due to the own voltage of the LED, that we do is interpose a resistor between the positive and the own LED. For the calculation of this resistance we detail a formula:

R = (Vs – Vf)/ If
Where:
Vs = supplied voltage
Vf= Voltaje of LED (V)
If = Intensity of LED (A)



For example, we have a Master exit that supplies 5V, a LED that necessite 2,2V and with an intensity of 20mA (0.02A), if we put it into the formula, staies us just as:

r = (5 – 2.2)/0.02 = 140 Ohms.

Therefore, we must use a resistance of 150 Ohms, because is the superior more near standard to 140 Ohm.

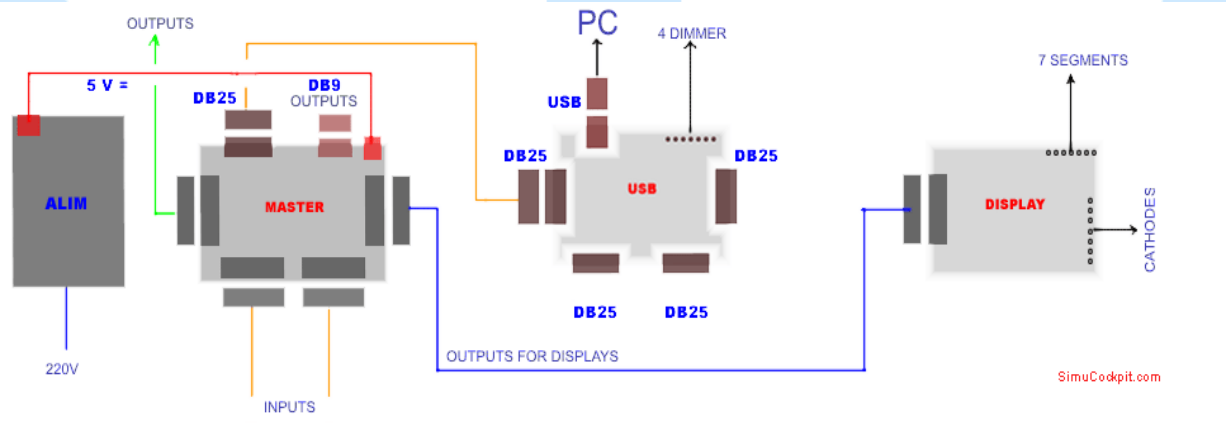
Also we must observe the position of montage of the LED, due to that has positive where (cathode) will solder the resistor connecting them to the GND of the connector that use the plate (J2 or P2) and negative (anode) we will connect to an exit of the IOCard Master. On the LED we will be able to easily distinguish it for two characteristic, in the case of the cathode is more short and also in the inside of the plastic it is seen that the foot is extended towards the side of the anode.

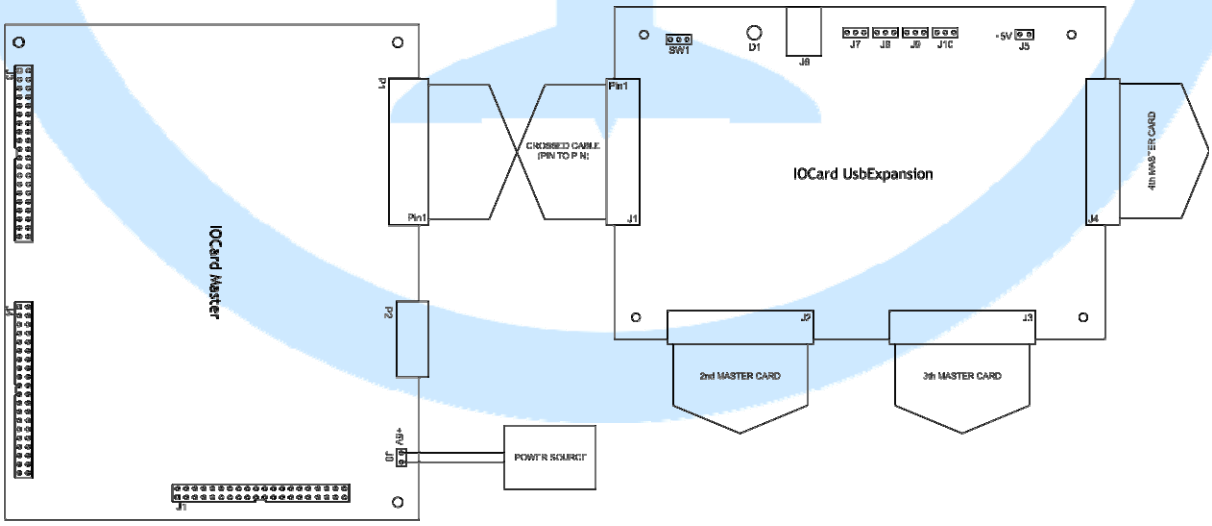
DISPLAYS:

Each Master card accepts in their bus of Displays until four cards of Displays II version.
The connector of this bus is the J1 connector and for the configuration of the different cards they use the own jumpers of the Displays card.
For a better configuration and use of this card of Displays, see the manual of the Displays II card, also available for your discharge on the Web of Opencockpits.

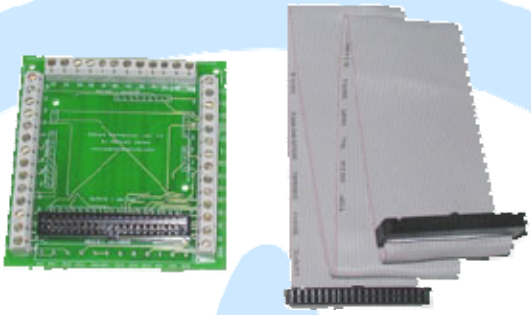
Starting of the cards:

Already we know the IOCards plates, base of our project of simulation, now we go to verifie the and connect the among them and to the computer:





We must connect the cards such as indicates in the drawing from above, by keeping in mind that is to must feed on 5V ONLY the Master card because the USBExpansion has a 5V OUTPUT for the Master or small accessories and will proceed to configure them to try their functioning.
For it the ideal one would be to connect elements to the exits and to the entrances, for a better verification of the different mechanisms. The connections of these elements had been descripted in this manual, in the corresponding sections (OUTPUTS, INPUTS, DISPLAYS).
There is also an option to connect the inputs and outputs, that facilitates us a lot of the labor of the connection, they are the 3T5 Conection Inputs and 3T4 Conection Outputs, references of our catalog.



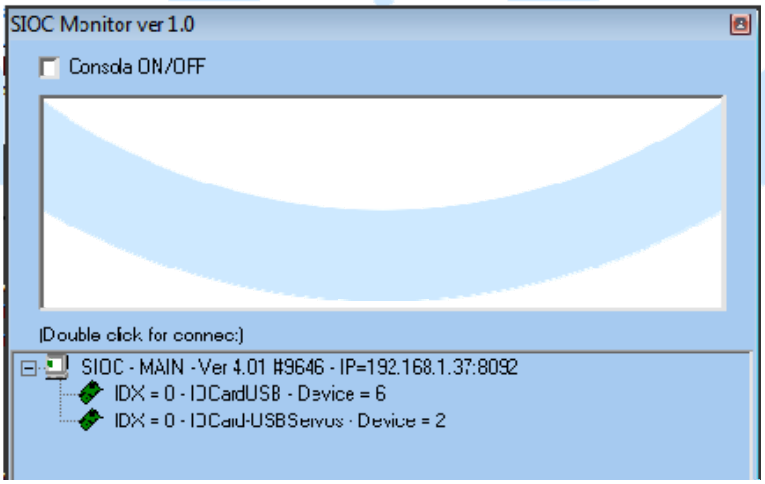
One time we have our different connected elements, will go in verify the cards in our PC.

