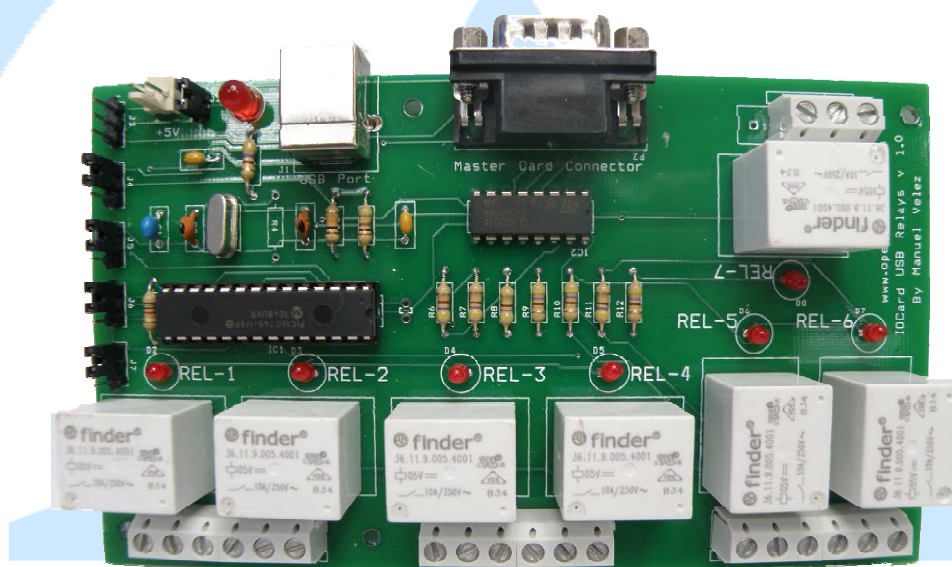




Opencockpits



Manual IOCard USB Relays

Índice:

MANUAL IOCARD USB RELAYS	1
ÍNDICE:.....	2
INTRODUCCIÓN:.....	3
USB RELAYS:.....	3
ESQUEMA Y COMPONENTES:.....	3
DESCRIPCIÓN DE LOS CONECTORES:	4
ESQUEMA DE CONEXIÓN:.....	4
PUESTA EN MARCHA DE LA TARJETA:	5
INSTALACIÓN Y CONFIGURACIÓN:	7
DECLARACIÓN DE RELÉS:	8
DECLARACIÓN DE ENTRADAS ANALÓGICAS:	8
EJEMPLOS DE SCRIPTS:.....	8
<i>Relés:</i>	8
<i>Entradas analógicas:</i>	9
LINKS DE INTERÉS:.....	9

Introducción:

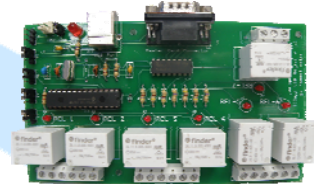
La tarjeta IOCard USB Relays ha sido diseñada para poder manejar hasta 7 relés de potencia y además posee 5 entradas analógicas.

La conexión de la tarjeta al ordenador es mediante el puerto USB.

Usb Relays:

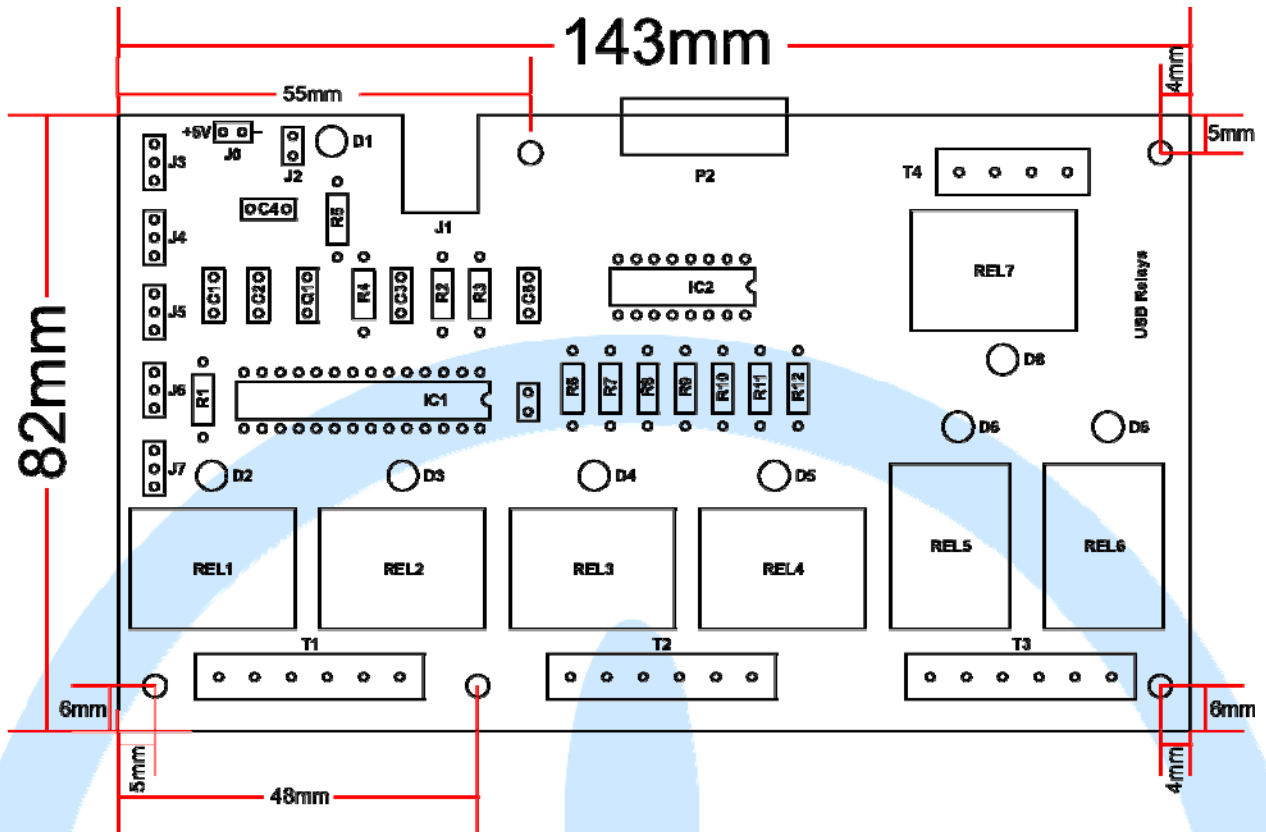
La tarjeta se auto instala como dispositivo HID y se controla mediante el protocolo IOCP. Como opción la tarjeta se puede conectar directamente a la tarjeta Master y funcionar mediante las salidas de dicha tarjeta.

Existe asimismo una variante con la opción de conectarse a través del puerto P2 con el USB desactivado, siendo solamente posible conectarla a través de salidas de la tarjeta Master.



Esquema y componentes:

- C1 = CONDENSADOR 220nf
- C2, C3 = CONDENSADORES 22pF
- C4, C5 = CONDENSADORES 0.1mF
- D1...D8 = DIODOS LED
- IC1 = MICROCONTROLADOR 16C745
- IC2 = CIRCUITO INTEGRADO ULN2003A
- J0 = CONECTOR ALIMENTACION +5V
- J1 = CONECTOR USB
- J2 = JUMPER SELECTOR ALIMENTACION EXTERNA/INTERNA
- J3, J4, J5, J6, J7 = CONECTORES DE 3 PINES
- P2 = CONECTOR DB9
- Q1 = CRISTAL CUARZO 6Mhz
- REL1...REL7 = RELÉS OMRON G5LE-1
- R1 = RESISTENCIA 1K5
- R2 = RESISTENCIA 10K
- R3 = RESISTENCIA 100R
- R4 = NO USADO (FUTURAS OPCIONES)
- R5 ... R12 = RESISTENCIAS 470R
- T1, T2, T3, T4 = CONECTORES DE TORNILLO



