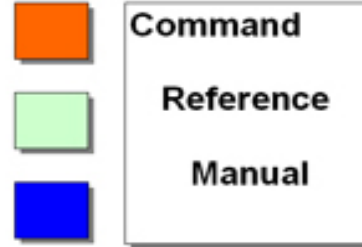


Operating Software Instruction Manual for Host Controlled Series of Motion Controllers



Control Panel



Command Reference Manual

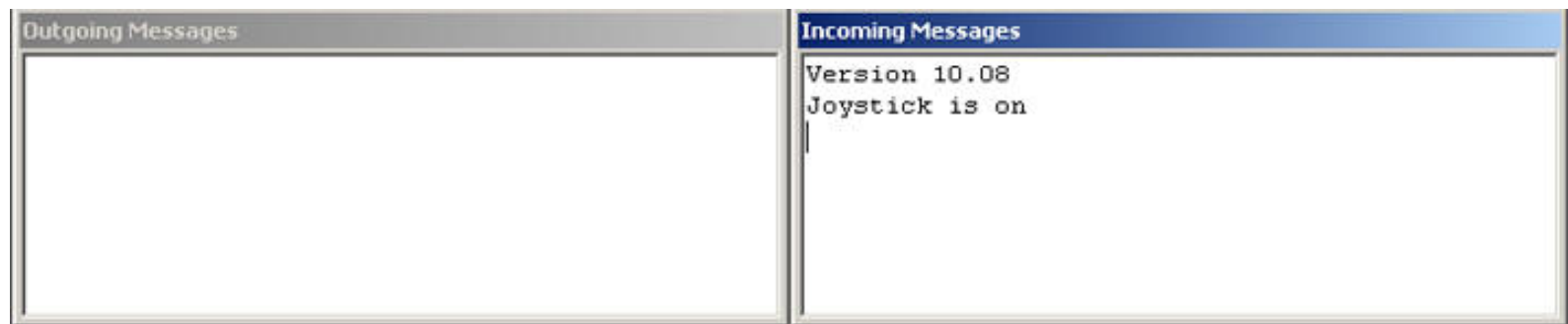


Host Controlled Operation

- [Power-up](#)
- [Help System](#)
- [Operation](#)
- [Teach Mode](#)
- [Position Entry Mode](#)
- [Repeating the Saved Motion Profile](#)
- [Setting the Motion Parameters](#)
- [Controlling the Outputs and Monitoring the Inputs](#)

Power-up

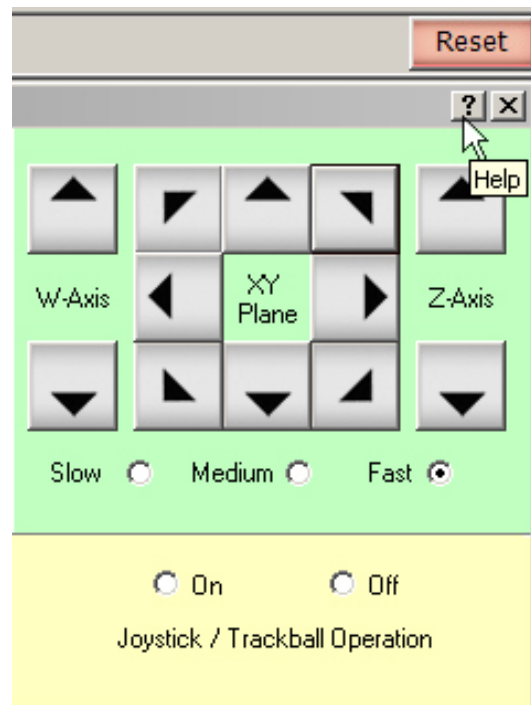
After power-up the status LED blinks and then remains lit. The controller sends a message to the PC showing its revision number that would be displayed in the Incoming Messages pane.



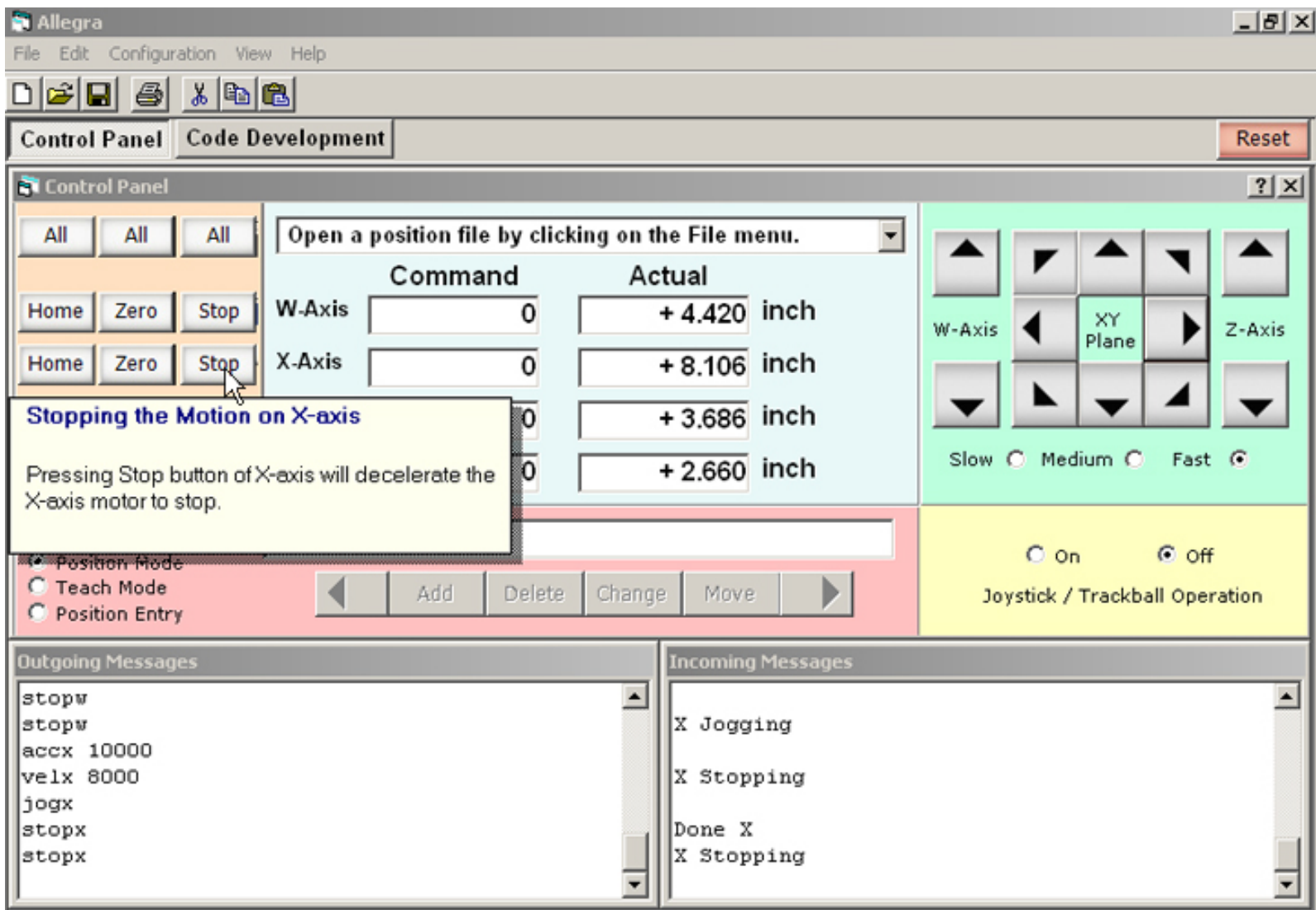
Help System

After installation of the software the online help system will be available. The online help system may be accessed by;

- 1) Click on "?" located on upper right corner of the control panel,

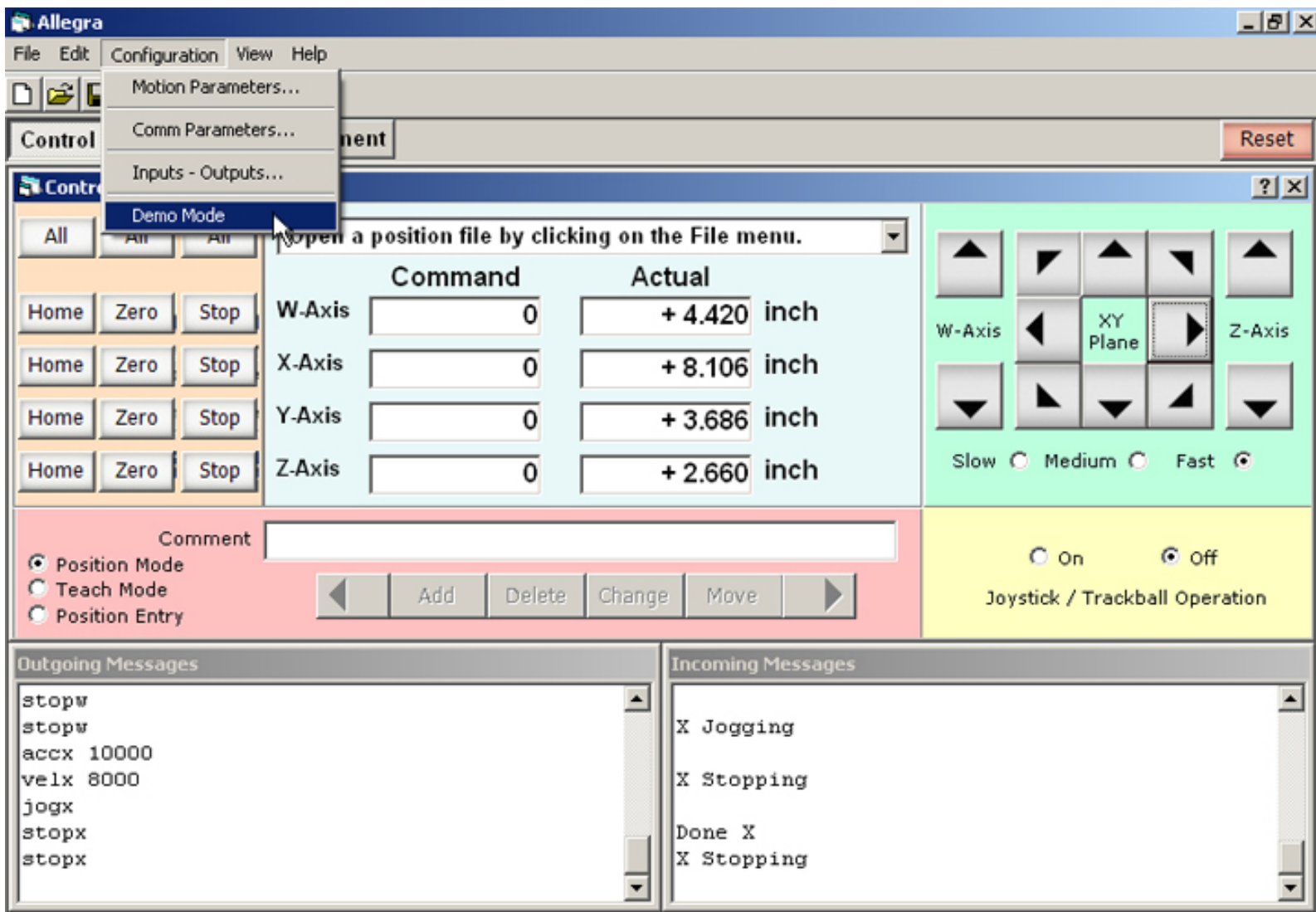


then drag it to any button and click. A message will explain the functionality of that button.

