

## Servo Driver Board Software (1-927)

The Servo Driver Board Controller software is designed for use with the 8 and 12 Servo Driver Boards supplied by Milford Instruments.

It can be used to control the servos in real time using the control panel, or for the creation and development of scripts to automate a series of servo motions for a wide range of applications (not enabled in the freeware version).

### Setting up the servos

The first screen that you are presented with, when the application begins is the Servo Settings screen. This allows you to tell the application which servo driver board you are using, and define the home position of the servos, and their maximum and minimum positions. You can also name your servos for your convenience.

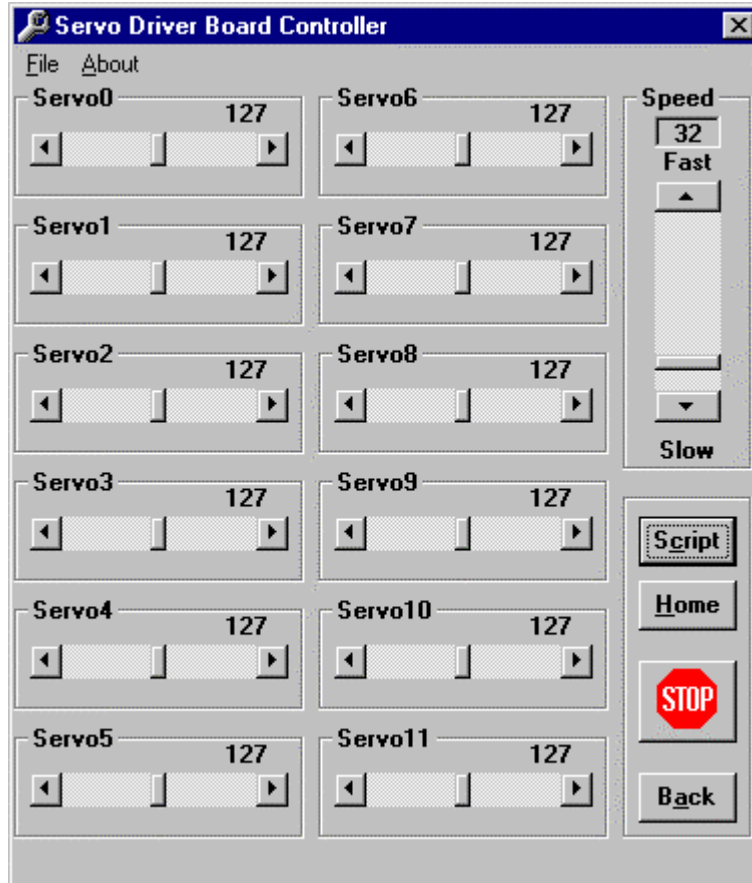
	Name	Min	Home	Max
Servo 0	Servo0	0	127	254
Servo 1	Servo1	0	127	254
Servo 2	Servo2	0	127	254
Servo 3	Servo3	0	127	254
Servo 4	Servo4	0	127	254
Servo 5	Servo5	0	127	254
Servo 6	Servo6	0	127	254
Servo 7	Servo7	0	127	254
Servo 8	Servo8	0	127	254
Servo 9	Servo9	0	127	254
Servo 10	Servo10	0	127	254
Servo 11	Servo11	0	127	254

Select either the servo-8 or the servo-12 board options, the required comms port (1-4) and the required board rate (servo-8 board only).

Once you have made all the alterations that are necessary, you can save them in an .SST file (Servo Settings). The format of this file is described later in the document, so that (if you wish) you can alter or create .SST files with any ASCII editor.

## Controlling the servos

Clicking on the Controls button will open the following screen:



With this screen, you can control the servos directly. The number above each servo scroll bar shows the current position of each servo.

By moving the scrollbar and then releasing it, you can indicate where you would like the servo to move to, and it will move there at the rate dictated by the 'Speed' scrollbar. Clicking on the arrows on the end of the servo scrollbars will also move the servo at an equivalent rate.

The Speed is given in steps per second. Hence 255 means the servo will turn full range in 1 second. By moving the slider, the speed of each servo will be set to the value indicated in the box above the slider.