Descripción de los conectores:

- J0 = Conector de alimentación +5V (la placa no está protegida contra inversión de polaridad).
- J1 = Conector USB al ordenador. El ordenador reconocerá la tarjeta automáticamente como un dispositivo HID.
- J2 = Jumper para la selección de la fuente de alimentación: si el jumper está cerrado la alimentación provendrá internamente (USB), si está abierto la alimentación será externa (fuente de alimentación).
- J3 a J7 = Conectores para las entradas analógicas.
- P2 = conector tipo DB9, para conectar la tarjeta directamente a la tarjeta Master, sin usar el puerto USB (nunca deberán estar conectadas a los dos puertos a la vez).

Esquema de conexión:

El conexionado es simple, cada relé tiene 3 salidas correlativas en los conectores de tornillo T1 a T4:

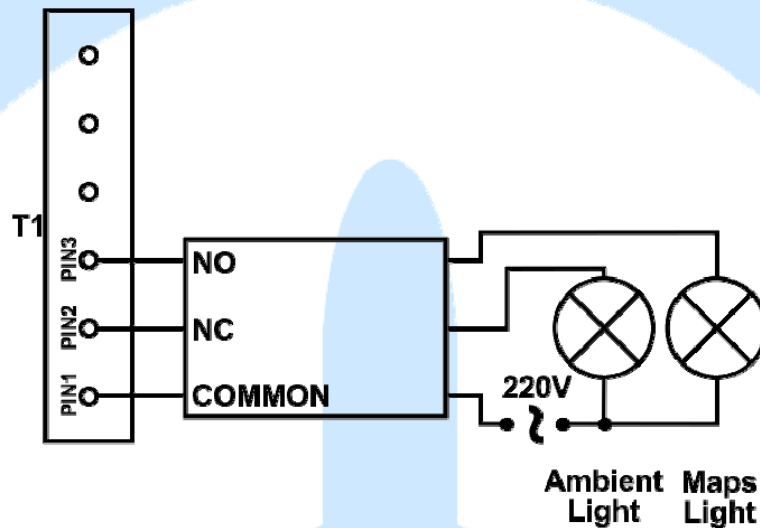
PIN	FUNCIÓN
1	COMMON
2	NC (normalmente cerrado)
3	NO (normalmente abierto)

El relé en reposo tiene siempre cerrado el circuito del pin 2 y abierto el del pin 3, cuando el relé es activado, cambia la situación siendo el circuito del pin 2 el que queda abierto y el del pin 3 cerrado.

Ejemplo de conexión:

Trabajemos con dos lámparas de 220V, queremos que al activar el foco de mapas automáticamente se desactive la lámpara de luz ambiente de la cabina, que está SIEMPRE encendida, para ello conectamos un polo de la lámpara de luz ambiente al pin COM y el otro al pin NC (normalmente cerrado) y por otro lado, un polo de la lámpara de mapas al pin COM y el otro al pin NO (normalmente abierto), sin olvidar de alimentar las lámparas desde la red eléctrica (ver dibujo de conexión) así, al conectar la tarjeta, se encenderá la lámpara de luz ambiente y la de mapas permanecerá apagada, pero al activar el relé, se apagará la lámpara de luz ambiente y se encenderá la de mapas y al desactivarlo volverá a encenderse la luz ambiente y se apagará la luz de mapas.

