DSPMCv2 – pn7762S, 7762M

Ethernet Motion Controller
Data Acquisition System
PID Controller

User Guide

Document Revision 3.01
(Updated August 19, 2010)

Phoenix, AZ USA

For more information please visit the product web page:
www.vitalsystem.com/dspmc

CONTENTS

1. Overview

The DSPMC is an Ethernet based controller for motion control, data acquisition, and general PID control system applications. Utilizing the latest DSP technology, the DSPMC offers a comprehensive set of features for your demanding applications.

DSPMC controller can be applied in a variety of applications involving PC based Motion Control, Storage and Retrieval Systems and CNC Milling / Lathe Machines. Equipped with a rich set of hardware interfaces, it can also be used for wide variety of applications involving PID control, e.g., speed, oven temperature control and so on.

Following in the key features of DSPMC v2:

- **8 Channels Servo Drive Analog Outputs, Range ±10V, 16-Bit Resolution**
- **8 Differential Quadrature Encoder Inputs. 32-Bit Resolution**
- **4 Mhz Max Encoder frequency. Encoder resolution multiplied by 4 thru Hardware.**
- **8 Step and Direction Channels. Upto 2MHz Step Frequencies (Currently not available)**
- **8 Channel Analog Inputs, Range ±10Volts, 16-bit Resolution (7762M Only)**
- **96 Digital I/O (64 Inputs & 32 Outputs)**
- **Ethernet 100Mb connectivity using TCP/IP interface.**
- **Wide input power range 10-40VDC**
- **Simple UDP Socket Programming Interface.**
- **Visual Studio 2008 .Net Managed Library for C#, C++, and VB.Net Software Developers.**
- **Standalone Operation by programming the unit with BASIC programming language.**

DSPMC board comes with GUI software tools to test the hardware, setup PID controller, run motion control commands, and upgrade new firmware. Following gives a brief description of the software tool set:

- **DSPMC *AxisWorks*** – A GUI based software tool to exercise the user’s hardware installation. After configuring the I/Os and PID settings, this program can be used to execute motion related commands to verify the installation is setup properly.

- **DSPMC Firmware Upgrade** – A GUI based software tool to re-flash (burn) the firmware stored on the DSPMC board. New versions of this program and firmware can be obtained from the factory.

- **Mach3 Plugin** – Driver Software with Integrated ***AxisWorks*** for Mach3 CNC software. Users don’t need to use ***AxisWorks*** with Mach3 as this software is builtin the Mach3 plugin.

- **Windows .Net Library** – Software Library for custom pc software development.

Extremely Important Reminder

When operating machines, take extreme precautions. The machines can have enormous power even with a small motor. Never come inside a machine path while powered. Operating machines without necessary precautions can result in lost of limbs or even death.