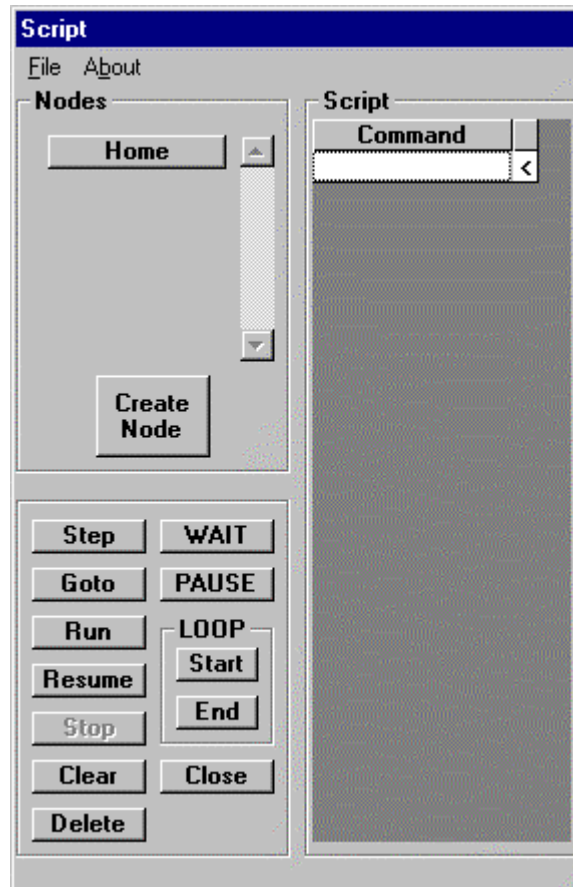
By clicking the 'Home' button, you can send all of the servos to their home positions. They will travel there at the speed indicated by the 'Speed' scrollbar.

Clicking the 'STOP' button will bring the servos to an immediate halt and move the Servo Control sliders to the position indicated by the number above the slider.

Clicking the 'Back' button will take you back to the settings screen.

## Scripting

Click on the 'Script' button on the Controls screen, the following additional window will open:



This is the scripting window. Here you can automate sequences of movements and play them over and over. This is done in the following manner.

### Creating Nodes

First, move the servos to your first chosen position using the Controls window. Click the 'Create Node' button to store the position of each servo at the node and the speed at which the servo should move to the current position (set by the 'Speed' scrollbar on the Controls screen).

Name the node as appropriate and it will appear below the 'Home' Button.

Create further nodes in a similar manner as required. The Home node is predefined and cannot be changed. It mirrors the function of the 'Home' button on the Controls screen.

Once Nodes have been created, you can edit their properties by right-clicking the node button and choosing 'Edit'. This will send to servos to the node and open the 'Edit Node' window.

Using the Edit Node window, you can change the name of the node and the Positions recorded in that node.

	Position	Speed
0	127	1
1	127	1
2	127	1
3	127	1
4	127	1
5	127	1
6	127	1
7	127	1
8	127	1
9	127	1
10	127	1
11	127	1

The 'Controls' screen remains active and by moving the sliders on the controls screen, you change the values in the Position boxes on the 'Edit Node' screen. You can also change the positions by typing them in manually.

You can change the speed associated with each node, but this must be done manually. You can work out the value to put in here by looking at the speed box on the Controls screen.

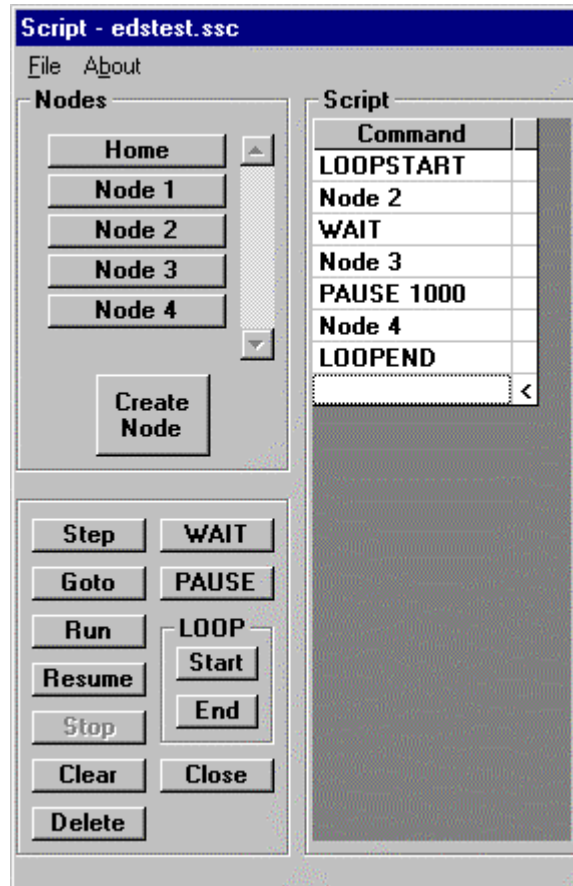
If you decide that you no longer need a node, you can delete it. This will remove the button from the list of nodes, and will remove all instances of that node from the script.

Once you are happy with all of your nodes, you can create your script.