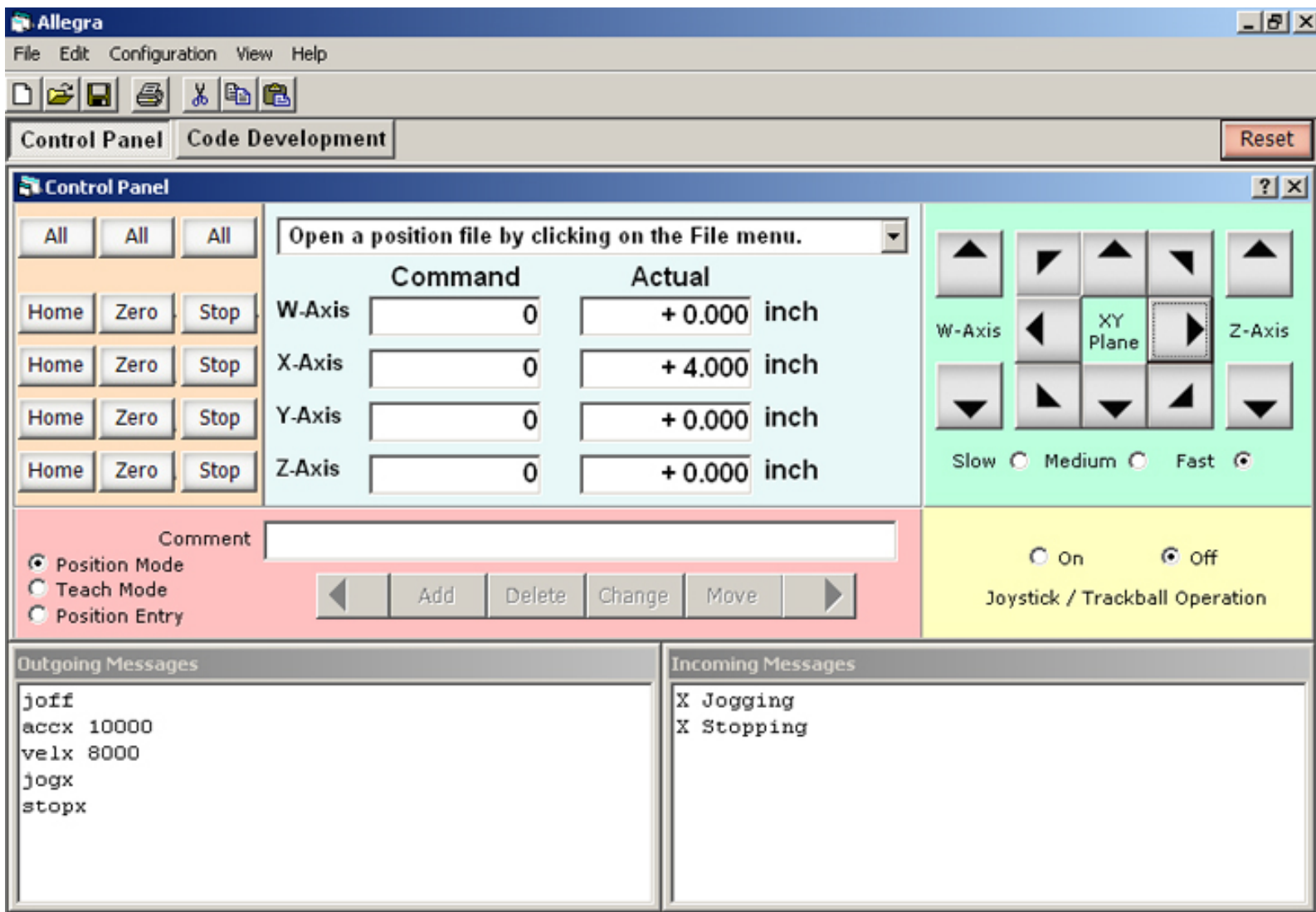


Operation

If the controller is not connected to the PC, Click on 'Configuration' then select 'Demo Mode' from the drop down menu to test drive the software without the controller.



If you click on any button, you would see the outgoing messages to the controller and the incoming messages from the controller in the corresponding panes.

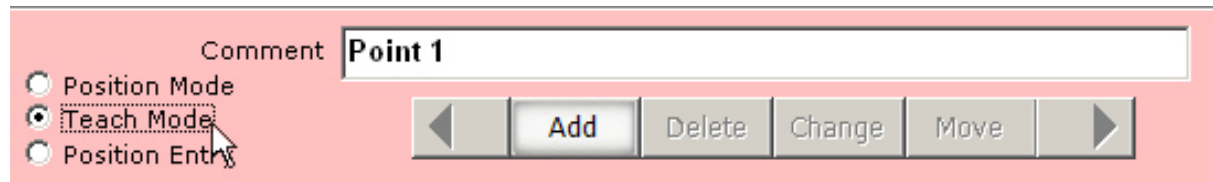


Teach Mode

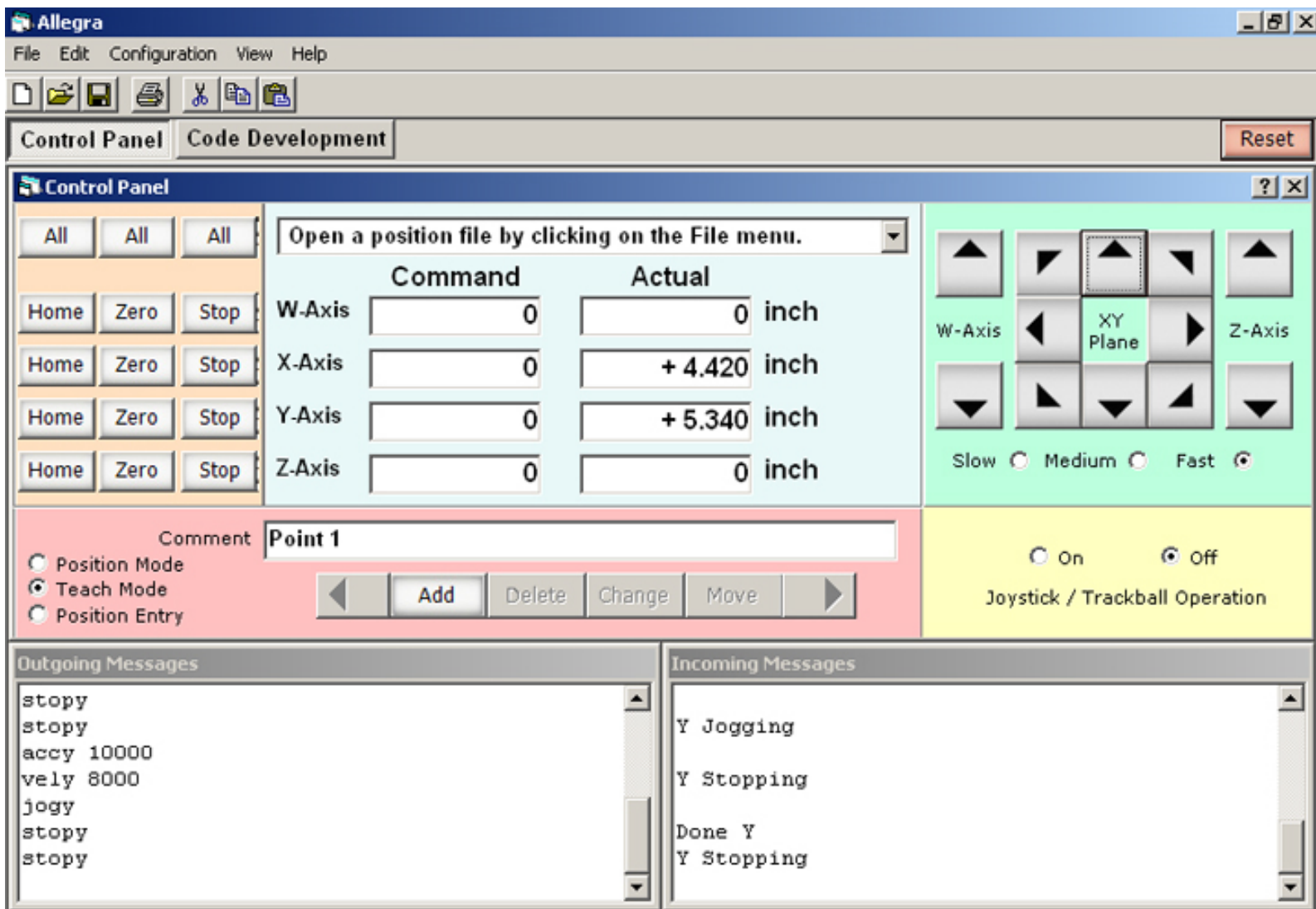
The controller may be used to learn and execute a motion profile without any programming.

To use the teach mode capability;

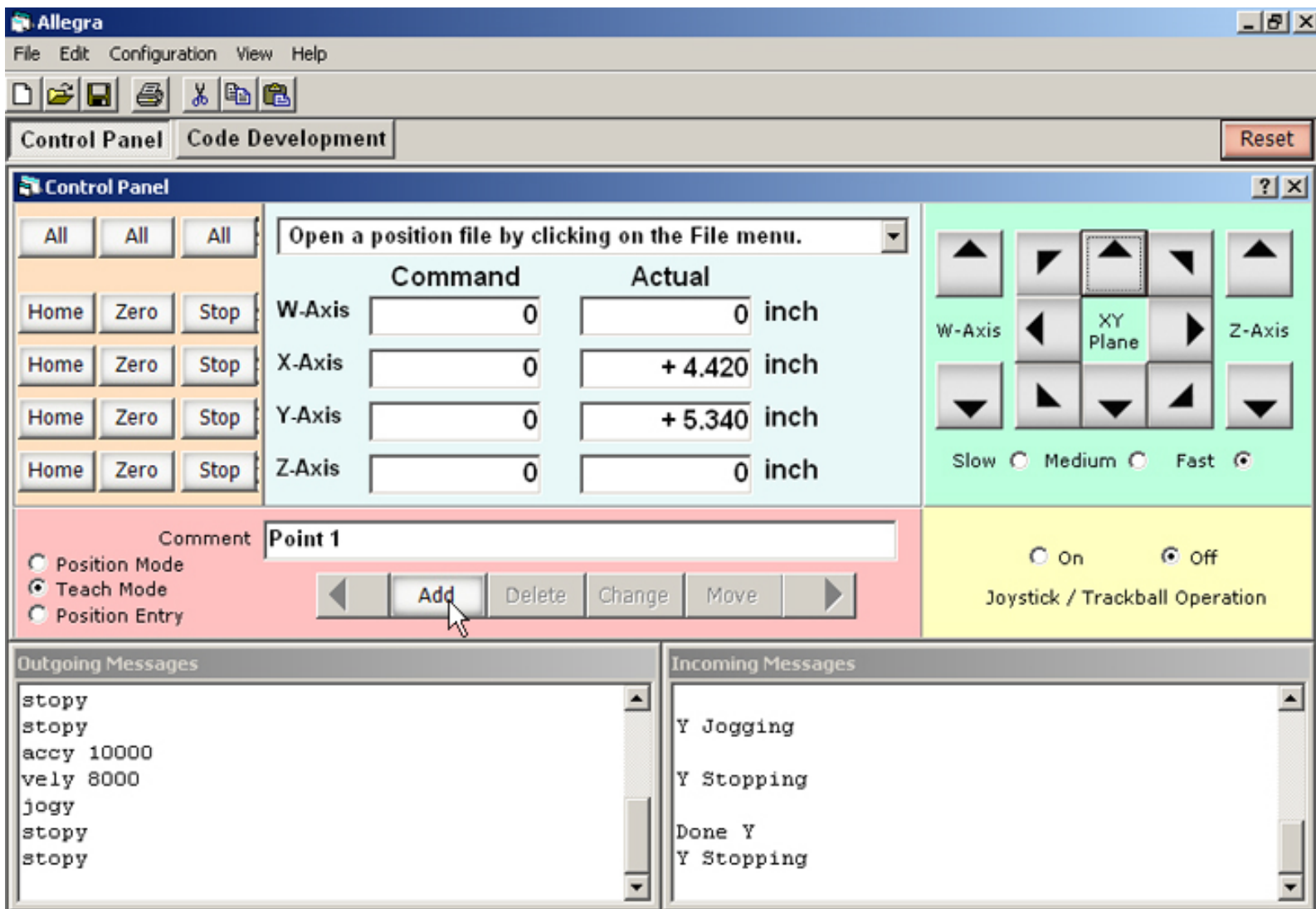
- 1) Select 'Teach Mode' option located on the lower left corner of the pane with the light red background.