Installation of software and configuration:

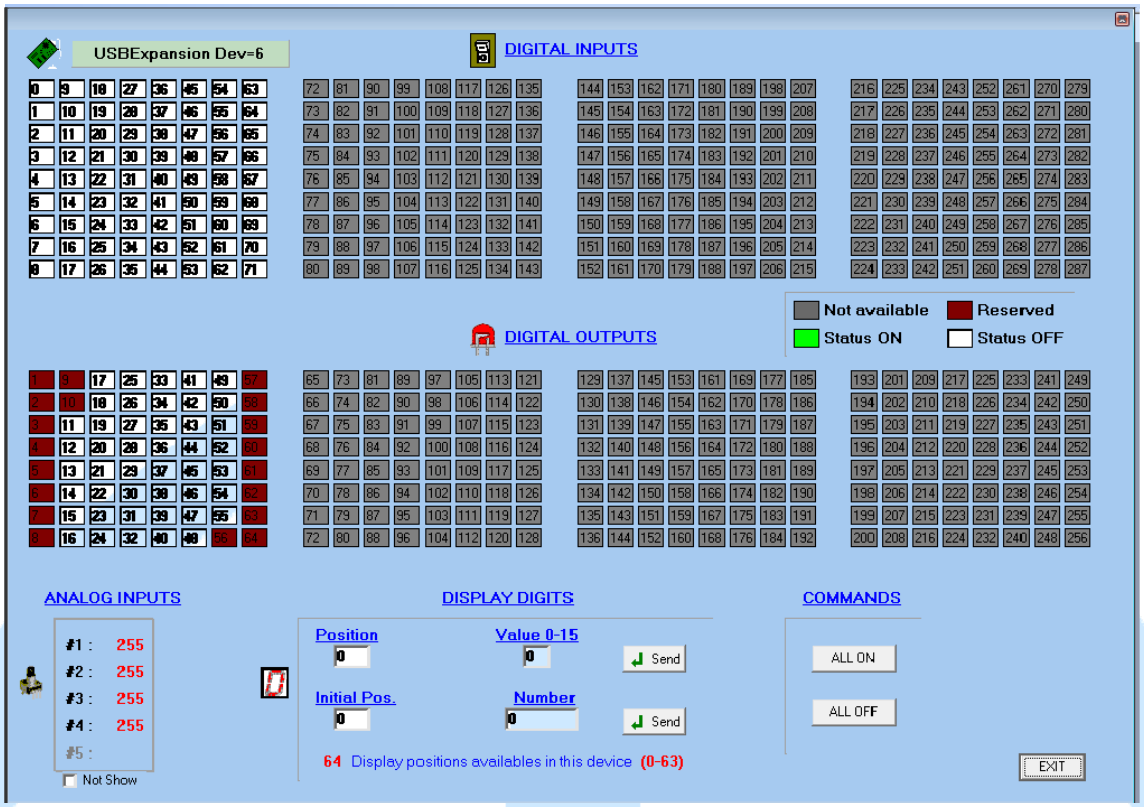
SIOC,
USBCHECK,
The flight simulator FS2004, FSX, Xplane, etcetera,
the necessary plugins for the simulator,
Etcetera.
At the end of this document there is a list of links to download the necessary software to put into practice this manual.

Proof of functioning of the cards:

1.- We start the SIOC (last version), push the SIOC Monitor button, a screen is showed in which appears all IOCards connected to our computer and we can connect to any of them:

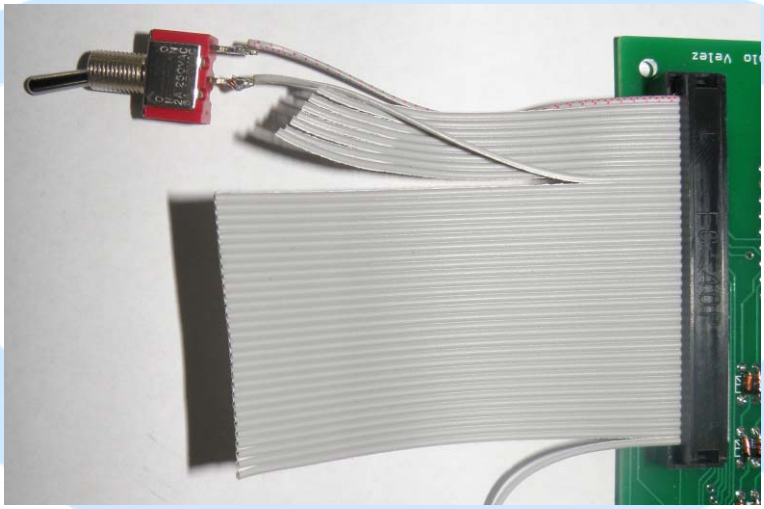


We do double click in the USBExpansion that we have connected the switch and the LED (in our IDX case = 0, IOCardUSB, Device = 6).

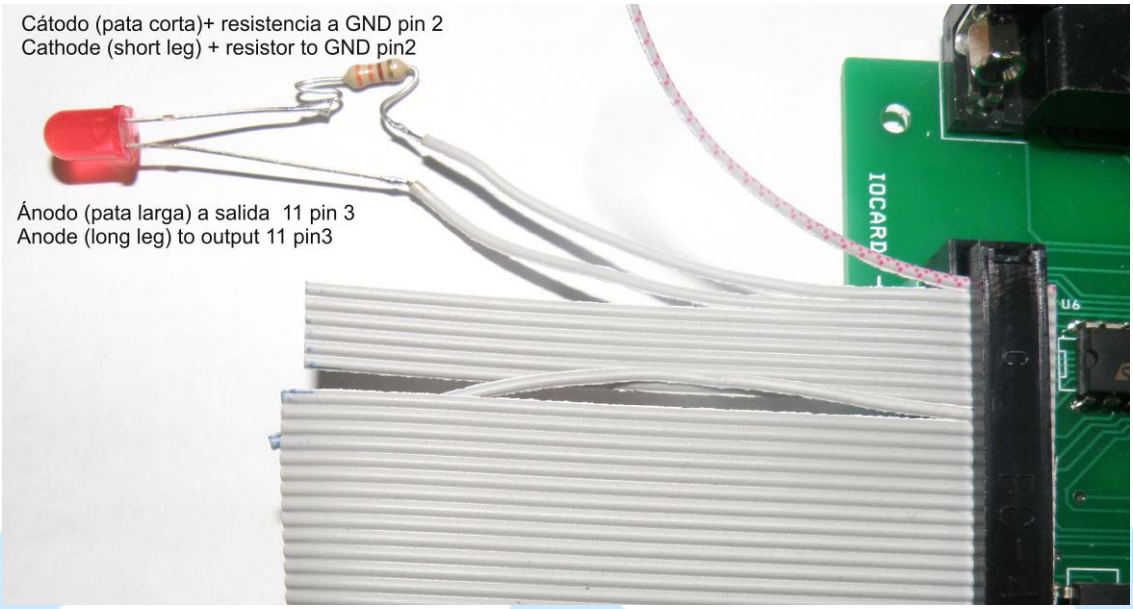


2.- Connecting an input and an output in the IOCard Master.

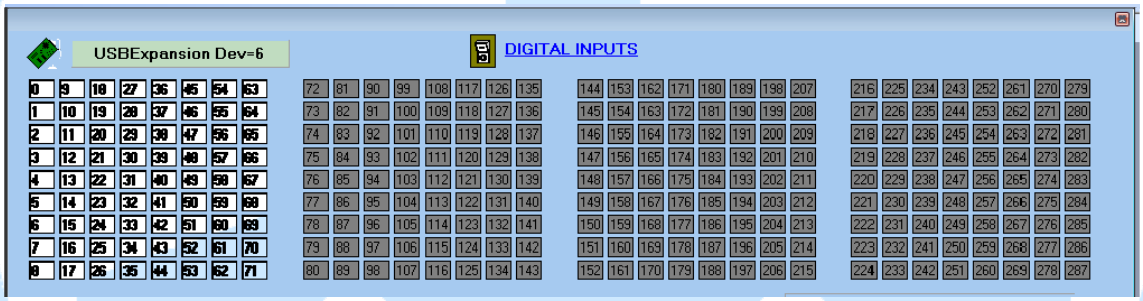
The entrance: in our case a connected switch to the logical entrance 1 that it is the physical entrance 1 (pin1) and to the common earth GND (pin 10) of the same group (the first of the J3 connector):



