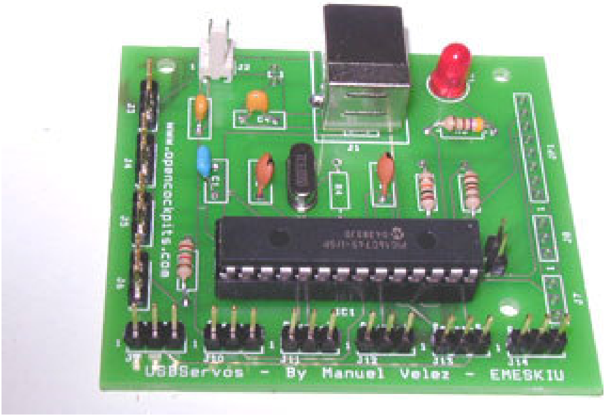


IOCard USBServos

User Manual

Author: Manuel Vélez
Translation by: Manuel Hernández-Peña

www.opencockpits.com

ver 1.0

INTRODUCTION

USBServos card has been designed to control up to 6 servo motors (R/C type) and 4 analogues inputs.

The card connects to an USB port and its controller is managed by IOCP protocol.

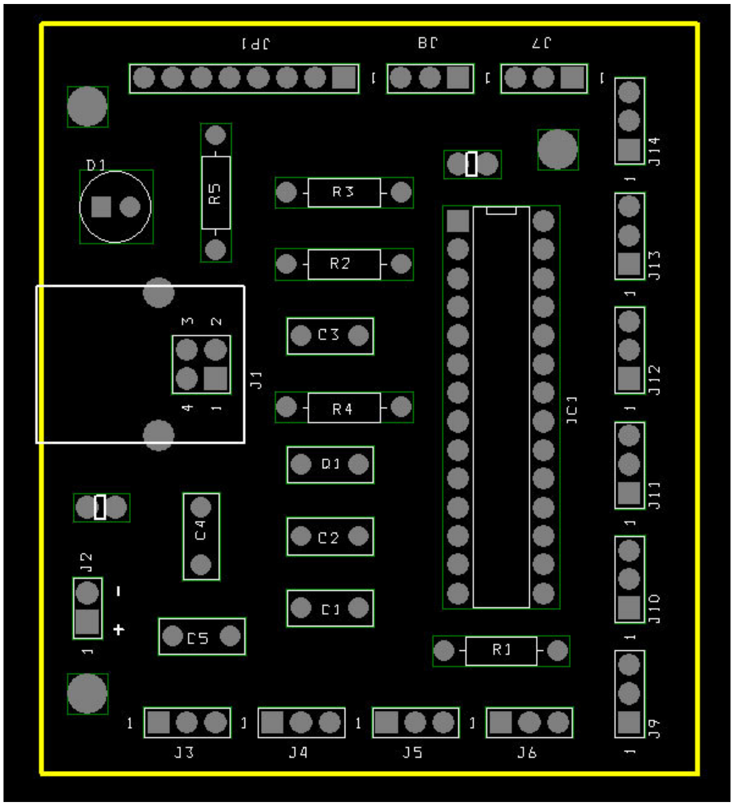
TECHNICAL SPECIFICATIONS

- USB connection
- Able to control up to 6 servo motors.
- Able to connect up to 4 A/D converters (8 bits)
- Software to connect to IOCP.
- Servo motors are controlled with a 10 bits resolution; possibility of disconnecting any motor.

PART LIST

C1	= Condenser 220nf
C2,C3	= Condensers 22Pf
C4,C5	= Condensers 0,1uF
D1	= Led
IC1	= Microcontroller 16C745
J1	= USB connector
J2	= External power source connector (2 pins)
J3 a J14	= 3 pins connectors
Q1	= clock 6MHZ
R1	= Resistor 1K5
R2	= Resistor 10K
R3	= Resistor 100R
R5	= Resistor 470R
SW1	= Reset (2 pins)
Other components	= *not connected, for future options*

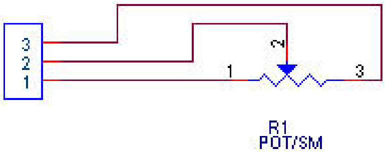
CONNECTIONS



J1 = USB connector.
J2 = Alimentación.
J3 – J6 = Analogue inputs.
J9 – J14 = Servo motors connectors.
J7, J8 y JP1 = Not used.