- 2) Using the keypad located on the right hand side of the screen, and/or the joystick, and/or the trackball move the mechanism to different locations.



3) Press the 'Add' button. This will include the current point to the Position Log File that can be saved and retrieved.



4) The above steps may be repeated as many times as necessary.

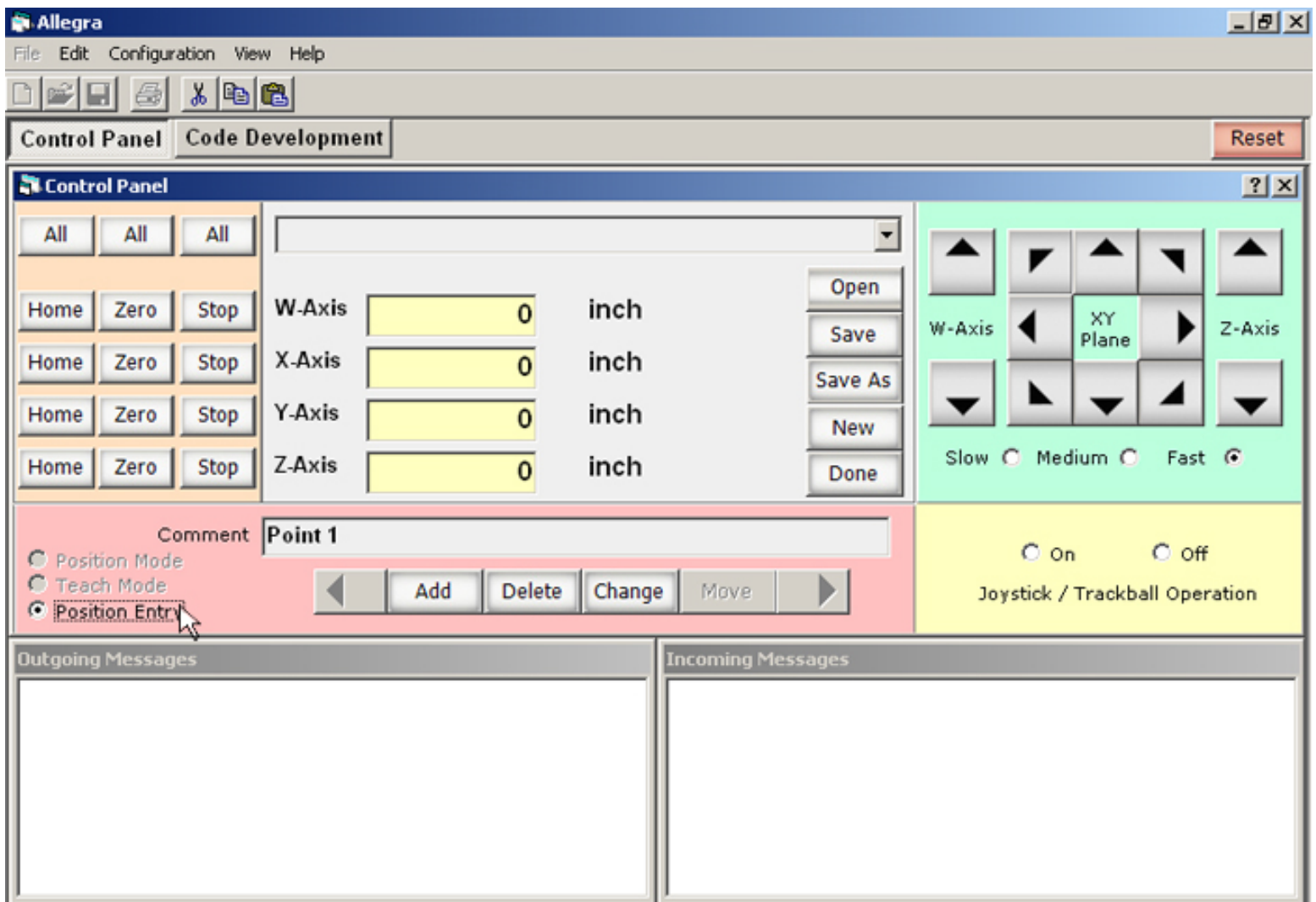
5) After completion, save the information in the Position Log File by clicking on 'File' then 'Save as'.

Position Entry Mode

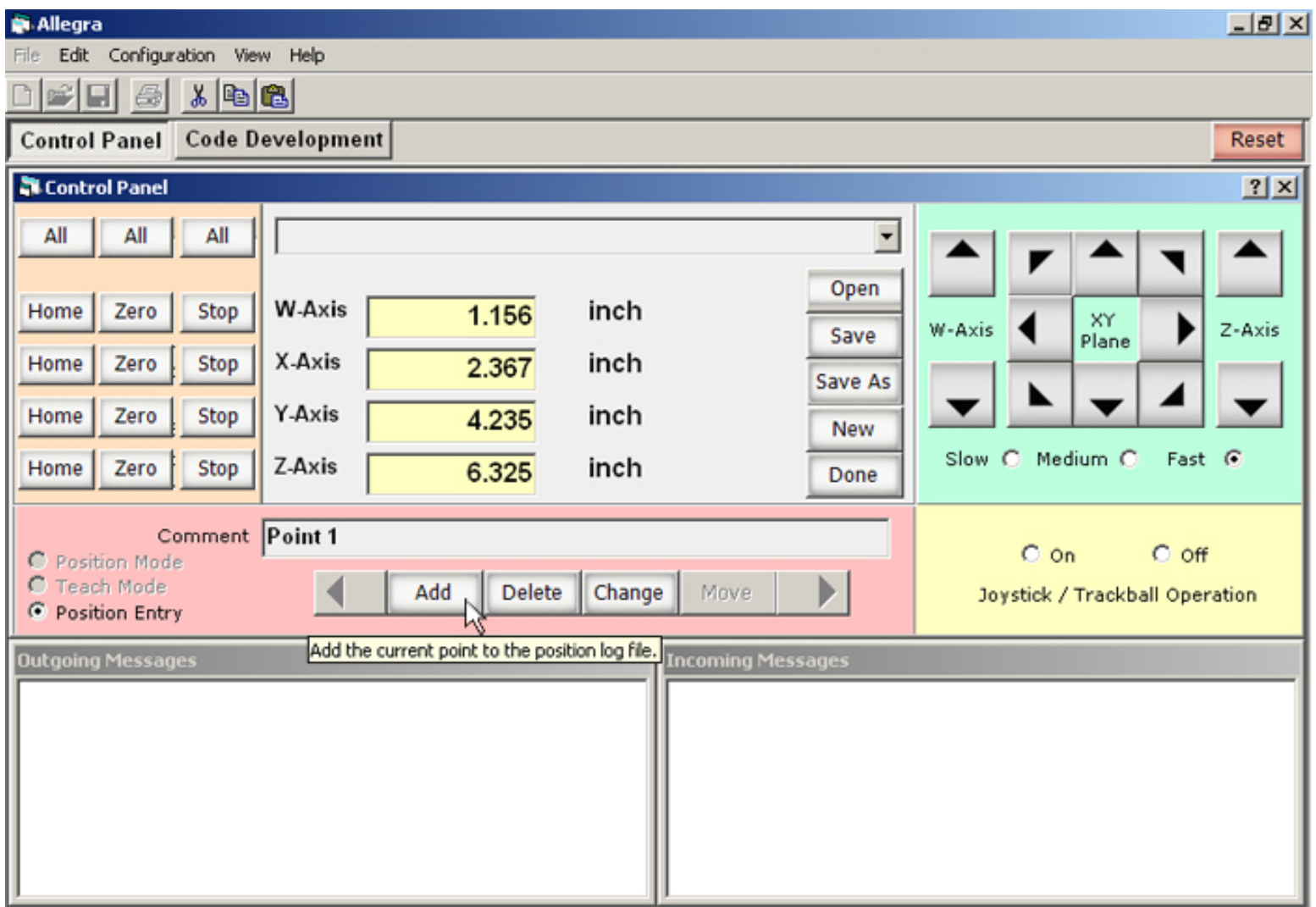
The software allows the user to enter the position of the points to move to.

To use the Position Entry Mode capability;

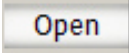
1) Select 'Position Entry' option located on the lower left corner of the pane with the light red background.

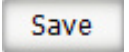



2) Enter the position information in each corresponding field then click on 'Add' button. This will include the current point to the Position Log File that can be saved and retrieved.

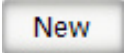


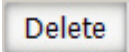
4) The above steps may be repeated as many times as necessary.

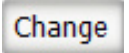
Clicking on  button will open an already existing file on the hard disk.

Clicking on  button will save the current information to the hard disk.

Clicking on  button will allow the user to save the information with a different name to the hard disk.

Clicking on  button will erase all the entered information.

Clicking on  button will erase the displayed record.

Clicking on  button will change the current record to the displayed values.

Clicking on  button will end the Position Entry mode and will return to Position Mode.