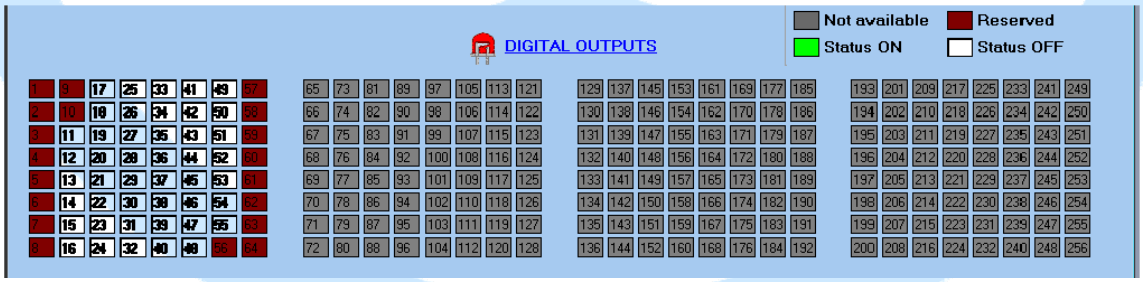
The exit: in our case a red LED connected to the logical exit 11 of the J2 (pin 3) and to the common earth (pin 2).



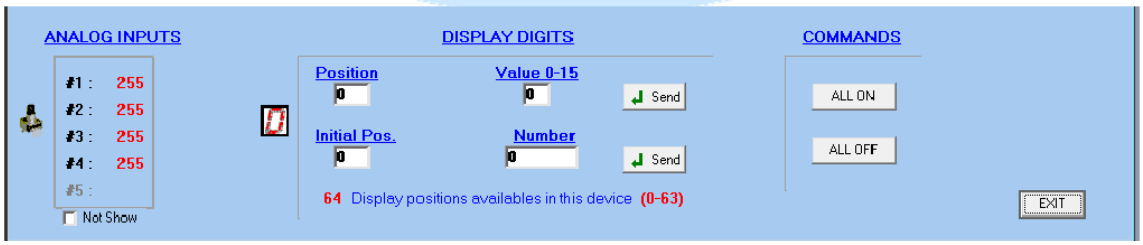
In the screen, we can see in the upper position the tables of inputs, those who be active (as many as Master cards connected) will show with white background and white buttons and those who not be connected in gray background buttons.



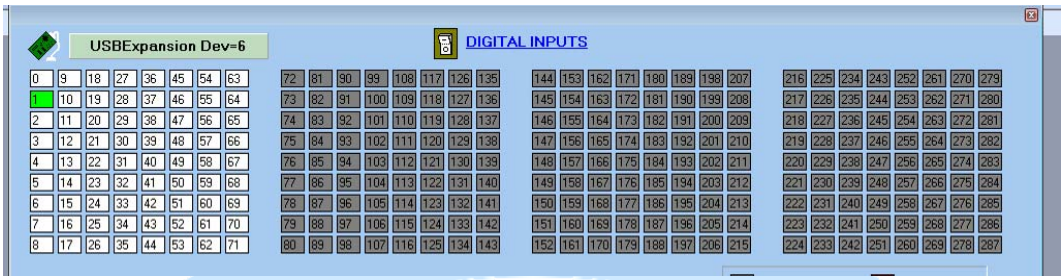
In the central part of the screen it shows us the tables of the exits of the Master active with white background and inactive in gray background.