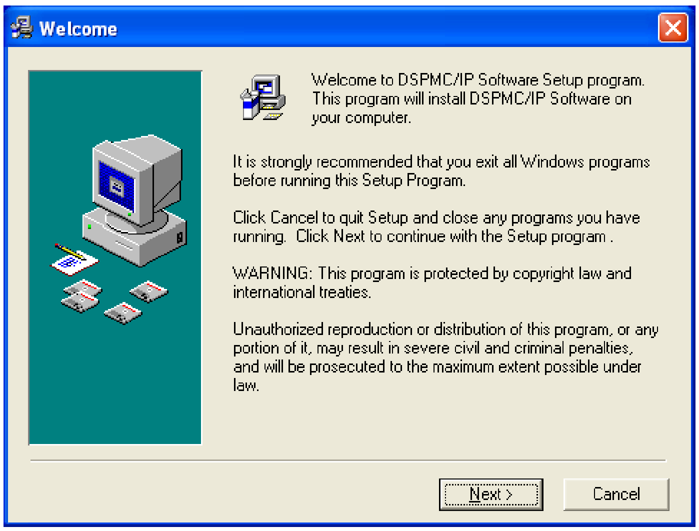
2. Software setup

2.1 Installing the software tools

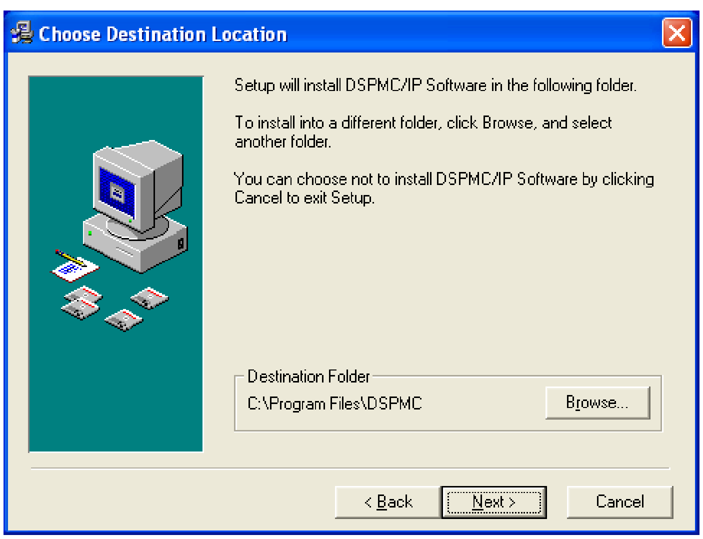
Following are the steps to download and install the setup program.

Download the setup software from http://www.vitalsystem.com/dspmc, click on the setup software link, and save the file to an appropriate directory. Unzip this file and double click on the setup.exe file.

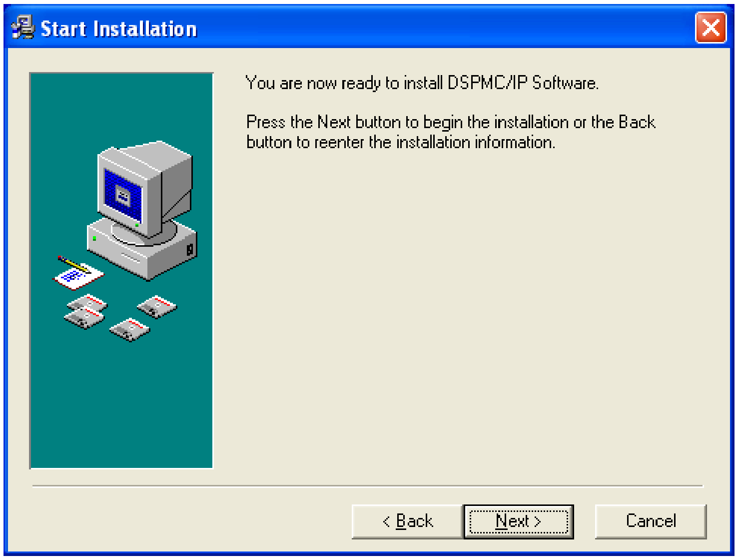
The following Welcome screen appears. Click on next button.