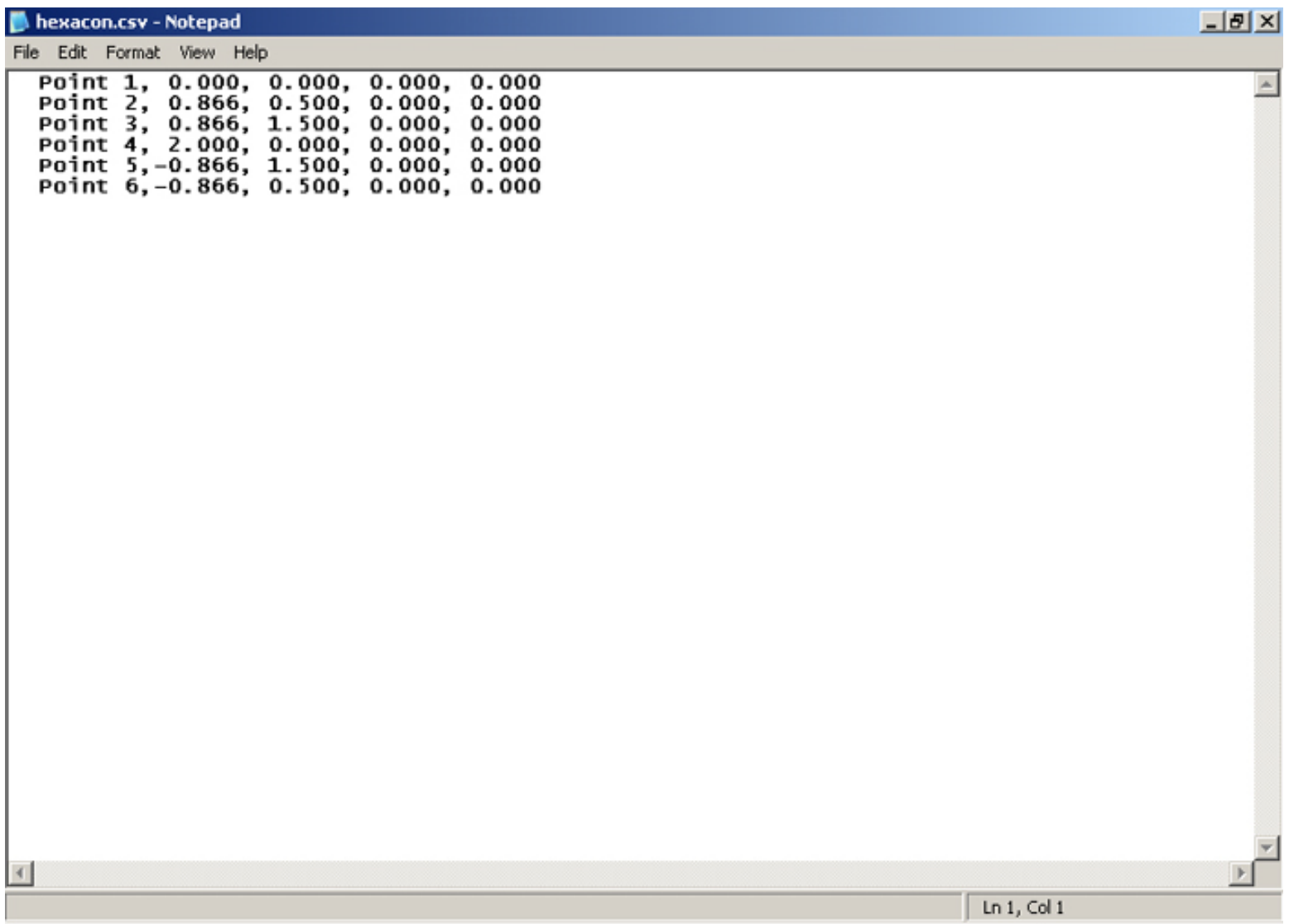
Make sure that motion parameters are properly set. See [Setting the Motion Parameter](#).

The position information is saved in .csv (Comma Separated Value) format. It is possible to generate and edit a profile using a spread sheet or text editor. However, the file must be saved as .csv file.

The screenshot shows a Microsoft Excel window titled "Microsoft Excel - hexacon.csv". The interface includes a menu bar (File, Edit, View, Insert, Format, Tools, Data, Window, Help, Adobe PDF), a search bar, a toolbar with various icons, and a ribbon with the "Arial" font and size "10" selected. The active cell is A1, containing the text "Point 1". The spreadsheet contains the following data:

	A	B	C	D	E	F	G	H	I	J	K	L
1	Point 1	0	0	0	0							
2	Point 2	0.866	0.5	0	0							
3	Point 3	0.866	1.5	0	0							
4	Point 4	2	0	0	0							
5	Point 5	-0.866	1.5	0	0							
6	Point 6	-0.866	0.5	0	0							
7												
8												
9												
10												
11												
12												
13												
14												
15												
16												
17												
18												
19												
20												
21												
22												
23												
24												

The status bar at the bottom shows "Ready" on the left and "NUM" on the right. The file name "hexacon" is visible in the bottom-left corner of the window.



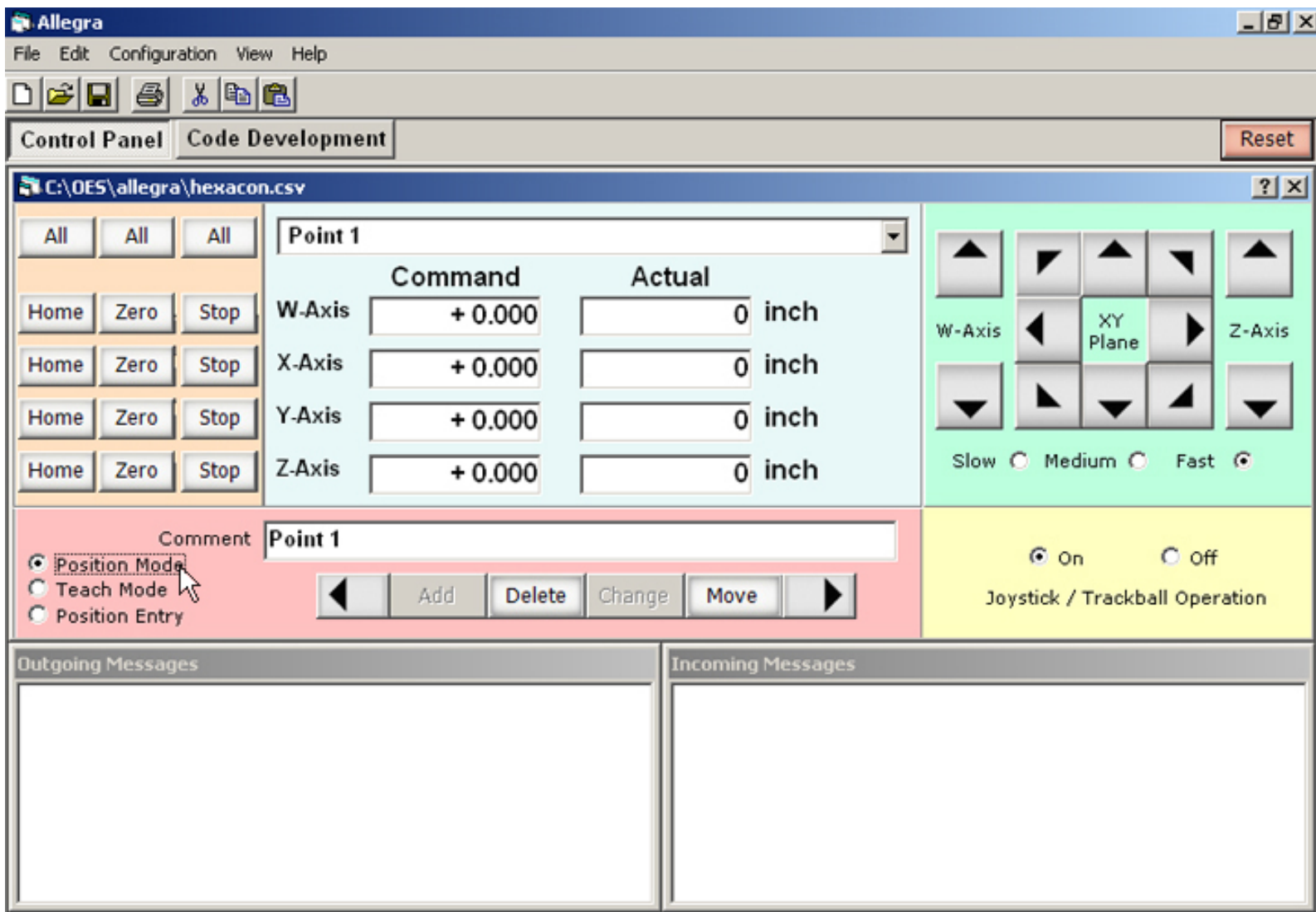
The image shows a Notepad window with the title 'hexacon.csv'. The menu bar includes 'File', 'Edit', 'Format', 'View', and 'Help'. The text content is as follows:

```
Point 1, 0.000, 0.000, 0.000, 0.000
Point 2, 0.866, 0.500, 0.000, 0.000
Point 3, 0.866, 1.500, 0.000, 0.000
Point 4, 2.000, 0.000, 0.000, 0.000
Point 5, -0.866, 1.500, 0.000, 0.000
Point 6, -0.866, 0.500, 0.000, 0.000
```

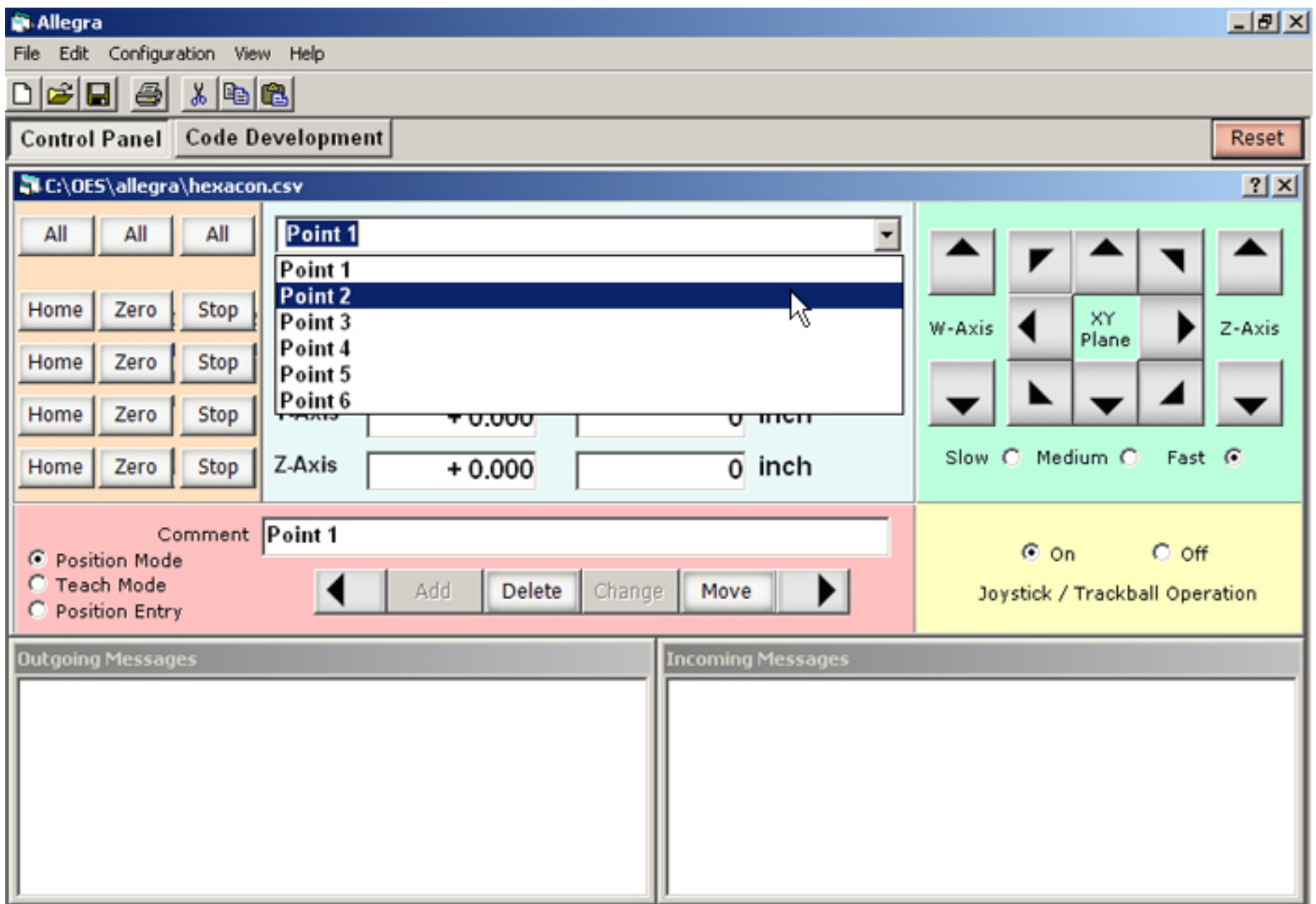
The status bar at the bottom right indicates 'Ln 1, Col 1'.