J3 a J6 – Analogue inputs

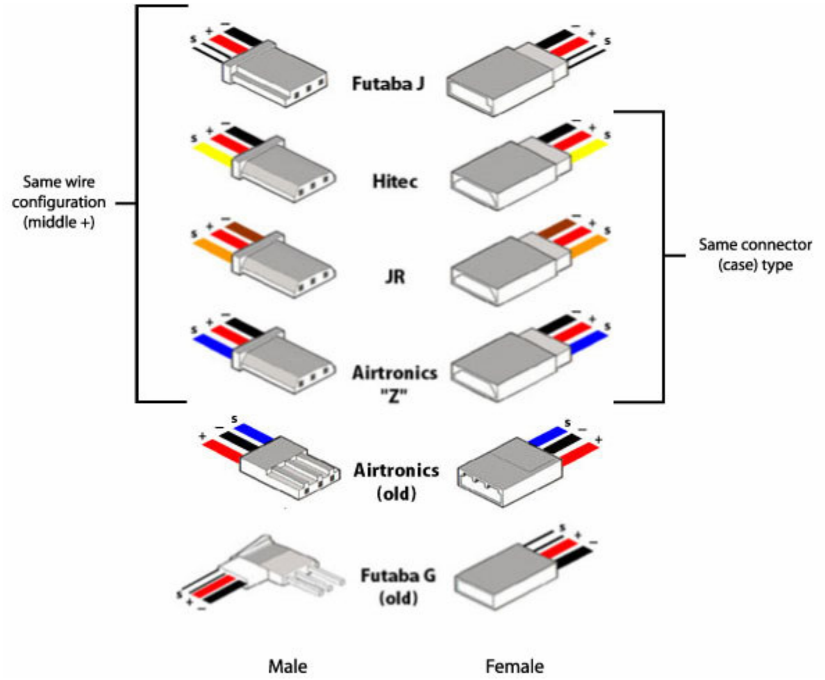
Pin 1 = GND
Pin 2 = Data
Pin 3 = +5V



POTENCIOMETRO 10K

J9 a J14 – Servo motors

Pin 1 = +5V
Pin 2 = Data (S)
Pin 3 = GND

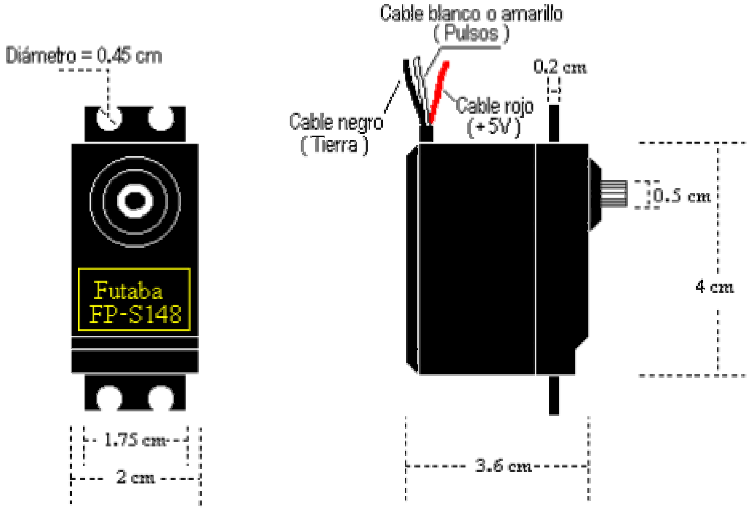


J2 – External power source +5V

Pin 1 = +5V
Pin 2 = GND

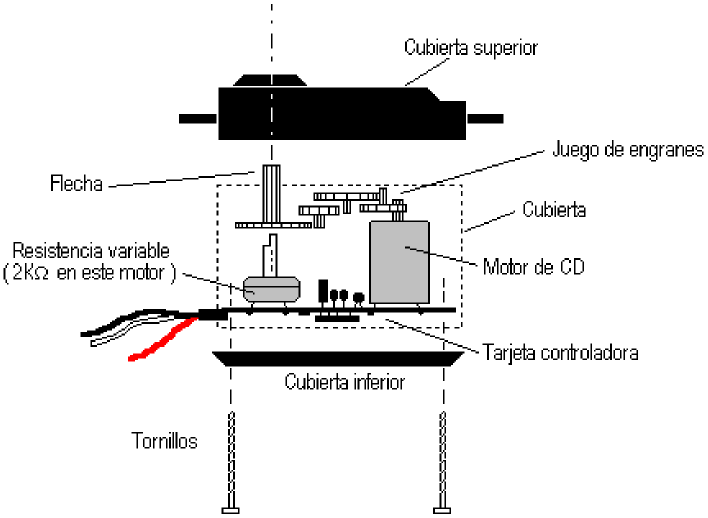
R/C SERVOMOTOR DIAGRAM

This is the diagram for a typical R/C servo motor:



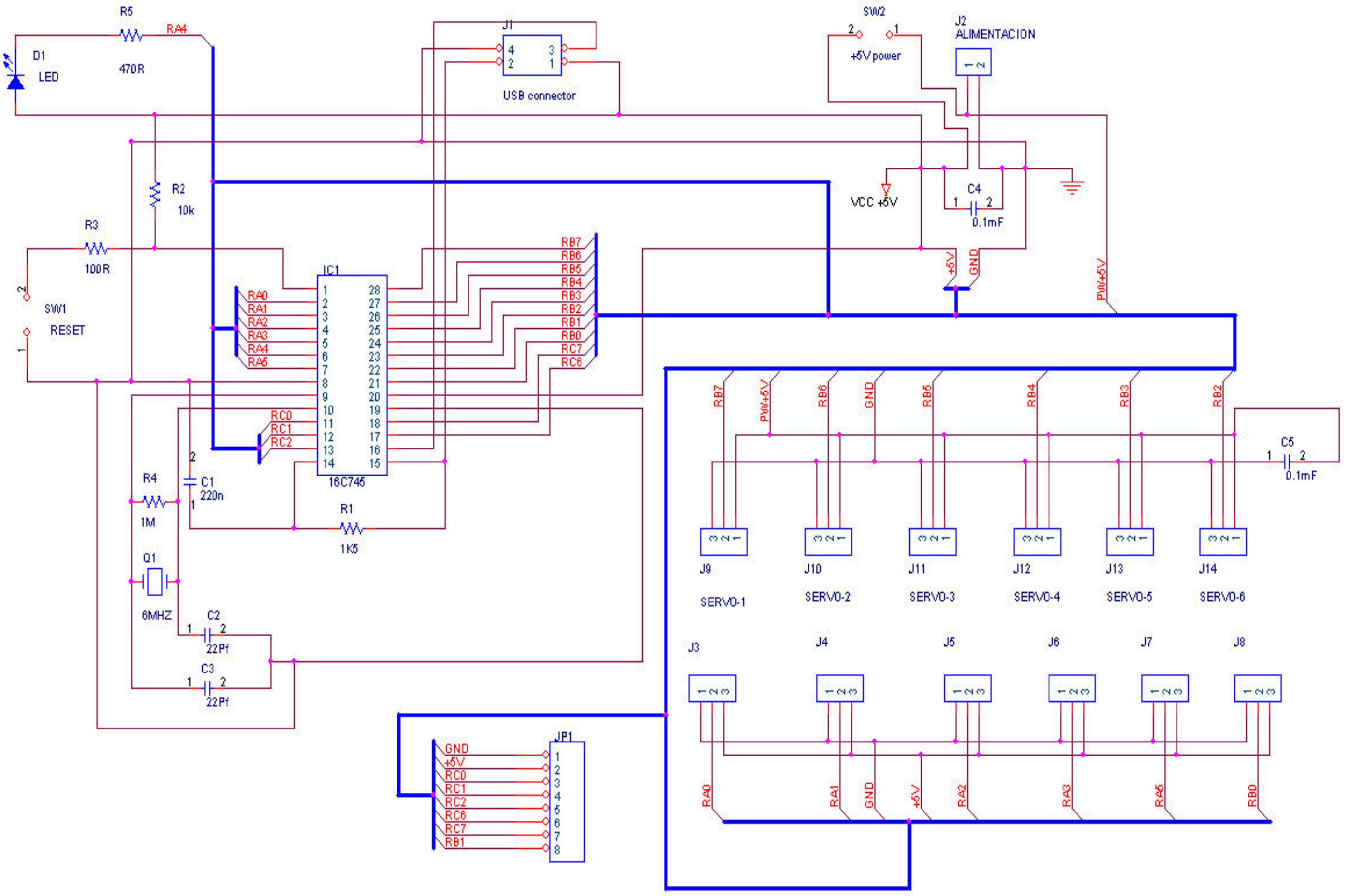
A servo motor is an electric motor that only turns 180 degrees. There are 3 cables. The res one is the +5V feed; the black one is GND; and the third one (white or yellow) is the control cable that set the position the motor has to reach.

Inside the servo motor body, there is a controller card that reads the information given by the third cable and tranforms it into orders to the electric motor.



The variable resistor is attached to the axle ("flecha" in the picture) so the controller knows the axle position. With this information and the one from the third cable, the controller is able to decide in which direction the motor has to be turned.