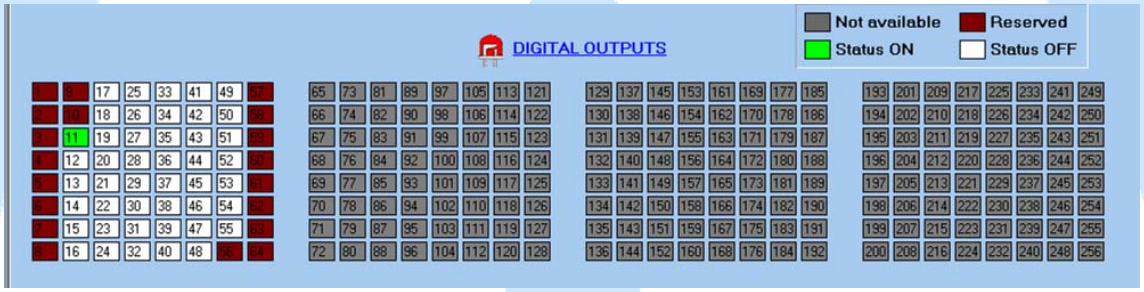
In the bottom of screen we see the other inputs and outputs (potentiometers and displays).



In our example, we need to verify if the entrance and the exit that we have connected runs, a switch and a LED, for the entrance simply activate the switch and in the table it see that the corresponding cell puts green (ON).

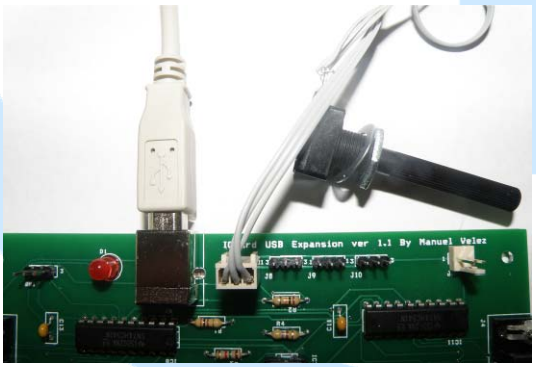


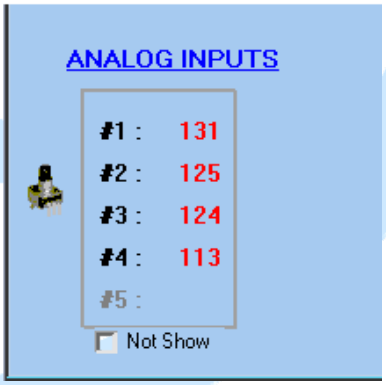
We see that the cell lights ON but our LED not, to see if it function we go to the cell that corresponds to the LED (in our example the output number 11) and do CLICK, see that whenever do click exchanges the state of the LED, if lights, all gone well, if not, it is necessary to revise the connections.



In the case of a 7 segments Display, will be able to see it in the manual of the IOCard Display II. card.

For the analogical inputs (potentiometers), the screen shows us the movement and consequently, the values that it take thoses entrances from their pertinent potentiometers:





Already we have learned to verify entrances and exits with the SIOC Monitor.

Pass now to use a programming language, SIOC to use our switch and our led, for major depth of use You can visit the Opencockpits website entering the forums and download the manual.

We will use a real element of a cabin that functions with a switch and an optical messenger: **Switch + Panel indicator (parking brake).**

To begin we will create a file of call text " prueba1.txt ":

```
// ********************************************************************************
// * Config_SIOC ver 4.01  - By Manolo Vélez - www.opencockpits.com
// ********************************************************************************
// * FileName : prueba1.txt
// * Date : 01/12/2011
Var 0400, Value 0 // Inicialización / Var init
Var 0001, Link FSUIPC_INOUT, Offset $0BC8, Length 2 // reading state of the parking brake
Var 0002, Link IOCARD_SW, Input 1 // reading state of the switch
{
IF V0002 = 1
{
V0001 = 32767
V0003 = 1
}
ELSE
{
V0001 = 0
V0003 = 0
}
}
Var 0003, Link IOCARD_OUT, Output 11
//  End of file prueba1.txt
```

The unique that we must change in the script is the marked in red, that represent the numbers that identify the entrances and exits in our card.