## Movement Scripting

The scripting window allows you to 'record' a series of movements that you can play over and over, edit, save and reopen as your needs dictate.

Please note that the freeware version of the software limits the number of nodes to 2 and does not allow script sessions to be saved or retrieved.



To create a script, you merely click the buttons corresponding to the command that you wish to place in the script. The '<' character indicates the current insertion point, i.e. where the next command will be placed.

Clicking on a node places a command that, when executed, makes the servos move to that node. By right clicking on the node button, you also have the choice of making the servos go to that point, without inserting the command into the script.

The WAIT command suspends the running of the script, until the user responds to a request to continue.

The PAUSE command suspends the running of the script for a user-defined period measured in milliseconds.

The LOOPSTART and LOOPEND commands create a loop such that all commands between these two are processed over and over again until the running of the script is stopped.

By clicking anywhere on the script, you can move the insertion point to that location. If you then insert a command, the program will overwrite the current command with your new command.

To place a command between two pre-existing ones, double click the lower of the two, and a space will be created for you to place your new command.

To delete a command, click on it and press the 'Delete' button.

To get rid of the entire script, press the 'Clear' button.

On pressing the 'Step' button, The current insertion point steps to the next command on the script and executes it. If the current insertion point is the end of the script then it goes up to the start. By pressing the 'Run' button, the script will be executed from the very beginning without stopping until the end is reached, or the 'Stop' button is pressed. The 'Resume' button, on the other hand, commences execution from the current point.

If you have finished using the Script window, press the 'Close' button. You can do this, even if servos are still moving. On closing the Script window, the servos' speeds will revert to whatever is displayed on the Speed scrollbar.

You can save or load your scripts and node sets as a ServoScript file (\*.SSC). This file contains all the information about the nodes and the script. Opening a SSC file will overwrite all information that you have currently on screen. The format of an SSC file is included below.

NB SSC and SST files are loaded independently. If your servo settings are such that the Max and Min values of a given servo are exceeded by the position of that servo in the SSC file then you will be warned, but allowed to carry on.

i.e. Servo 1 has Min=10, Max=50, but Node 1 of your loaded file has Servo 1 going to 120, then problems will occur.

It is recommended that you do not carry on without rectifying this, otherwise your script may get stuck when it tries to put a servo in a position that it is not allowed to go to!

## File Formats

SST Files (Servo Settings files) contain the information saved and opened on the settings screen. They look like this:

```
"Servo0","0","127","254",0
"Servo1","0","127","254",1
"Servo2","0","127","254",2
"Servo3","0","127","254",3
"Servo4","0","127","254",4
"Servo5","0","127","254",5
"Servo6","0","127","254",6
"Servo7","0","127","254",7
"Servo8","0","127","254",8
"Servo9","0","127","254",9
"Servo10","0","127","254",10
"Servo11","0","127","254",11
"***",123
```

The format is straightforward. For every servo whose characteristics you save, a line is created:

*"name","min","home","max",servonumber*

Servonumber just tells the program which servo to ascribe those attributes to. The file entries do not have to be in servo order, and you don't need to include a line for every servo.

What you **must** include is the final line:

```
"***",123
```

This is the end of file indicator (three asterisks). It is only strictly necessary for files with less than 12 entries, but it is good practice to include it always.

SSC files (ServoScript files) save all your nodes' characteristics and the details of your script. They look like this:

```
"SSCFILE"
"Node 1"
"127,2,127,2,127,2,127,2,127,2,127,2,127,2,127,2,127,2,"
"Node 2"
"225,2,35,2,219,2,127,2,127,2,127,2,127,2,127,2,127,2,"
"Node 3"
"35,10,221,10,37,10,127,10,127,10,127,10,127,10,127,10,127,10,127,10,"
"Node 4"
"75,10,35,10,77,10,127,10,127,10,127,10,127,10,127,10,127,10,127,10,"
" *** "
"LOOPSTART"
"Node 2"
"WAIT"
"Node 3"
"PAUSE 1000"
"Node 4"
"LOOPEND"
" *** "
```

The first line **must** be "SSCFILE" to allow the program to recognise the file type, and prevent the loading of invalid files.

The following lines hold the node definitions. The Home node does not need to be included; in fact including it will merely lead to the repetition of the Home node.

Each node is defined in the following manner:

```
"nodename"
"Position 0,Speed 0, Position 1, Speed 1,...,Position 11, Speed 11,"
```

Note in particular, the last comma. This **is** required. The Position and Speed values can be any number from 0 to 255.

Next comes

```
" *** "
```

This is required to separate the node definitions and the script.

The script is then follows. Each command must be on a separate line and enclosed in double quote marks.

The end of file is marked by another three-asterisk string:

```
" *** "
```