CARD SCHEME


R5
RA4
470R
D1
LED
J1
4
2
3
1
USB connector
SW2
2
1
+5V power
J2
ALIMENTACION
1
2
R2
10k
R3
100R
VCC +5V
C4
1
2
0.1mF
SW1
RESET
2
1
IC1
RA0
RA1
RA2
RA3
RA4
RA5
RC0
RC1
RC2
1
2
3
4
5
6
7
8
9
10
11
12
13
14
28
27
26
25
24
23
22
21
20
19
18
17
16
15
16C745
RB7
RB6
RB5
RB4
RB3
RB2
RB1
RB0
RC7
RC6
+5V
GND
PW+5V
C1
220n
2
1
R4
1M
R1
1K5
Q1
6MHZ
C2
1
2
22Pf
C3
1
2
22Pf
RB7
PW+5V
RB6
GND
RB5
RB4
RB3
RB2
C5
1
2
0.1mF
3 2 1
J9
J10
J11
J12
J13
J14
SERVO-1
SERVO-2
SERVO-3
SERVO-4
SERVO-5
SERVO-6
J3
J4
J5
J6
J7
J8
1 2 3
RA0
RA1
GND
+5V
RA2
RA3
RA5
RB0
JP1
GND
+5V
RC0
RC1
RC2
RC6
RC7
RB1
1
2
3
4
5
6
7
8

USBSERVOS SOFTWARE

IOCServos.exe: connection to IOCP

The card is configures with the file IOCServos.ini

In this file we can set the following parameters:

MUSB=No, we will set YES in case we have more than one card connected to the same PC. We'll have to set the device number too.

deviceUSB=2048, will inform the controller about the device number. If we don't know the device number, we can connect the card writing down the number shown by the controller. This number is different for each USB port.

IOCP_host=localhost, will set the IOCP server address where to connect.

IOCP_port=8092, will set the IOCP server port where to connect.

The IOCP server (normally SIOC server) can be located in any PC in our network. So we can install our cards at any PC.

USB_AD=4, number of A/D converters that we will use.

IOCP_timeout=4000, maximum timeout for IOCP packages.

IOCP_Var0=0, IOCP variable that controls #1

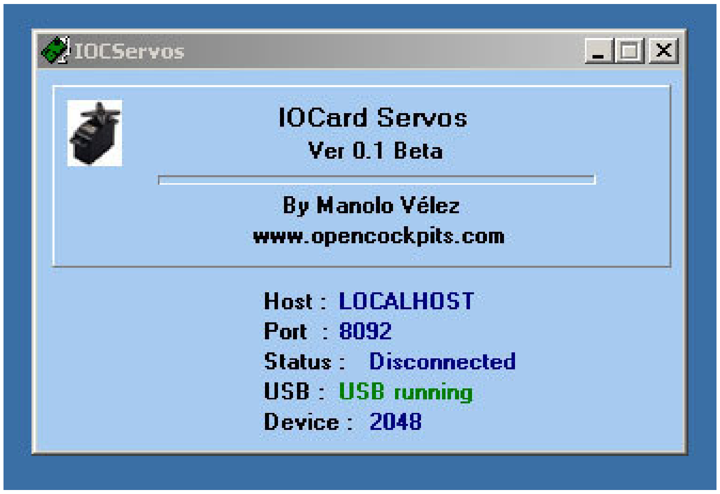
…

IOCP_Var5=5, IOCP variable that controls #5

IOCP_Var6=6, IOCP variable where to read the A/D #1

…

IOCP_Var9=9, IOCP variable where to read the A/D #4



IOCard Servos
Ver 0.1 Beta

By Manolo Vélez
www.opencockpits.com

Host : LOCALHOST
Port : 8092
Status : Disconnected
USB : USB running
Device : 2048

When the program starts, it will inform the connection address (if it has connected with the IOCP server and if there are an USB card connected and working) and the number of device of the card.

To control the servo motors, we just have to write a small SIOC program in which we'll set the servo variable with the following values:

0 = Motor disabled.
1-1023 = servo minimum and maximum positions.

Example of SIOC configuration: Vertical speed indicator.

```
Var 0010, Link FSUIPC_IN, Offset $02C8, Length 4, Type 1     // Vertical Speed
{
  L0 = V0010 * 0.7895     // FSUIPC conversion
  L1 = L0 * 0.085166     // 12000 FPM = 1022 positions
  L2 = 511 - L1     // Center
  IF L2 > 1022     // Upper limit
  {
    L2 = 1022
  }
  IF L2 < 1     // Lower Limit
  {
    L2 = 1
  }
  V0000 = L2
}

Var 0000     // Servo Motor
```