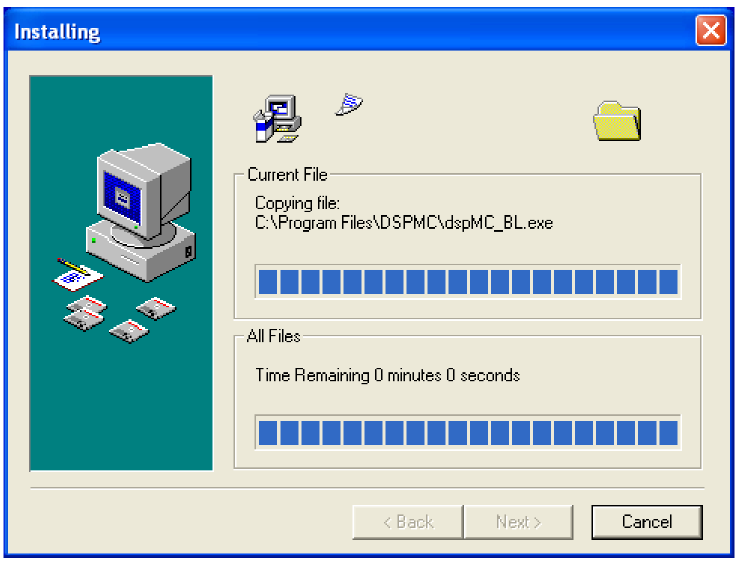


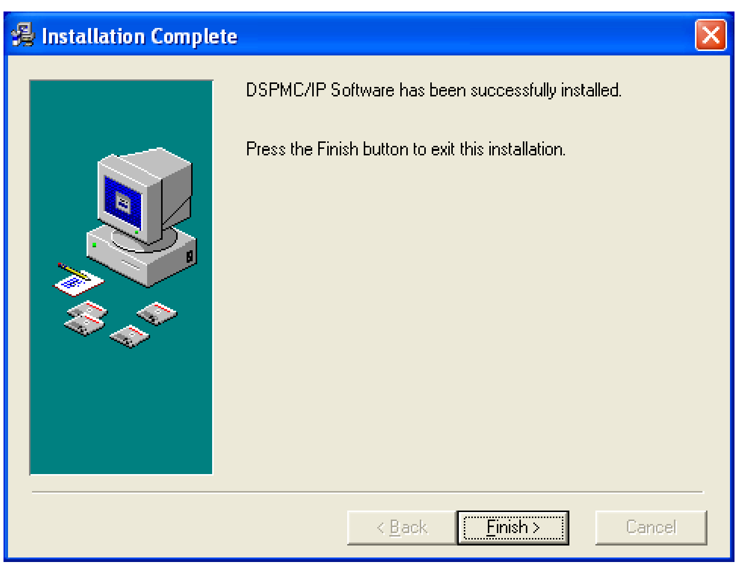
Select the destination of the programs to be installed.



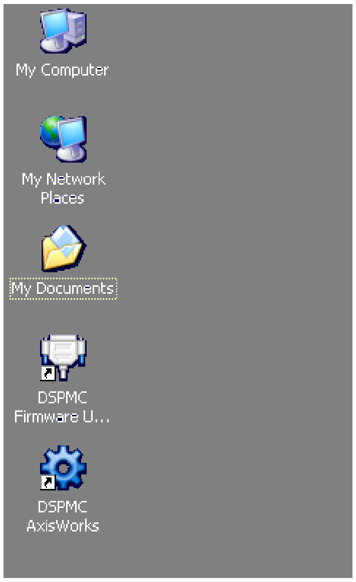
Click on the Next Button to start the installation.


Start Installation
You are now ready to install DSPMC/IP Software.
Press the Next button to begin the installation or the Back button to reenter the installation information.
< Back
Next >
Cancel


Installing
Current File
Copying file:
C:\Program Files\DSPMC\dspMC_BL.exe
All Files
Time Remaining 0 minutes 0 seconds
< Back
Next >
Cancel


Installation Complete
DSPMC/IP Software has been successfully installed.
Press the Finish button to exit this installation.
< Back
Finish >
Cancel

The installation is complete if you see the above window. Click on ‘Finish’ to exit the setup program. This installation creates two shortcut icons on the desktop, one for the AxisWorks testing program and the other for Firmware upgrade software.