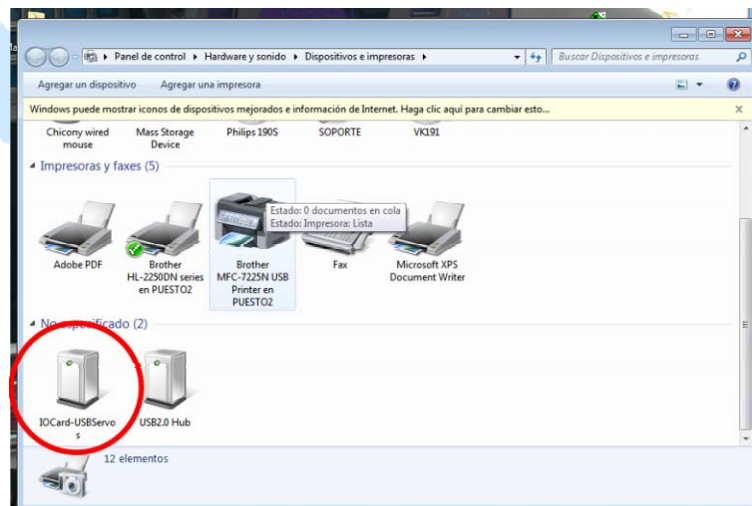
Para una mejor explicación del conexionado, a continuación se adjunta una imagen:



Puesta en marcha de la tarjeta:

Ya conocemos la teoría de conexión de la IOCard USB Relays, ahora pasemos a comprobarla y conectarla para ver los resultados.

En nuestro ejercicio conectaremos la tarjeta USB Relays a un puerto USB y veremos que nuestro ordenador la reconoce.



Ha sido reconocida, así que practicaremos con el ejemplo anterior, pero en vez de trabajar con lámparas de 220V lo haremos con leds de 5V por comodidad y seguridad.

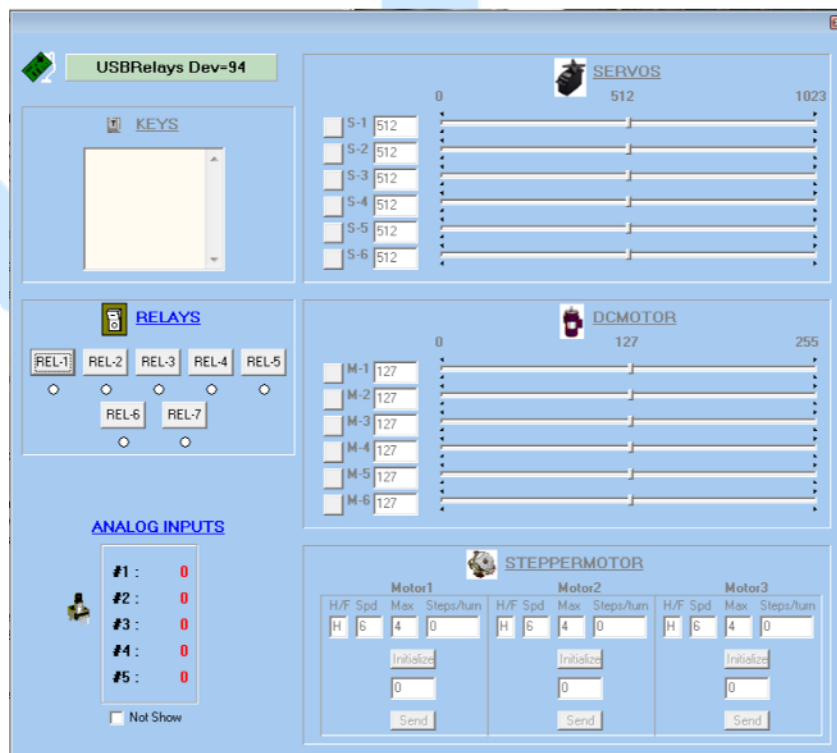
En esta prueba cerraremos el puente J2 para que la placa se alimente del USB porque vamos a poner poca carga (sólo un relé), en caso de conectar todos los relés es aconsejable abrir el puente J2 y alimentar la USB Relays con 5V y 1A externos.

