MILFORD INSTRUMENTS Ltd

Rev1.2 18/4/02

SERIAL SERVO-8 DRIVER BOARD (1-904)

The Serial Servo Driver Board allows the control of up to 8 Hobby type servos from an RS232 input. The SSDB is supplied fully assembled and tested.
The SSDB has a number of configuration jumpers;-
-the SSDB address may be preset to one of two addresses allowing two such units to be piggy-backed to control up to 16 servos from the same RS232 input,
-the operating range of the servos can be selected from 90 deg maximum to 180 deg maximum depending on the type of servo used,
-the SSDB may be driven from an external RS232 source or from an onboard BS2-IC,
-the baud rate of the SSDB may be selected to either 2400 or 9600 baud.
-circuit board size of 50x107mm

Windows 95/98 freeware is available from the admin section of our web site- www.milinst.com

THEORY OF OPERATION
The Serial Servo Driver Board receives RS232 signals and generates 8 continuous pulse streams suitable for driving and holding in position, 8 hobby type servos.
RS232 signals may be at either 2400 (jumper out) or 9600 baud (jumper in) and must conform to the following 3 byte protocol:

Byte 1	255	(Synchronization byte)
Byte 2	Servo # (0-7)	(Address jumper out)
	Servo # (8-15)	(Address jumper in)
Byte 3	Servo Position (0-254)	

eg the following string will send servo 6 to the mid point of its travel:
<255> <6> <127>

The default movement range setting for the servos is 90 degrees thereby giving a resolution of 90/254= 0.36 degrees/unit signal change.
With the range setting jumper in place the range is extended to 180 degrees thereby giving a resolution of 180/254= 0.72 degrees. Not all servos can accommodate a full 180 degree range- before using this facility please refer to the section on SERVOS.
Note that the configuration jumpers are only read at power-up- changes made after power-up will not be effected until the unit is re-powered

SIGNAL SOURCE

At power up, the red LED will light. When a suitable serial signal is received, the LED will extinguish and thereafter only light momentarily when a valid synchronisation byte is received.

With the D-9/BS2 jumper switched to the D-9 position, the SSDB may be driven directly from a PC using a standard 9 way serial cable.
With the jumper in the BS2 location, the input to the controller is transferred from the D-9 socket to pin10 of the BS2-IC socket. The Controller is then under the full control of any programme that you may wish to run from the BS2-IC. With the signal source jumper set to the BS2 position, the D-9 socket may be used to programme an on-board BS2-IC as per the standard BS2-IC project board.

POWER SUPPLY

The SSDB uses a separate regulator for the Servos and one for the electronics. The SSDB will accept DC voltages from 7.5 to 12v DC via the supplied power plug (2.1mm dia type, centre positive).
The servo regulator is rated at 1Amp and should be sufficient for most applications though, as servos may draw up to 200mA each when under load, high load applications involving a number of servos may need to be provided with a separate power supply (4.8-6v DC). If the heatsink becomes too hot to touch then consider a separate power supply and/or reduce the supply voltage.

HOOK-UP

Select the required baud rate, SSDB address, servo range and signal source using the supplied jumpers.
Connect the servos to the 3 pin headers at the end of the SSDB making sure the colour of the leads corresponds to those noted on the board- especially the red and black wires. Futaba servos use a white cable for the signal line; Hitec use yellow. Both will run equally well on the SSDB.
Connect a suitable power supply to the power socket.

Reg Office: Milford House, 120 High Street, South Milford, LEEDS LS25 5AQ United Kingdom
Reg Number: 4222329 Tel: 01977 683665 Fax 01977 681465 www.milinst.com

The red LED should light and the servos move to their mid positions. The servos should resist being changed by finger pressure.
If this is not the case then switch off immediately and check the polarity of the incoming power supply and the servo wire colours.
With the servos in their mid position run the control software either from the BS2 or from a PC using a serial cable connected to port 1 or 2.
Once a valid control signal is received, the LED will extinguish and will thereafter only light momentarily when a synchronization byte is received.
If this is not the case and/or the servos will not respond under software control then check the setting jumpers and the serial link.

PROGRAMMING EXAMPLES

The following code snippets illustrate how to run the SSDB from the BS2-IC and QBASIC.

```
'BS2-IC (Stamp2)
sync      con     255       'sync byte set to 255
servo     var     byte      'servo number
position  var     byte      'servo position
n96n      con     $4054     'comms set to 9600 baud
signal    con     11        'Controller set to pin stamp pin 11
start:
          for servo = 0 to 7                          ' each servo in turn
                    for position= 0 to 254 step 1 ' sweep the range in increments of 0.36 deg
                    serout signal,n96n,[sync,servo,position]          'send the signal
                    next
                    for position= 254 to 0 step 1 ' sweep the range in increments of 0.36 deg
                    serout signal,n96n,[sync,servo,position]          'send the signal
                    next
          next
goto start:                                   'Repeat again

QBASIC
DEFINT A-Z
SYNC.BYTE=255
' The following sets up comms port 1 for 9600 baud operation, ensure the SSDB is also set to
' 9600 baud.

OPEN "com1:9600,N,8,1,CD0,CS0,DS0,OP0"  FOR OUTPUT AS #1
PRINT "AT THE PROMPT, TYPE THE SERVO NUMBER (0 TO 15), A COMMA,
PRINT "AND A POSITION VALUE (0 TO 254)
PRINT " PRESS <CNTL>- <BREAK> TO END"
AGAIN:
          LOCATE 8,1
          PRINT "
          LOCATE 8,1
          INPUT  "SERVO, POSITION>", SERVO,POSITION
          IF SERVO>15 THEN SERVO = 15
          IF SERVO<0 THEN SERVO = 0
          IF POSITION>254 THEN POSITION = 254
          IF POSITION<0 THE POSITION = 0
          PRINT #1,CHR$(SYNC.BYTE); CHR$(SERVO); CHR$(POSITION);
GOTO AGAIN
```

SERVOS

Most hobby type servos operate in a standard manner. Their movement is proportional to the pulse width of an incoming signal stream. These pulses are normally between 1 and 2 ms in duration and must be repeated every 20msecs or so. A pulse width of 1.5msecs will generally centre the servo, a pulse width of 2msec will cause the servo to move to the +45deg position and one of 1.0msecs will move the servo to the -45 deg position.
Hobby servos are generally rated for a total movement of 90 degrees though to allow for set-up, the actual range is often greater than 90 degrees. The SSDB allows you to use this extra range but care should be taken to ensure the servo is not driven beyond its maximum physical range otherwise damage may occur.
If using the 180 degree range, it is recommended that the servo be moved a few degrees at a time until the end of its travel is detected (the servo stalls). Once the maximum travel limits have been established, these should be incorporated into the driving software to prevent servo damage.