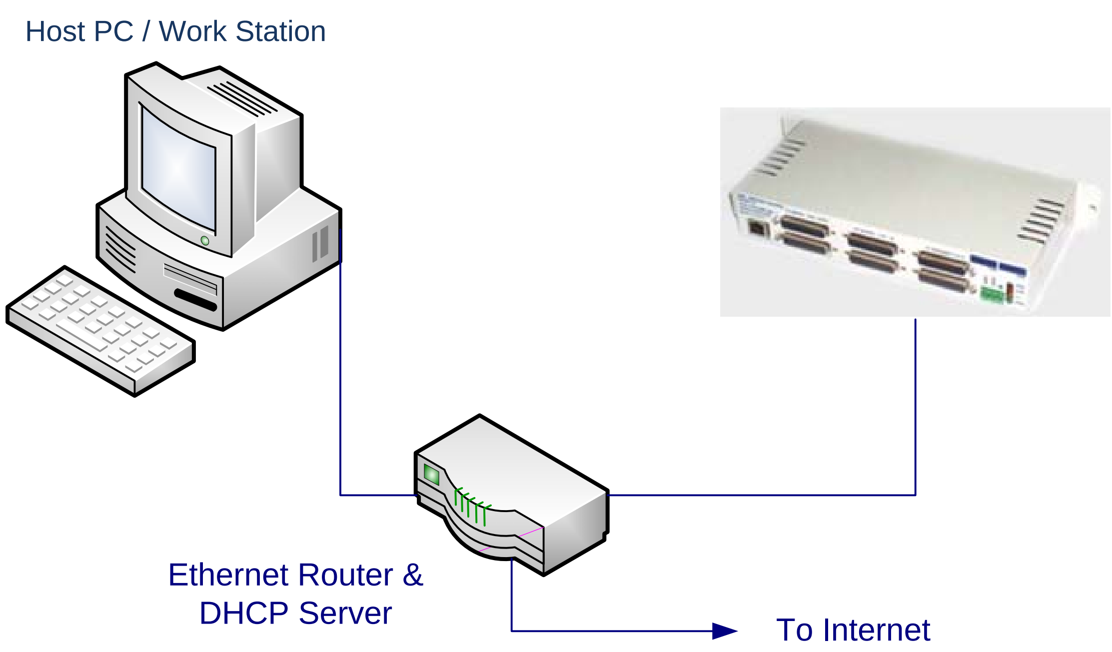
3. Network Connection Setup

You can connect the DSPMC board directly to your PC or connect via an Ethernet switch or router. The DSPMC board can use the firmware pre-assigned IP address, ie, 192.168.0.50, or it can get a unique IP address from an external DHCP server on your network. In the latter case, the firmware pre-assigned IP address is ignored.

There are two ways to setup the IP addresses of your PC and the DSPMC board.

1. Using a Router with DHCP Server
2. Manually assigning an IP Address to your PC

3.1 Setup IP address using a Router with DHCP Server



The figure above shows a basic setup using a router on your network. Connect the Ethernet cable from the J1- Ethernet port of the DSPMC board to the DHCP server/Router. Connect another Ethernet cable from the DHCP Server/Router to the PC. The DHCP server dynamically assigns IP address both to the PC as well as to the DSPMC board, and therefore completes the network setup without requiring any intervention from the user.

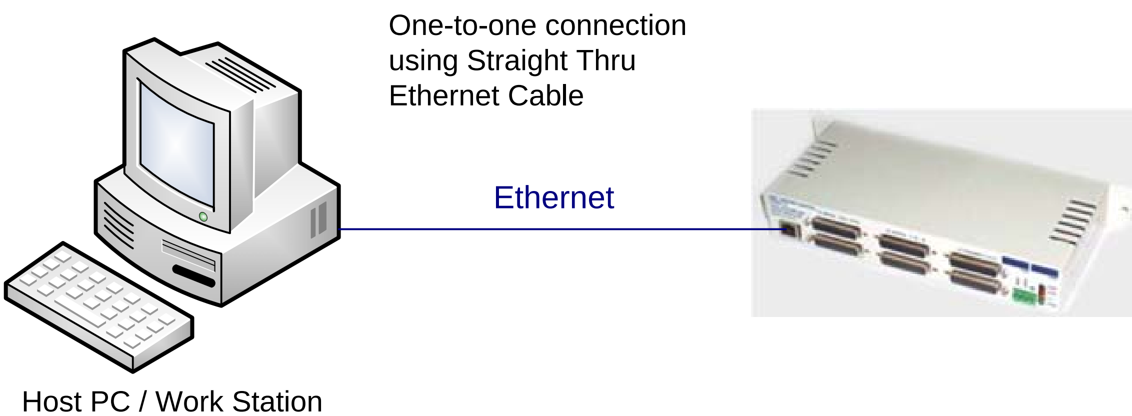
Note: If your network device does not support Auto-MDIX feature, a crossover Ethernet cable may be required.

3.2 Manually assigning an IP Address to the PC

With TCP/IP networking, the PC and the DSPMC, both, need their own unique IP address.

When connecting the PC directly to the DSPMC board, you will need to manually assign an IP address to your PC. The DSPMC board will use its firmware pre-assigned IP address, i.e., 192.168.0.50.