As can see the functioning it is very simple:

-First depose you the variable of FSUIPC $0BC8, that it is the variable that manages the parking brake, and is where will write the values that send you from our hardware.

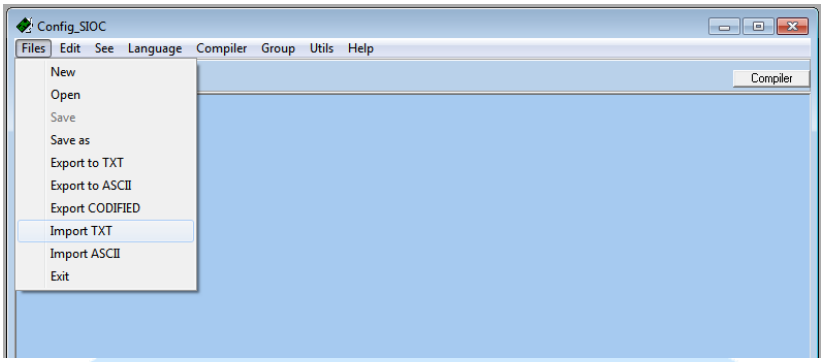
-In the following line we can see that the connected switch to the input “1” (in our case), to the is activated (value in 1) it sent to you the V0001 variable the value 32767 that it is the necessary value for activate the parking brake (for it has consulted previously the values in the FSUIPC offset list) and in turn sent you a 1 to the V0003 variable (that it is an exit for the light of parking brake), which does that lights the indicator panel in the plane.

-If the value of the switch out any other, then sent to the V0001 the value 0, that do that detaches the parking brake and the same value that to the V0003 variable and so extinguishes the indicator.

-The last line is the statement of the exit, that in our case it is connected to the number 11.

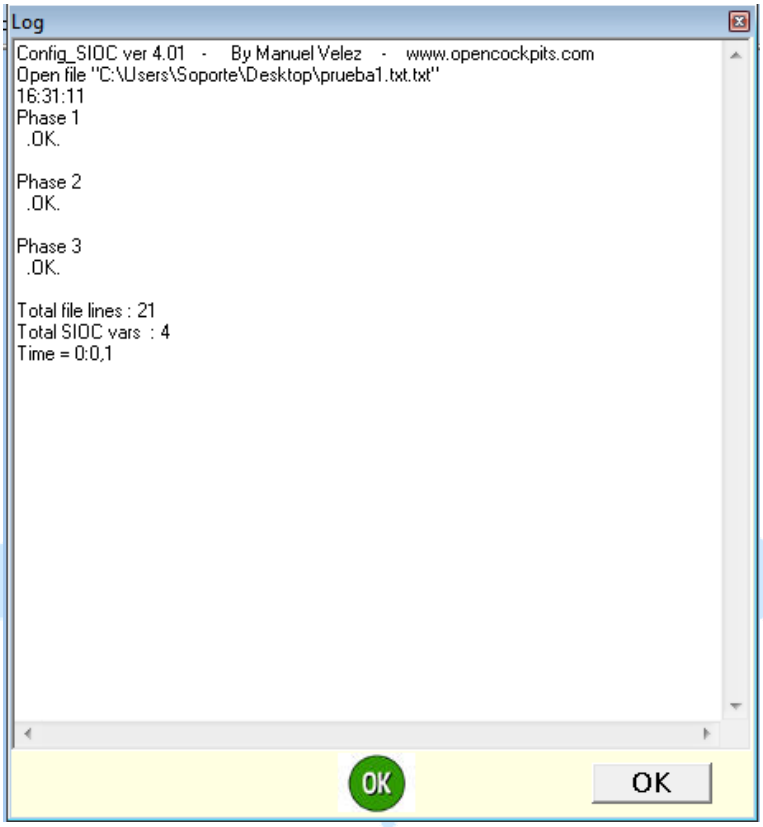
Now us must save the file in the directory of the SIOC, by being cautious that do not exist truncated lines:
Var 0401, Link FSUIPC_INOUT, offset $0BDC, longitude 4, value 0 // Mando of the |flaps|,
three positions
must appear as:
Var 0401, Link FSUIPC_INOUT, offset $0BDC, longitude 4, value 0 // Mando of the |flaps|, three positions
because the SIOC interprets the text three positions as an order of programming and it gives an error.