Repeating the Saved Motion Profile

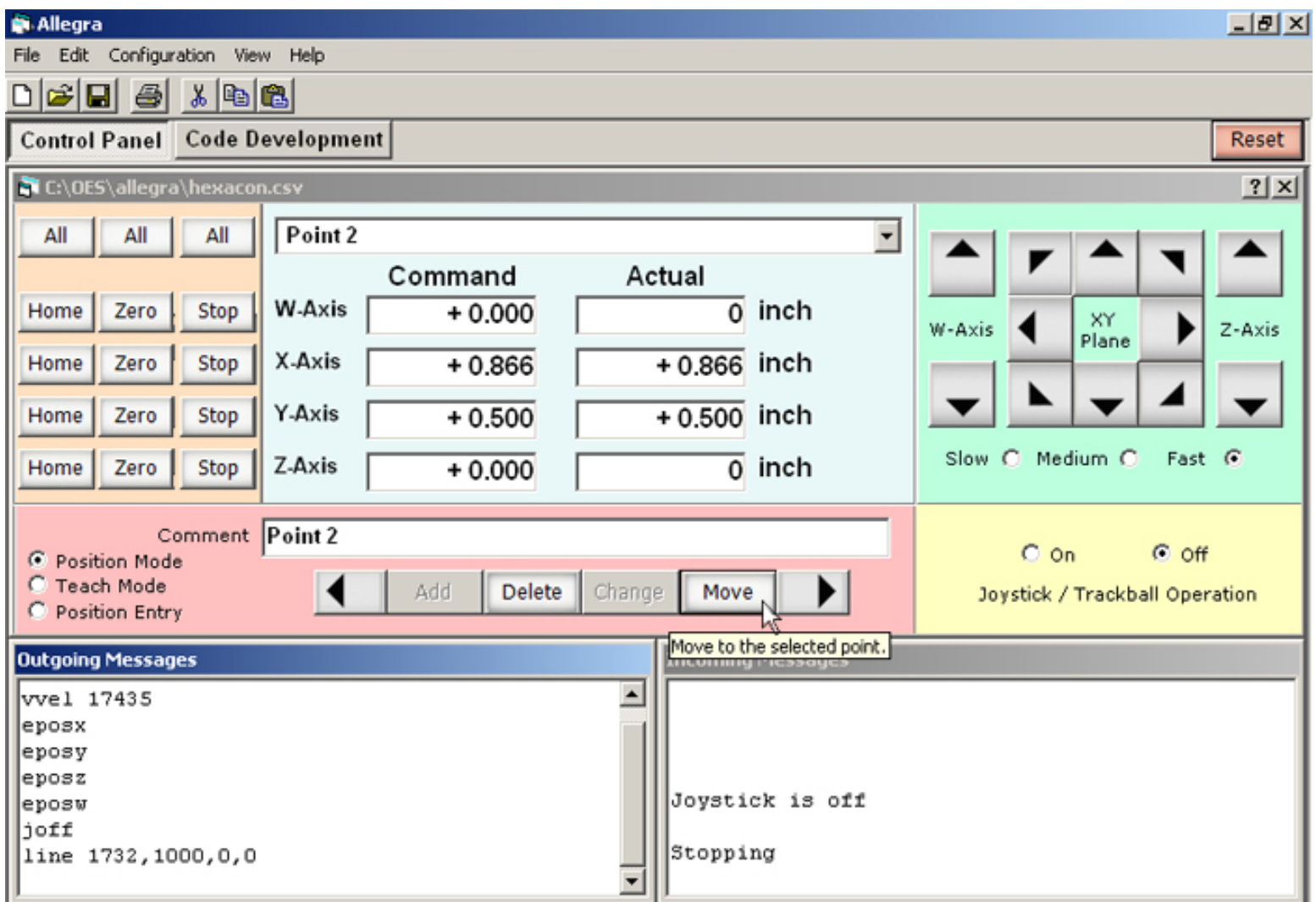
- 1) Select 'Position Mode' option located on the lower left corner of the pane with the light red background.



2) Using the drop down menu, located at the top of the pane with blue background, or the 'East' and 'West' pointing arrows in the pane with light red background, select a point to move to.



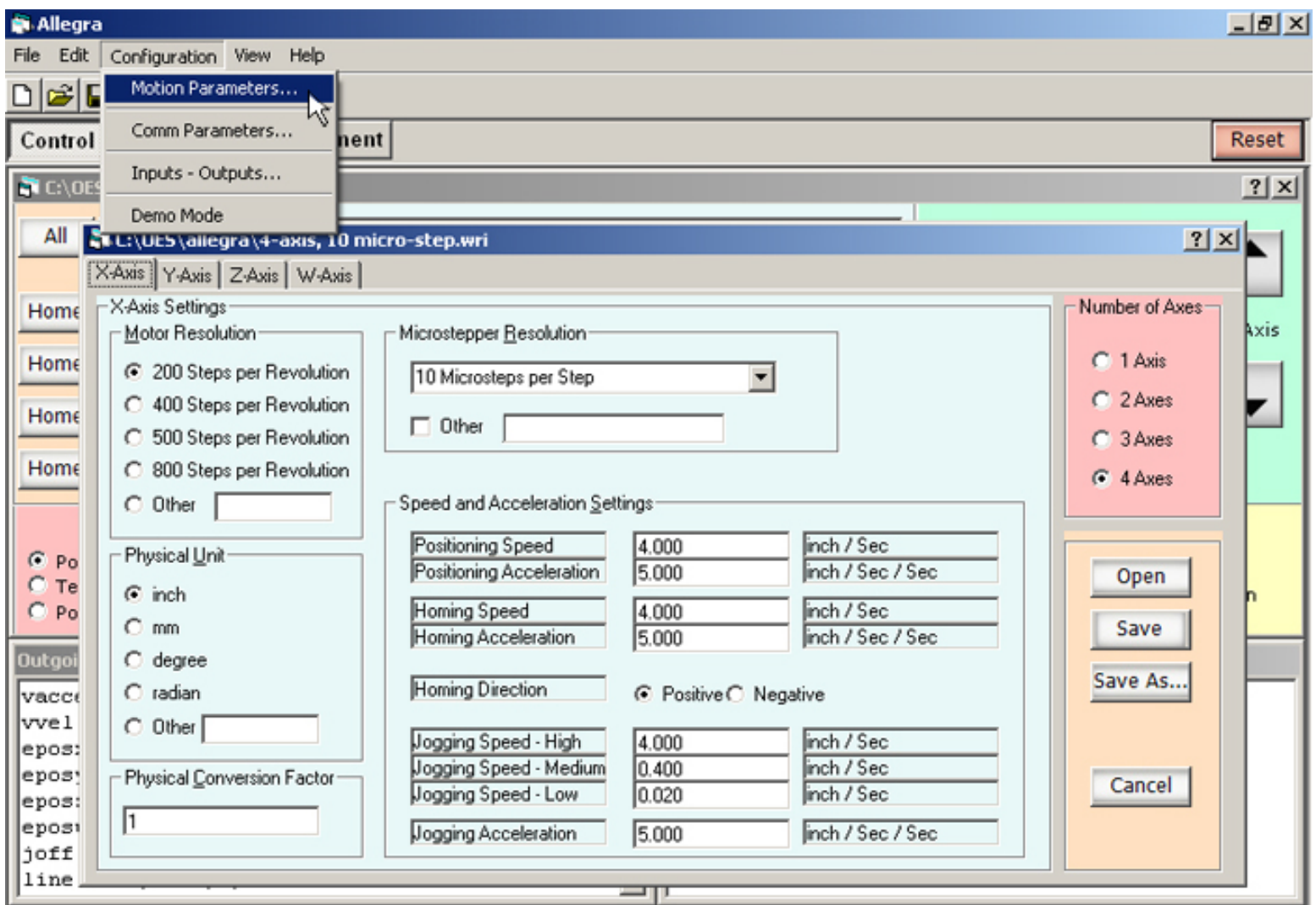
3) Press 'Move'; the mechanism will go to the selected point.



It is that simple! You do not have to write a single line of code.