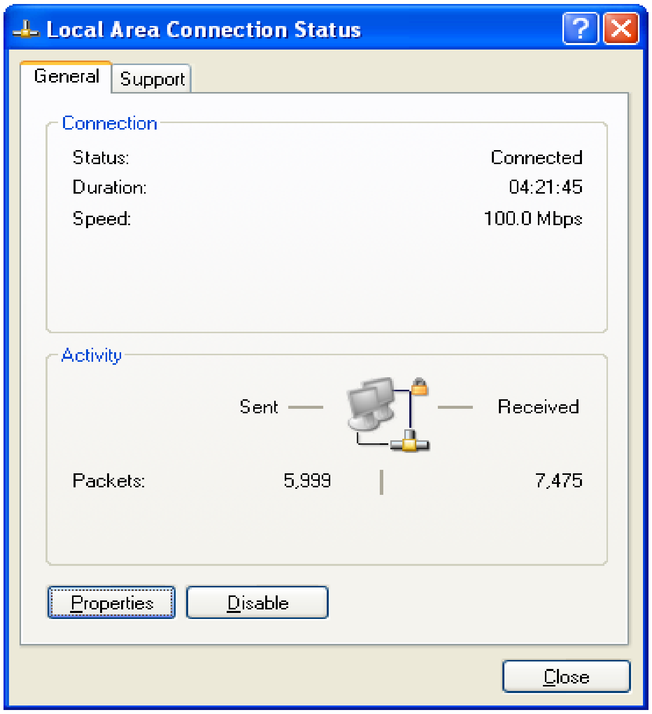
The Ethernet cable is connected from the J1-ethernet port of the DSPMC board to the PC as shown below:



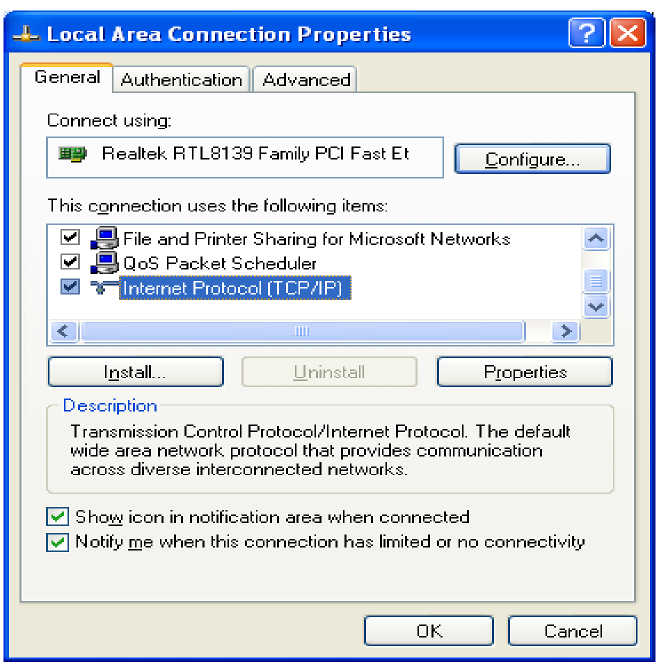
The PC IP Address can be configured manually in windows XP as follows. For other operating systems, please consult the respective user guides for changing the IP address.

1. Double click on the ‘My Network Places’ icon in Windows XP and open the ‘available network connections’.

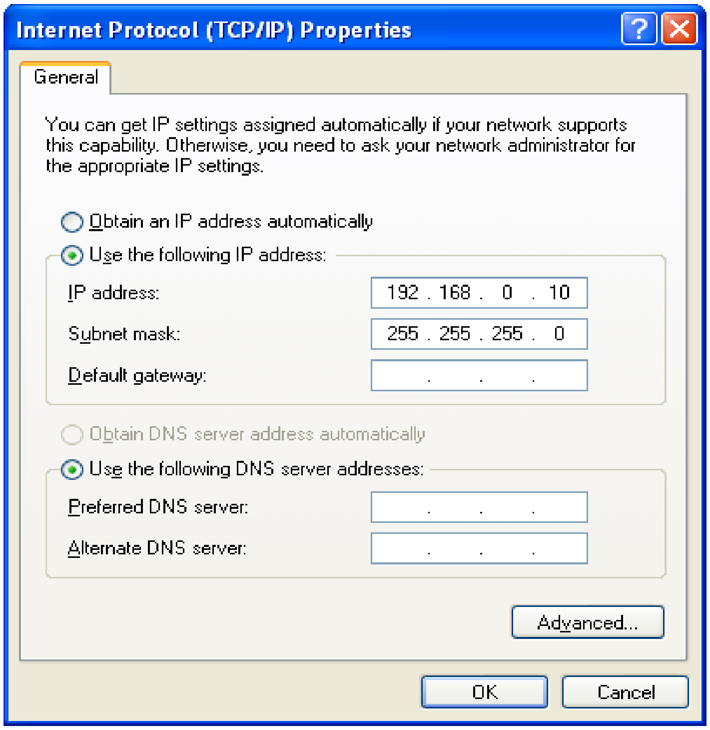
2. Double click on the corresponding LAN Connection over which the device will be setup. The following window appears.