También conectaremos un potenciómetro para comprobar el funcionamiento de las entradas analógicas, en total: 1 potenciómetro en J3, 1 led blanco en el pin 3 (luz mapa), 1 led rojo en el pin 2 (luz ambiente) y en el pin 1 conectaremos el común de las luces de cabina y su alimentación externa, además para evitar que las lecturas de las entradas analógicas no usadas fluctúen hemos cerrado el circuito de las mismas con unos puentes:



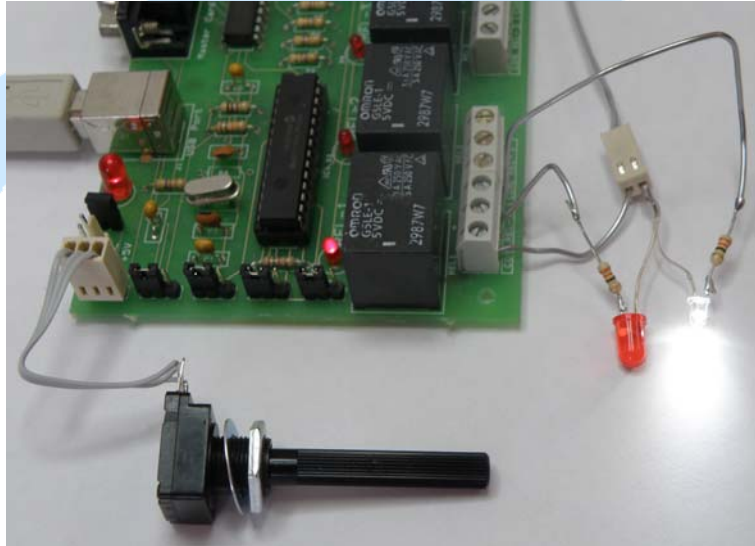
Advertencia: en cuanto conectemos la alimentación externa el led rojo permanecerá encendido.

Ahora probaremos con SIOC Monitor si nuestro montaje ha salido bien y todo funciona. Arrancamos SIOC, miramos si nos la ha reconocido y si es así pulsamos el botón Monitor, en IOCard USB Relays y chequeamos la instalación:



Como podemos observar la entrada analógica #1 muestra el valor de nuestro potenciómetro y las demás muestran el valor 0, si giramos el potenciómetro veremos cómo cambia el valor representado de más a menos o al revés (depende de cómo hagamos las conexiones de los pines), si no nos interesa el sentido de subida o bajada de los valores podemos intercambiar los pines exteriores del conector J3 o hacerlo por software mediante un script SIOC.

Probemos ahora el relé 1, si pulsamos el botón REL-1 el relé se activará y se encenderá el led blanco, se apagará el rojo y podremos observar que el led de control del relé se activará tanto en el monitor como en la placa. Si le damos de nuevo al REL-1 el relé se desactivará y el led rojo volverá a lucir y el blanco se apagará.



Deberíamos hacer estas pruebas para todos los conectores.

Instalación y configuración:

Para poder hacer uso de las características de la USB Relays debemos tener instalado SIOC (última versión a ser posible), el simulador de vuelo FS, FSX, Xplane, etc.

Al final de este documento hay un listado de links para poder descargar el software necesario para poder poner en práctica este manual.

Para gestionar esta tarjeta se usan scripts, para lograr que SIOC controle la IOCard USB Relays deberemos editar la entrada en el fichero sioc.ini, de tal manera que asignemos un índice de dispositivo a cada tarjeta que instalemos, creando una entrada en el fichero sioc.ini por cada tarjeta USB Relays conectada. Es importante recordar que la USB Relays tiene dos funciones, una la de control de relés y la otra de entradas analógicas que también deben ser declaradas en sioc.ini, será en el siguiente formato:

USBRelays=XX,YY

XX indica el número de índice (IDX) dentro de nuestro sistema de tarjetas.

YY indica el número de dispositivo del puerto USB al que está conectado (DEVICE).

USBANalogic=XX,YY

XX indica el número de índice (IDX) dentro de nuestro sistema de tarjetas.

YY indica el número de dispositivo del puerto USB al que está conectado (DEVICE).