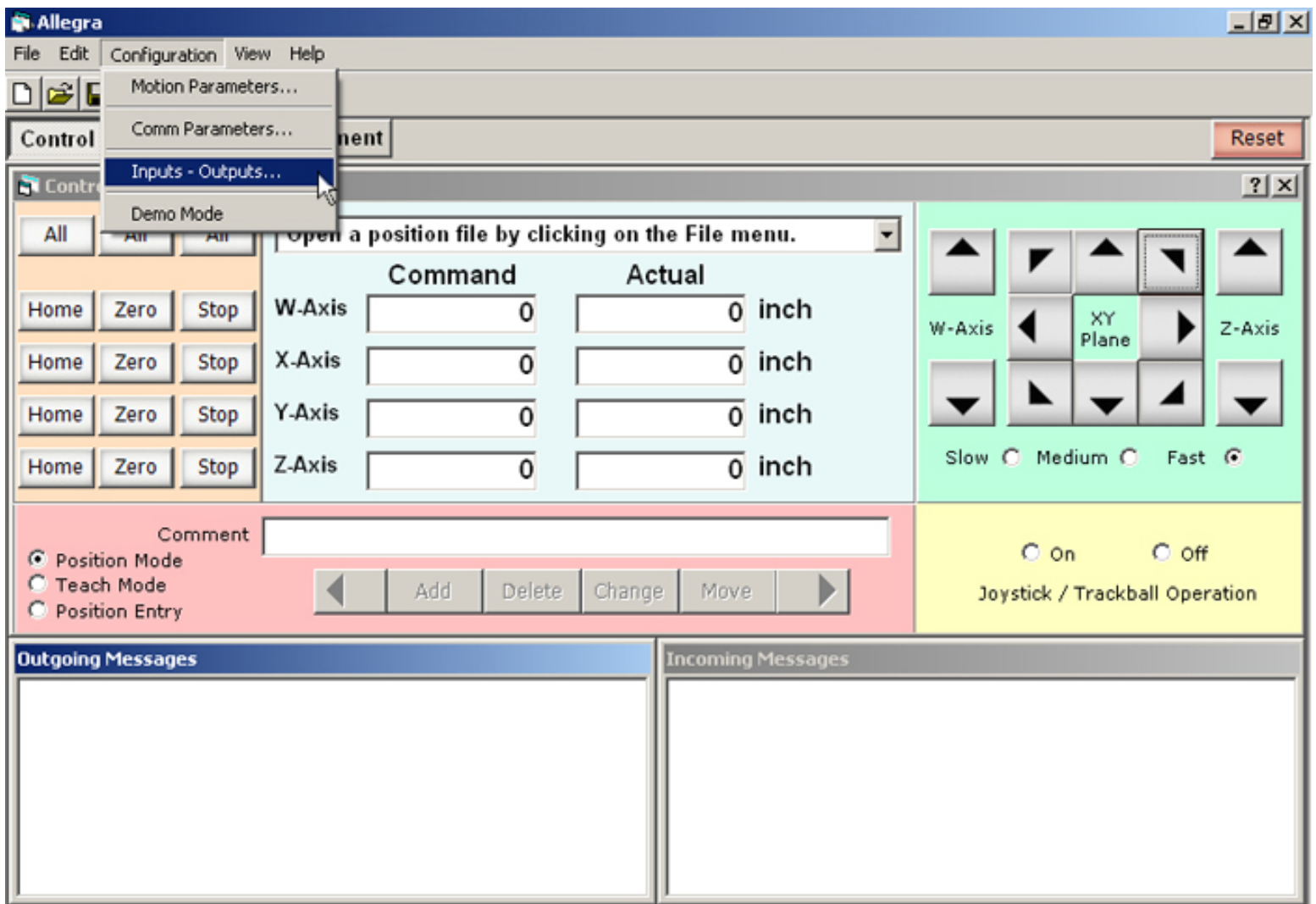
Setting the Motion Parameters

All the motion parameters are set in the 'Configuration Panel' under 'Configuration' -> 'Motion Parameters'.



Controlling the Outputs and Monitoring the Inputs

To control the outputs and monitor the inputs, click on 'Configuration' -> 'Inputs - Outputs'.



The following display will appear that allows the user to set or reset each output individually and monitor all analog and digital inputs.

Allegra File Edit Configuration View Help

Control Panel Code Development Reset

In - Output

Discrete Outputs				Discrete Inputs				X-Motor	Y-Motor	Z-Motor	W-Motor
Out 1	<input checked="" type="radio"/> High <input type="radio"/> Low	Out 9	<input type="radio"/> High <input type="radio"/> Low	In 1	<input type="text" value="High"/>	In 9	<input type="text" value="High"/>	+ Limit <input type="radio"/>	+ Limit <input type="radio"/>	+ Limit <input type="radio"/>	+ Limit <input type="radio"/>
Out 2	<input type="radio"/> High <input type="radio"/> Low	Out 10	<input type="radio"/> High <input type="radio"/> Low	In 2	<input type="text" value="High"/>	In 10	<input type="text" value="High"/>	- Limit <input type="radio"/>	- Limit <input type="radio"/>	- Limit <input type="radio"/>	- Limit <input type="radio"/>
Out 3	<input type="radio"/> High <input type="radio"/> Low	Out 11	<input type="radio"/> High <input type="radio"/> Low	In 3	<input type="text" value="High"/>	In 11	<input type="text" value="High"/>	Home <input type="radio"/>	Home <input type="radio"/>	Home <input type="radio"/>	Home <input type="radio"/>
Out 4	<input type="radio"/> High <input type="radio"/> Low	Out 12	<input type="radio"/> High <input type="radio"/> Low	In 4	<input type="text" value="High"/>	In 12	<input type="text" value="High"/>	Analog Inputs		Joystick Keys	
Out 5	<input type="radio"/> High <input type="radio"/> Low	Out 13	<input type="radio"/> High <input type="radio"/> Low	In 5	<input type="text" value="High"/>	In 13	<input type="text" value="High"/>	Analog-X	<input type="text" value="127"/>	Upper Right <input type="radio"/>	
Out 6	<input type="radio"/> High <input type="radio"/> Low	Out 14	<input type="radio"/> High <input type="radio"/> Low	In 6	<input type="text" value="High"/>	In 14	<input type="text" value="High"/>	Analog-Y	<input type="text" value="127"/>	Middle Right <input type="radio"/>	
Out 7	<input type="radio"/> High <input type="radio"/> Low	Out 15	<input type="radio"/> High <input type="radio"/> Low	In 7	<input type="text" value="High"/>	In 15	<input type="text" value="High"/>	Analog-Z	<input type="text" value="131"/>	Lower Left <input type="radio"/> Lower Right <input type="radio"/>	
Out 8	<input type="radio"/> High <input type="radio"/> Low	Out 16	<input type="radio"/> High <input type="radio"/> Low	In 8	<input type="text" value="High"/>	In 16	<input type="text" value="High"/>	Quadrature Inputs			
								Quad-1	<input type="text" value="0"/>	Quad-X	<input type="text" value="0"/>
								Quad-2	<input type="text" value="0"/>	Quad-Y	<input type="text" value="0"/>
										Quad-Z	<input type="text" value="0"/>
										Enable	<input type="checkbox"/>

Done

Outgoing Messages

Incoming Messages

MOTION

[ACCW](#) *value* Sets the linear acceleration of the W-axis to *value* .

[ACCX](#) *value* Sets the linear acceleration of the X-axis to *value* .

[ACCY](#) *value* Sets the linear acceleration of the Y-axis to *value* .

[ACCZ](#) *value* Sets the linear acceleration of the Z-axis to *value* .

[HOMEW](#) Instructs home searching sequence of the W-axis.

[HOMEX](#) Instructs home searching sequence of the X-axis.

[HOMEY](#) Instructs home searching sequence of the Y-axis.

[HOMEZ](#) Instructs home searching sequence of the Z-axis.

[JOGW](#) Instructs jogging mode of the W-axis.

[JOGX](#) Instructs jogging mode of the X-axis.

[JOGY](#) Instructs jogging mode of the Y-axis.

[JOGZ](#) Instructs jogging mode of the Z-axis.

[MOVAALL](#) Begins an absolute move on all axes.

[MOVAW](#) Begins an absolute move on the W-axis.

[MOVAX](#) Begins an absolute move on the X-axis.

[MOVAY](#) Begins an absolute move on the Y-axis.

[MOVAZ](#) Begins an absolute move on the Z-axis.

[MOVRALL](#) Begins a relative move on all axes.

[MOVRW](#) Begins a relative move on the W-axis.

[MOV RX](#) Begins a relative move on the X-axis.

[MOVRY](#) Begins a relative move on the Y-axis.

[MOV RZ](#) Begins a relative move on the Z-axis.

[POSW](#) *value* Sets the distance to move of the W-axis to *value* .

[POSX](#) *value* Sets the distance to move of the X-axis to *value* .

[POSY](#) *value* Sets the distance to move of the Y-axis to *value* .

[POSZ](#) *value* Sets the distance to move of the Z-axis to *value* .

[STOPALL](#) Stops the motion of all axes.

[STOPW](#) Stops the motion of the W-axis.

[STOPX](#) Stops the motion of the X-axis.

[STOPY](#) Stops the motion of the Y-axis.

[STOPZ](#) Stops the motion of the Z-axis.

[VELW](#) *value* Sets the maximum step rate of the W-axis to *value* .

[VELX](#) *value* Sets the maximum step rate of the X-axis to *value* .

[VELY](#) *value* Sets the maximum step rate of the Y-axis to *value* .

[VELZ](#) *value* Sets the maximum step rate of the Z-axis to *value* .

COORDINATED MOTION

LINE	Makes a coordinated linear motion.
VACCEL <i>value</i>	Sets the vector acceleration to <i>value</i> .
VVEL <i>value</i>	Sets the vector velocity to <i>value</i> .
ARC	Generates a 2-dimensional circular arc segment.

GENERAL CONFIGURATION

EPOSW	Equates the current position of the W-axis to the value of the corresponding step counter.
EPOSX	Equates the current position of the X-axis to the value of the corresponding step counter.
EPOSY	Equates the current position of the Y-axis to the value of the corresponding step counter.
EPOSZ	Equates the current position of the Z-axis to the value of the corresponding step counter.
JOFF	Disables the joystick and trackball operation.
JON	Enables the joystick and trackball operation.
MOFFW	Turns the motor driver of the W-axis off.
MOFFX	Turns the motor driver of the X-axis off.
MOFFY	Turns the motor driver of the Y-axis off.
MOFFZ	Turns the motor driver of the Z-axis off.
MONW	Turns the motor driver of the W-axis on.
MONX	Turns the motor driver of the X-axis on.
MONY	Turns the motor driver of the Y-axis on.
MONZ	Turns the motor driver of the Z-axis on.
MSGOFF	Turns off the feedback messages sent from the controller.
MSGON	Turns on the feedback messages sent from the controller.
SPOSW <i>value</i>	Sets the current position of the W-axis to <i>value</i> .
SPOSX <i>value</i>	Sets the current position for the X-axis to <i>value</i> .
SPOSY <i>value</i>	Sets the current position for the Y-axis to <i>value</i> .
SPOSZ <i>value</i>	Sets the current position for the Z-axis to <i>value</i> .
SQUADW <i>value</i>	Sets the W-axis quadrature decoder to <i>value</i>
SQUADX <i>value</i>	Sets the X-axis quadrature decoder to <i>value</i> .
SQUADY <i>value</i>	Sets the Y-axis quadrature decoder to <i>value</i> .
SQUADZ <i>value</i>	Sets the Z-axis quadrature decoder to <i>value</i> .

DATA FLOW

CLRBIT <i>value</i>	Resets the discrete output specified by <i>value</i> to low state.
IN	Report the input ports
OUT <i>value</i>	Writes the <i>value</i> to the output ports.

RQUADW	The controller sends the W-axis quadrature decoder value to the serial port.
RQUADX	The controller sends the X-axis quadrature decoder value to the serial port.
RQUADY	The controller sends the Y-axis quadrature decoder value to the serial port.
RQUADZ	The controller sends the Z-axis quadrature decoder value to the serial port.
RSTSW	Reports the status of W-axis
RSTSX	Reports the status of X-axis
RSTSY	Reports the status of Y-axis
RSTSZ	Reports the status of Z-axis.
RW	Reports the value of W-axis step counter.
RX	Reports the value of X-axis step counter.
RY	Reports the value of Y-axis step counter.
RZ	Reports the value of Z-axis step counter.
SETBIT <i>value</i>	Sets the discrete output specified by <i>value</i> to high state.

Host Controlled Operation

In this mode the host will send a series of ASCII commands to the controller over the RS-232 serial port. The controller process to the incoming commands and responses with the proper messages.