We keep it and we open the SIOC, we play the CONFIG button. Config_sioc it opens with a blank page of sioc.ssi. Are about to create other sioc.ssi as of our prueba1.txt file.



We select our file prueba1.txt and nothing more else make click we have two things:

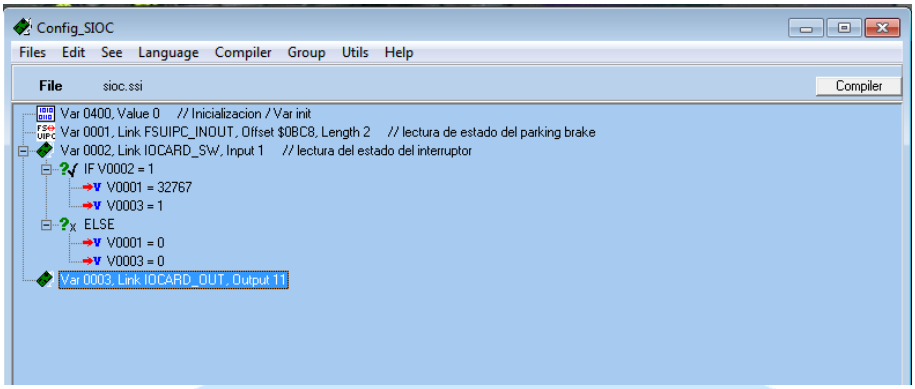
First a log of compilation



Already we see that the "compilation" of a text file in program, it made in 3 stages. Always is so. and when all this OK, can see down an green OK that tranquilizes us. That means that all it has gone out well. If, instead of the OK green, went out a red cross them be important, for example, that it have forget to copy a piece of the file prueba1.txt and can be that a key } has disappeared, or that has occured a leap of line. This is accustomed to occur often when we copy a piece of program of somebody.

Note 1: we have seeing that the transformation of a text file in program (the "compilation") it made automatically: the COMPILE´s button of Config_SIOC page has little utility.
Note 2: Creating a SIOC program as of a text it is chancy. The computer operators do it to go blind, but those who are not sure do to him bundles between the { and the }. To create a new.ssi or add things you to one existent, it is a lot of sure work directly in the page .ssi of Config_SIOC.

Then, we close the log of compilation clicking OK and now see our text in the window of the SIOC, but converted in a .ssi.



We have to remember always to save it, as will do whenever modify a file.ssi: Files/Save as and of name it is accustomed sioc.ssi, file that from now on no longer will appear empty.
If close and it open again sioc.exe already see that the SIOC is opened with the the last one fichero.ssi that have used.

Well, try sioc.ssi in the FS. For it we execute SIOC and we leave it resident mode (TRAY button), we start the FSX (with FSXUIPC installed) or another and fit an airplane with brakes, when we are in the cabin, the screen show us the message of pushing "." key to untie the brakes, if you change the switch that have programed will disappear the message and see that the LED changes state.

With this gives end to this manual, we invite you to read the manuals of the others Opencockpits elements and of the SIOC software and give you the thanks for trusting in we.

Links of interest:

Support area for clients:

http://www.opencockpits.com/catalog/info/