Por ejemplo, si conectamos dos tarjetas USB Relays con los números de dispositivos 94 y 42 (estos números se pueden averiguar fácilmente con el propio programa SIOC) entonces las declararíamos en el sioc.ini de la siguiente manera:

USBRelays=3,94 (como en nuestro ordenador de pruebas, 3: es el IDX y 94 es el device).
 USBRelays=4,42 (4 es el IDX y 42 es el device).
 USBAnalogic=3,94 (como en nuestro ordenador de pruebas, 3: es el IDX y 94 es el device).
 USBAnalogic=4,42 (4 es el IDX y 42 es el device).

Declaración de relés:

Para hacer referencia al número de relé exacto, debemos tener en cuenta el número de índice que le hemos asignado a cada una de las tarjetas USB Relays. Debemos definir la salida:

Var VVVV, name NNNN, Link USB_RELAYS, device DD, Output S

VVVV = número variable.

NNNN = nombre variable (opcional).

DD = numero de índice definido en sioc.ini (si la tenemos declarada como 0: no hace falta poner la referencia al número de Device).

S = numero de relé 1-7.

Ejemplo de definición:

Var 0001, name rele_maps, Link USB_RELAYS, Device 3, Output 1

Declaración de entradas analógicas:

Para la lectura de las entradas analógicas se deberá usar el formato siguiente:

Var VVVV, name NNNN, Link USB_ANALOGIC, Device DD, Input EE, posL LLL, posC CCC, posR RRR

VVVV = número variable.

NNNN = nombre variable (opcional).

DD = numero de índice definido en sioc.ini (si la tenemos declarada como 0: no hace falta poner la referencia al número de Device).

EE = numero de entrada analógica 1-5.

LLL = posición máxima del dispositivo a la izquierda.

CCC = posición central del dispositivo.

RRR = posición máxima del dispositivo a la derecha.

Ejemplo de definición de entrada analógica:

Var 0002, name pot_flaps, Link USB_ANALOGIC, Device 3, Input 2, posL 1, posC 128, posR 255

Ejemplos de scripts:

Relés:

El ejemplo que hemos usado anteriormente nos servirá para probar el funcionamiento de la USB Relays junto con una Master y un interruptor, nos basaremos en el montaje de prueba del manual de la USB Expansion + Master. En este caso no nos hará falta el simulador.

Para empezar crearemos un archivo de texto llamado "prueba_rele1.txt":


```
// *****
// * Config_SIOC ver 4.01 - By Manolo Vélez - www.opencockpits.com
// *****
// * FileName : prueba_rele1.txt
// * Date : 13/03/2012

Var 0001, name rele, Link USB_RELAYS, Device 3, Output 1
Var 0002, name map_switch, Link IOCARD_SW, Input 1
{
  If &map_switch = 1
  {
    &rele = 1
  }
  ELSE
  {
    &rele = 0
  }
}
// End of file
```

Escribiremos el script lo guardaremos y lo ejecutaremos con SIOC, veremos que el led rojo (luz ambiente) permanecerá encendido hasta que activemos el interruptor para encender el led blanco (luz de lectura de mapas).

Entradas analógicas:

Ahora usaremos el potenciómetro conectado para mover un servo conectado a una USB Servos declarada como IDX = 0 y device = 2.

Para empezar crearemos un archivo de texto llamado "prueba_rele1_servo1.txt":

```
// *****
// * Config_SIOC ver 4.01 - By Manolo Vélez - www.opencockpits.com
// *****
// * FileName : prueba_rele1_servo1.txt
// * Date : 13/03/2012

Var 0001, name servo, Link USB_SERVOS, Device 0, Output 1, PosL 190, PosC 512, PosR 1023
Var 0002, name trim, Link USB_ANALOGIC, Device 3, Input 1, posL 1, posC 128, posR 255
{
  &servo = &trim
}
// End of file
```

Escribiremos el script lo guardaremos y lo ejecutaremos con SIOC, no nos hace falta arrancar el simulador, si giramos el potenciómetro el servo se moverá en el rango definido por el potenciómetro.

Con esto damos fin a este manual, os invitamos a leer los manuales de los demás elementos de Opencockpits y del software SIOC y os damos las gracias por confiar en nosotros.

Links de interés:

Zona de soporte para clientes:

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