Programming Example in Visual BASIC

The following example sets the linear acceleration to 500,000 Steps / sec² , step rate at 100 KHz, and the distance to travel equal to 200,000 steps. Then the controller is commanded to make an absolute move on the X Axis.

```
Private Sub Command1_Click()
```

```
    'Function Prototype
```

```
    Declare Function SioPuts Lib "WSC32.DLL" (ByVal Port As Long, ByVal Buffer As String,
    ByVal Size As Long) As Long
```

```
    Dim Code As Long
```

```
    Dim StringToBeTransmtd As String
```

```
    ' Set the linear acceleration of X Axis to 500,000 steps / sec / sec
```

```
    StringToBeTransmtd = "accx 500000" + vbCr
```

```
    Code = SioPuts(ThePort, StringToBeTransmtd, Len(StringToBeTransmtd))
```

```
    ' Set the linear velocity of X Axis to 100,000 steps / sec
```

```
    StringToBeTransmtd = "velx 100000" + vbCr
```

```
    Code = SioPuts(ThePort, StringToBeTransmtd, Len(StringToBeTransmtd))
```

```
    ' Set the position to move of X Axis to 200,000 steps
```

```
    StringToBeTransmtd = "posx 200000" + vbCr
```

```
    Code = SioPuts(ThePort, StringToBeTransmtd, Len(StringToBeTransmtd))
```

```
    'Command the X Axis of the controller to make an absolute move
```

```
    StringToBeTransmtd = "movax" + vbCr
```

```
    Code = SioPuts(ThePort, StringToBeTransmtd, Len(StringToBeTransmtd))
```

```
End Sub
```

Programming Example in 'C'

The following example sets the acceleration at 10 Million Steps / sec² , step rate at 100 KHz, and the distance to travel equal to 100,000 steps. Then the controller is commanded to make an absolute move.

```
void send_command(void)
```

```
{
```

```
    char StringToBeTransmtd[80];
```

```
    // Set the linear acceleration of X Axis to 500,000 steps / sec / sec
```

```
    strcpy(StringToBeTransmtd,"accx 500000\n");
```

```
    SioPuts(Port,StringToBeTransmtd,strlen(StringToBeTransmtd));
```

```

// Set the linear velocity of X Axis to 100,000 steps / sec
strcpy(StringToBeTransmtd,"velx 100000\n");
SioPuts(Port,StringToBeTransmtd,strlen(StringToBeTransmtd));

// Set the position to move of X Axis to 200,000 steps
strcpy(StringToBeTransmtd,"posx 200000\n");
SioPuts(Port,StringToBeTransmtd,strlen(StringToBeTransmtd));
// 'Command the X Axis of the controller to make an absolute move
strcpy(StringToBeTransmtd,"movax\n");
SioPuts(Port,StringToBeTransmtd,strlen(StringToBeTransmtd));

}

```

The following is the information that you need to establish communication with OES line of controllers.

- 1) The baud rate is 19.2 K, 8-bit, no parity, one stop bit.
- 2) ASCII characters should be terminated with CR or LF.
- 3) After sending each packet of data to the OES' controller, sufficient time should be given to the controller to process it, usually 100msec.

To receive characters, a buffer is setup and all the incoming characters are stored in it until they are fetched by the application.

The following commands the X-axis to stop, and checks the receiving buffer for "X Stopping" string to make sure the X-axis has received and performed theCommand.

```
Private Sub cmdXPlus_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
Dim S As String
Dim Start As Single
Dim Counter As Integer
Dim Where As Long
```

```
StringToBeTransmtd = "stopx" & vbCr
```

```
Do While (Counter < NoofRetries And Where = 0)
Code = SioPuts(ThePort, StringToBeTransmtd, Len(StringToBeTransmtd))
Start = Timer()
Do While Timer() < Start + DelayTime
DoEvents ' Yield to other processes.
Loop
Where = InStr(strRcvdMsg, "X Stopping")
Counter = Counter + 1
Loop
```

```
If (Counter >= NoofRetries) Then
```

```
    If MsgBox("The controller is not responding! Turn the controller's power off.", vbOKOnly, "Error46")
```

```
= vbOK Then Exit Sub
```

```
End If
```

```
End Sub
```

OES uses a serial communication package from www.marshallsoft.com.

www.OESIncorp.com

Copyright © 2002 - 2014 Optimal Engineering Systems, Inc.

accw

Syntax

accw value

Function

Sets the linear acceleration of the *W*-axis to *value* .

This value is used to accelerate and decelerate the motor. The acceleration may not be modified while moving. The unit of *value* is in steps per sec² .

Range of Value

40,000 - 40,000,000

Controller Returns

None

Mode

Command

accx

Syntax

accx value

Function

Sets the linear acceleration of the *X*-axis to *value* .

This value is used to accelerate and decelerate the motor. The acceleration may not be modified while moving. The unit of *value* is in steps per sec² .

Range of Value

40,000 - 40,000,000

Controller Returns

None

Mode

Command

accy

Syntax

accy value

Function

Sets the linear acceleration of the Y-axis to *value* .

This value is used to accelerate and decelerate the motor. The acceleration may not be modified while moving. The unit of *value* is in steps per sec² .

Range of Value

40,000 - 40,000,000

Controller Returns

None

Mode

Command

accz

Syntax

accz value

Function

Sets the linear acceleration of the Z-axis to *value* .

This value is used to accelerate and decelerate the motor. The acceleration may not be modified while moving. The unit of *value* is in steps per sec² .

Range of Value

40,000 - 40,000,000

Controller Returns

None

Mode

Command

www.OESIncorp.com

Copyright © 2002 - 2014 Optimal Engineering Systems, Inc.

homew

Syntax

homew

Function

Instructs a home searching sequence on the W-axis.

The homing sequence is as follows:

The motor will start to move. The sign of the velocity determines the direction of the move. Once it hits the Home switch, it decelerates to stop at the same rate as acceleration.

Then the motor will move in reverse direction, at the slowest speed, until it senses the Home switch changes state then stops.

A normally closed or normally low switch should be connected to the Home pin. Please consult the Hardware Reference Manual for the location of the Home pin.

Controller Returns

W Homing

After completion of the sequence

Done W

Mode

Command

homex

Syntax

homex

Function

Instructs a home searching sequence on the X-axis.

The homing sequence is as follows:

The motor will start to move. The sign of the velocity determines the direction of the move. Once it hits the Home switch, it decelerates to stop at the same rate as acceleration.

Then the motor will move in reverse direction, at the slowest speed, until it senses the Home switch changes state then stops.

A normally closed or normally low switch should be connected to the Home pin. Please consult the Hardware Reference Manual for the location of the Home pin.

Controller Returns

X Homing

After completion of the sequence

Done X

Mode

Command

homey

Syntax

homey

Function

Instructs a home searching sequence of the Y-axis.

The homing sequence is as follows:

The motor will start to move. The sign of the velocity determines the direction of the move. Once it hits the Home switch, it decelerates to stop at the same rate as acceleration.

Then the motor will move in reverse direction, at the slowest speed, until it senses the Home switch changes state then stops.

A normally closed or normally low switch should be connected to the Home pin. Please consult the Hardware Reference Manual for the location of the Home pin.

Controller Returns

Y Homing

After completion of the sequence

Done Y

Mode

Command

homez

Syntax

homez

Function

Instructs a home searching sequence of the Z-axis.

The homing sequence is as follows:

The motor will start to move. The sign of the velocity determines the direction of the move. Once it hits the Home switch, it decelerates to stop at the same rate as acceleration.

Then the motor will move in reverse direction, at the slowest speed, until it senses the Home switch changes state then stops.

A normally closed or normally low switch should be connected to the Home pin. Please consult the Hardware Reference Manual for the location of the Home pin.

Controller Returns

Z Homing

After completion of the sequence

Done Z

Mode

Command

jogw

Syntax

jogw

Function

Instructs jogging mode of the W-axis.

Controller Returns

W Jogging
After stopping
Done W

Mode

Command

jogx

Syntax

jogx

Function

Instructs jogging mode of the X-axis.

Controller Returns

X Jogging
After stopping
Done X

Mode

Command

jogy

Syntax

jogy

Function

Instructs jogging mode of the Y-axis.

Controller Returns

Y Jogging
After stopping
Done Y

Mode

Command

jogz

Syntax

jogz

Function

Instructs jogging mode of the Z-axis.

Controller Returns

Z Jogging
After stopping
Done Z

Mode

Command

movall

Syntax

movall

Function

Begins an absolute move on all axes.

Controller Returns

W Abs. Move (*if applicable*)

X Abs. Move

Y Abs. Move

Z abs. Move (*if applicable*)

Mode

Command

movaw

Syntax

movaw

Function

Begins an absolute move on the W-axis.

Controller Returns

W Abs. Move

Mode

Command

movax

Syntax

movax

Function

Begins an absolute move on the X-axis.

Controller Returns

X Abs. Move

Mode

Command

movay

Syntax

movay

Function

Begins an absolute move on the Y-axis.

Controller Returns

Y Abs. Move

Mode

Command

movaz

Syntax

movaz

Function

Begins an absolute move on the Z-axis.

Controller Returns

Z Abs. Move

Mode

Command

www.OESIncorp.com

Copyright © 2002 - 2014 Optimal Engineering Systems, Inc.

movrall

Syntax

movrall

Function

Begins a relative move on all axes.

Controller Returns

W Rel. Move (*if applicable*)

X Rel. Move

Y Rel. Move

Z Rel. Move (*if applicable*)

Mode

Command

movrw

Syntax

movrw

Function

Begins a relative move on the W-axis.

Controller Returns

W Rel. Move

Mode

Command

movrx

Syntax

movrx

Function

Begins a relative move on the X-axis.

Controller Returns

X Rel. Move

Mode

Command

movry

Syntax

movry

Function

Begins a relative move on the Y-axis.

Controller Returns

Y Rel. Move

Mode

Command

movrz

Syntax

movrz

Function

Begins a relative move on the Z-axis.

Controller Returns

Z Rel. Move

Mode

Command

www.OESIncorp.com

Copyright © 2002 - 2014 Optimal Engineering Systems, Inc.

posw

Syntax

posw *value*

Function

Sets the distance to move of the W-axis to *value* .

Range of Value

-2147483647 to +2147483647

Controller Returns

None

Mode

Command

posx

Syntax

posx *value*

Function

Sets the distance to move of the X-axis to *value* .

Range of Value

-2147483647 to +2147483647

Controller Returns

None

Mode

Command

posy

Syntax

posy value

Function

Sets the distance to move of the Y-axis to *value* .

Range of Value

-2147483647 to +2147483647

Controller Returns

None

Mode

Command

posz

Syntax

posz value

Function

Sets the distance to move of the Z-axis to *value* .

Range of Value

-2147483647 to +2147483647

Controller Returns

None

Mode

Command

stopall

Syntax

stopall

Function

Stops the motion of all axes.

Controller Returns

W Stopping (if applicable)

X Stopping

Y Stopping

Z Stopping (if applicable)

Mode

Command

stopw

Syntax

stopw

Function

Stops the motion of the W-axis.

Controller Returns

W Stopping

Mode

Command

stopx

Syntax

stopx

Function

Stops the motion of the X-axis.

Controller Returns

X Stopping

Mode

Command

stopy

Syntax

stopy

Function

Stops the motion of the Y-axis.

Controller Returns

Y Stopping

Mode

Command

stopz

Syntax

stopz

Function

Stops the motion of the Z-axis.

Controller Returns

Z Stopping

Mode

Command

www.OESIncorp.com

Copyright © 2002 - 2014 Optimal Engineering Systems, Inc.

velw

Syntax

velw *value*

Function

Sets the maximum step rate of the W-axis to *value* .

This value sets the slew speed of the motor. The speed may not be modified while moving. The unit of *value* is in steps / Sec.

Range of Value

200 – 200,000

Controller Returns

None

Mode

Command

velx

Syntax

velx *value*

Function

Sets the maximum step rate of the X-axis to *value* .

This value sets the slew speed of the motor. The speed may not be modified while moving. The unit of *value* is in steps / Sec.

Range of Value

200 – 200,000

Controller Returns

None

Mode

Command

vely

Syntax

vely *value*

Function

Sets the maximum step rate of the Y-axis to *value* .

This value sets the slew speed of the motor. The speed may not be modified while moving. The unit of *value* is in steps / Sec.

Range of Value

200 – 200,000

Controller Returns

None

Mode

Command

velz

Syntax

velz *value*

Function

Sets the maximum step rate of the Z-axis to *value* .

This value sets the slew speed of the motor. The speed may not be modified while moving. The unit of *value* is in steps / Sec.

Range of Value

200 – 200,000

Controller Returns

None

Mode

Command

arc

Syntax

arc X-axis-end, Y-axis-end, X-axis-center, Y-axis-center, Direction

Function

Generates a 2-dimensional circular arc segment.

Controller Returns

None

Mode

Command

done?

Syntax

done?

Function

Halts the execution of the program until the coordinated motion is completed.

Controller Returns

None

Mode

Run

line

Syntax

line X-axis-end, Y-axis-end, Z-axis-end, W-axis-end

Function

Makes a coordinated linear motion

Controller Returns

None

Mode

Command

vaccel

Syntax

vaccel *value*

Function

Sets the vector acceleration of the coordinated move to *value* .

The *value* may not be modified while moving. The unit of *value* is in steps per sec².

Range of Value

40,000 - 40,000,000

Mode

Command

vvel

Syntax

vvel *value*

Function

Sets the vector velocity of the coordinated move to *value* .

The *value* may not be modified while moving. The unit of *value* is in steps / Sec.

Range of Value

200 – 200,000

Controller Returns

None

Mode

Command

eposw

Syntax

eposw

Function

Equates the current position of the W-axis to the value of the corresponding step counter.

Controller Returns

None

Mode

Command

eposx

Syntax

eposx

Function

Equates the current position of the X-axis to the value of the corresponding step counter.

Controller Returns

None

Mode

Command

eposy

Syntax

eposy

Function

Equates the current position of the Y-axis to the value of the corresponding step counter.

Controller Returns

None

Mode

Command

eposz

Syntax

eposz

Function

Equates the current position of the Z-axis to the value of the corresponding step counter.

Controller Returns

None

Mode

Command

joff

Syntax

joff

Function

Disables the joystick and trackball operation.

Range of Value

None

Controller Returns

Joystick is off.

Mode

Command

jon

Syntax

jon

Function

Enables the joystick and trackball operation.

Controller Returns

Joystick is on.

Mode

Command

moffw

Syntax

moffw

Function

Turns the motor driver of the W-axis off.

Controller Returns

None

Mode

Command

moffx

Syntax

moffx

Function

Turns the motor driver of the X-axis off.

Controller Returns

None

Mode

Command

moffy

Syntax

moffy

Function

Turns the motor driver of the Y-axis off.

Controller Returns

None

Mode

Command

moffz

Syntax

moffz

Function

Turns the motor driver of the Z-axis off.

Controller Returns

None

Mode

Command

www.OESIncorp.com

Copyright © 2002 - 2014 Optimal Engineering Systems, Inc.

monw

Syntax

monw

Function

Turns the motor driver of the W-axis on.

Controller Returns

None

Mode

Command

monx

Syntax

monx

Function

Turns the motor driver of the X-axis on.

Controller Returns

None

Mode

Command

mony

Syntax

mony

Function

Turns the motor driver of the Y-axis on.

Controller Returns

None

Mode

Command

monz

Syntax

monz

Function

Turns the motor driver of the Z-axis on.

Controller Returns

Z Motor on

Mode

Command

msgoff

Syntax

msgoff

Function

Turns off the feedback messages sent from the controller.

Controller Returns

None

Mode

Command

msgon

Syntax

msgon

Function

Turns on the feedback messages sent from the controller.

Controller Returns

None

Mode

Command

sposw

Syntax

sposw *value*

Function

Sets the current position of the W-axis to *value* .

Range of Value

-2147483647 to +2147483647

Controller Returns

None

Mode

Command

sposx

Syntax

sposx *value*

Function

Sets the current position of the X-axis to *value* .

Range of Value

-2147483647 to +2147483647

Controller Returns

None

Mode

Command

sposy

Syntax

sposy *value*

Function

Sets the current position of the Y-axis to *value* .

Range of Value

-2147483647 to +2147483647

Controller Returns

None

Mode

Command

sposz

Syntax

sposz *value*

Function

Sets the current position of the Z-axis to *value* .

Range of Value

-2147483647 to +2147483647

Controller Returns

None

Mode

Command

squadw

Syntax

squadw *value*

Function

Sets the W-axis quadrature decoder to *value*.

The optional quadrature decoder card is required.

Range of Value

-2147483647 to +2147483647

Controller Returns

None

Mode

Command

squadx

Syntax

squadx *value*

Function

Sets the X-axis quadrature decoder to *value*.

The optional quadrature decoder card is required.

Range of Value

-2147483647 to +2147483647

Controller Returns

None

Mode

Command

squady

Syntax

squady *value*

Function

Sets the Y-axis quadrature decoder to *value*.

The optional quadrature decoder card is required.

Range of Value

-2147483647 to +2147483647

Controller Returns

None

Mode

Command

squadz

Syntax

squadz *value*

Function

Sets the Z-axis quadrature decoder to *value*.

The optional quadrature decoder card is required.

Range of Value

-2147483647 to +2147483647

Controller Returns

None

Mode

Command

www.OESIncorp.com

Copyright © 2002 - 2014 Optimal Engineering Systems, Inc.

clrbit

Syntax

clrbit *value*

Function

Resets the discrete output specified by *value* . Refer to the hardware reference manual for the location of each pin.

Range of Value

1 to 16

Controller Returns

None

Mode

Command

setbit

Syntax

setbit *value*

Function

Sets the discrete output specified by *value* . Refer to hardware reference manual for the location of each pin.

Range of Value

1 – 16

Controller Returns

None

in

Syntax

in

Function

Reads a word from the input ports and reports it via the serial port. Refer to the hardware reference manual for the location of each pin.

Range of Value

0 to 65535

Controller Returns

None

Mode

Command

out

Syntax

out *value*

Function

Writes the *value* to the output ports. Refer to the hardware reference manual for the location of each pin.

Range of Value

0 to 65535

Controller Returns

None

Mode

Command

www.OESIncorp.com

Copyright © 2002 - 2014 Optimal Engineering Systems, Inc.

rquadw

Syntax

rquadw

Function

The controller sends the quadrature decoder value to the serial port.

The optional quadrature decoder card is required.

Range of Value

-2147483647 through +2147483647

Controller Returns

The value if no argument is specified.

Mode

Command

rquadx

Syntax

rquadx

Function

The controller sends the quadrature decoder value to the serial port.

The optional quadrature decoder card is required.

Range of Value

-2147483647 through +2147483647

Controller Returns

The value if no argument is specified.

Mode

Command

rquady

Syntax

rquady

Function

The controller sends the quadrature decoder value to the serial port.

The optional quadrature decoder card is required.

Range of Value

-2147483647 through +2147483647

Controller Returns

The value if no argument is specified.

Mode

Command

rquadz

Syntax

rquadz

Function

The controller sends the quadrature decoder value to the serial port.

The optional quadrature decoder card is required.

Range of Value

-2147483647 through +2147483647

Controller Returns

the value if no argument is specified.

Mode

Command

rstsw

Syntax

rstsw

Function

Reports the status of W-axis.

Bit 0 is 1 if in MOVE mode else 0

Bit 1 is 0 if Positive Limit Switch is hit else 1

Bit 2 is 0 if Negative Limit Switch is hit else 1

Bit 3 is 0 if Home Limit Switch is hit else 1

bit 4 - bit 31 Spare

Controller Returns

The *value* if no argument is specified.

Mode

Command

rstsx

Syntax

rstsx

Function

Reports the status of X-axis.

Bit 0 is 1 if in MOVE mode else 0

Bit 1 is 0 if Positive Limit Switch is hit else 1

Bit 2 is 0 if Negative Limit Switch is hit else 1

Bit 3 is 0 if Home Limit Switch is hit else 1

Bit 4 is spare

Bit 5 is spare

Bit 6 is spare

Bit 7 is spare

Bit 8 is 1 if Joystick Lower Left Key Is Pressed.

Bit 9 is 1 if Joystick Middle Right Key Is Pressed.

Bit 10 is 1 if Joystick Upper Right Key Is Pressed.

Bit 11 is 1 if Joystick Lower Right Key Is Pressed.

Bit 12 is spare.

Bit 13 is spare.

Bit 14 is spare.
Bit 15 is spare.
Bit 16 - bit 23 = X-axis analog value.
Bit 24 - bit 31 = Spare

Range of Value

-2147483648 to +2147483647

Controller Returns

The value if no argument is specified.

Mode

Command

rstsy

Syntax

rstsy

Function

Reports the status of Y-axis.

Bit 0 is 1 if in MOVE mode else 0
Bit 1 is 0 if Positive Limit Switch is hit else 1
Bit 2 is 0 if Negative Limit Switch is hit else 1
Bit 3 is 0 if Home Limit Switch is hit else 1
Bit 4 - bit 15 = Spare
Bit 16 - bit 23 = Y-axis analog value.
Bit 24 - bit 31 = Spare

Range of Value

-2147483648 to +2147483647

Controller Returns

The value if no argument is specified.

Mode

Command

rstz

Syntax

rstz

Function

Reports the status of Z-axis.

Bit 0 is 1 if in MOVE mode else 0

Bit 1 is 0 if Positive Limit Switch is hit else 1

Bit 2 is 0 if Negative Limit Switch is hit else 1

Bit 3 is 0 if Home Limit Switch is hit else 1

Bit 4 - bit 15 = Spare

Bit 16 - bit 23 = Z-axis analog value.

Bit 24 - bit 31 = Spare.

Range of Value

0 to +16777215

Controller Returns

The value if no argument is specified.

Mode

Command

rw

Syntax

rw

Function

Reports the value of W-axis step counter prefixed by W.

Controller Returns

The value of W-axis step counter prefixed by W.

Mode

Command

rx

Syntax

rx

Function

Reports the value of X-axis step counter prefixed by X.

Controller Returns

The value of X-axis step counter prefixed by X.

Mode

Command

ry

Syntax

ry

Function

Reports the value of Y-axis step counter prefixed by Y.

Controller Returns

The value of Y-axis step counter prefixed by Y.

Mode

Command

rz

Syntax

rw

Function

Reports the value of Z-axis step counter prefixed by Z.

Controller Returns

The value of Z-axis step counter prefixed by Z.

Mode

Command