3. Click on the Properties and select the Internet Protocol (TCP/IP) Connection in ‘General’ Tab

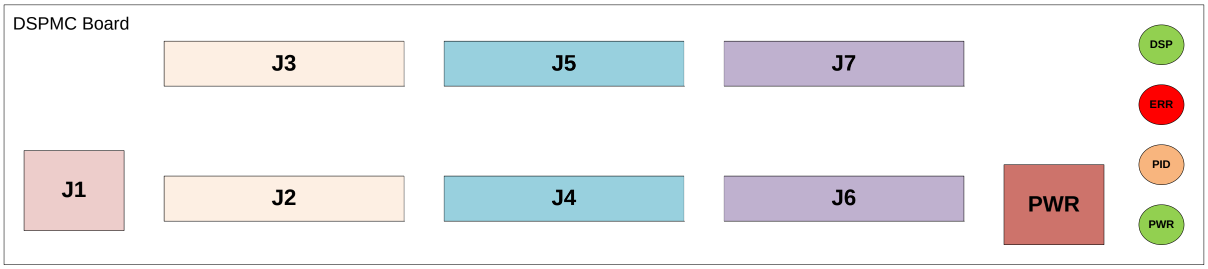


4. Click on the ‘Properties’ button and make the settings in your PC similar to the one shown in the figure below. After settings are done, click ‘OK’ button to finish the setup


Internet Protocol (TCP/IP) Properties
General
You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.
Obtain an IP address automatically
Use the following IP address:
IP address:
192 . 168 . 0 . 10
Subnet mask:
255 . 255 . 255 . 0
Default gateway:
Obtain DNS server address automatically
Use the following DNS server addresses:
Preferred DNS server:
Alternate DNS server:
Advanced...
OK
Cancel

4. Hardware Interface description

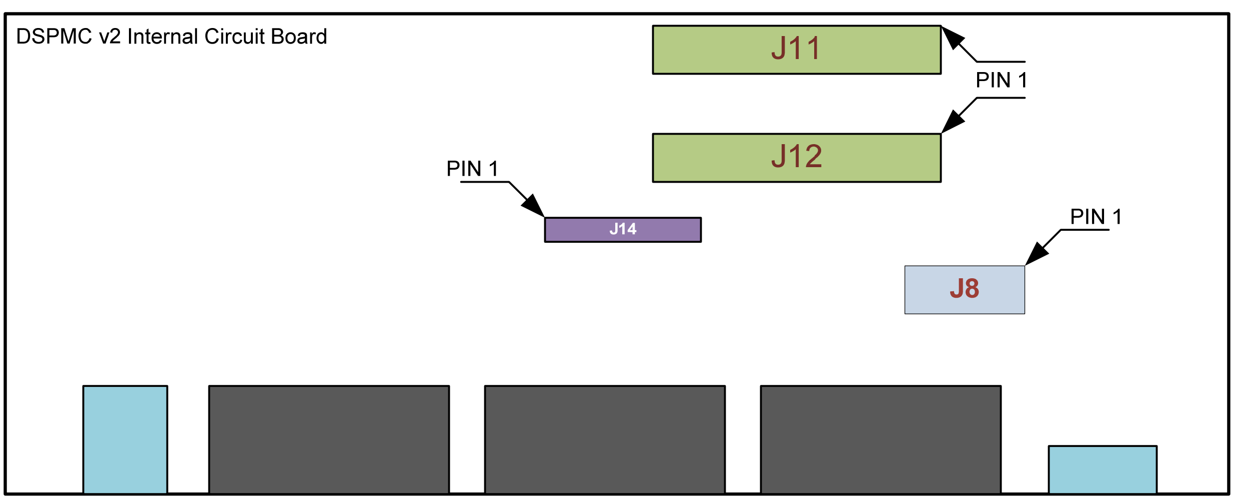
The DSPMC board has several interface ports and indicator LEDs. Figure below shows front side view of the DSPMC board with interface ports and other components:



J1 – Ethernet port connected to PC
J2 – Provides analog input and DAC output
J3 – Stepper Motor outputs, Differential Encoder Input (Index 6).
J4 – Digital I/Os, provide 16 inputs (0 to 15) and 8 outputs (0 to 7)
J5 – Digital I/Os, provide 16 inputs (16 to 31) and 8 outputs (8 to 15)
J6 – Differential Encoder inputs (Index 0, 1 and 2).
J7 – Differential Encoder inputs (Index 3, 4 and 5)
PWR – Power Connector

PWR LED – Green colored LED for Power indication; it glows steadily when Power is on
PID LED – Orange colored LED for PIDs in-control; it glows steadily when PID is armed
ERR LED – Red colored LED for error indication.
DSP LED – Green colored LED indicating DSP processor operation; blinks constantly during normal operation.

There are few connectors available on the internal circuit board, which are accessible by opening the top cover, or the ribbon cable slot:



J11 – Digital I/Os, provide 16 inputs (32 to 47) and 8 outputs (16 to 23)
J12 – Digital I/Os, provide 16 inputs (48 to 63) and 8 outputs (24 to 31)
J8 – Differential Encoder Channel Input (Index 7).
J14 – Stepper Step and Direction Output Channels 6 and 7 (**3.3Volts** signals)

4.1 Ethernet Port - J1

Connect to PC directly or via an Ethernet Hub or a switch. The DSPMC board supports both 10 MBit and 100 Mbit network speeds. TCP/IP network protocol in UDP mode is used for PC communications.

4.2 Analog I/O Port – J2

- **Analog Inputs** (Available on **7762-M** Models Only)
  - Input voltage Range: **-10 to +10 Volts**.
  - Input impedance: 10M Ohm.
  - Binary Resolution: 16 bits
  - Conversion Rate: up to 20KHz

- **Analog Outputs**
  - Analog Output range: +/-10 Volts.
  - Analog Output Resolution: 16 Bits
  - Maximum Output Current Per Output: 20mA

J2 Pin Assignments:

Pin#	Function	Pin#	Function
1	+12V, 100mA max	20	Analog Input 0
14	+5V, 500mA max	8	Analog Input 1
2	-12V, 50mA max	21	Analog Input 2
15	Analog Output 0	9	Analog Input 3
3	Analog Output 1	22	Ground (return)
16	Analog Output 2	10	Analog Input 4
4	Analog Output 3	23	Analog Input 5
17	Ground (return)	11	Analog Input 6
5	Analog Output 4	24	Analog Input 7
18	Analog Output 5	12	Ground (return)
6	Analog Output 6	25	Ground (return)
19	Analog Output 7	13	+5V, 500mA max
7	Ground (return)		

4.3 Stepper Outputs, Encoder Channel #6 on J3 and J14

J3 provides 6 channels Step and Direction Signals, and one differential encoder input. The Step/Dir output signals are 0...5V Range.

J3 Pin Assignments:

Pin#	Function	Pin#	Function
1	Step 0	20	Direction 5
14	Step 1	8	Encoder Channel 6 A+
2	Step 2	21	Encoder Channel 6 A-
15	Step 3	9	Encoder Channel 6 B+
3	Step 4	22	Encoder Channel 6 B-
16	Step 5	10	Encoder Channel 6 X+
4	+5V, 500mA	23	Encoder Channel 6 X-
17	Ground (return)	11	+5V 500mA
5	Direction 0	24	Ground (Return)
18	Direction 1	12	Reserved
6	Direction 2	25	Reserved
19	Direction 3	13	Ground (Return)
7	Direction 4		

J14 Pin Assignments: J14 is a 6 pin header, which is accessible by opening the top cover. It provides two extra channels of Step/Direction signals. These Stepper channels use the 3.3V standard. Stepper Channels 0..5 are available on J3 outside connector, and use 5 Volts standard.

Pin#	Function
1	**+3.3V, 100mA**
2	**Gnd (Return)**
3	**Step 6**
4	**Direction 6**
5	**Step 7**
6	**Direction 7**

4.4 Digital I/O Ports - J4, J5, J11, J12

On the DSPMC 7762 board, there are Four Digital I/O connectors, each providing sixteen inputs and eight outputs. Total, there are 64 digital inputs and 32 digital outputs.

These I/Os are not optically isolated. To get optical isolation, you can use the digital I/O breakout board ***pn 7535*** with DSPMC board. Please visit ***www.vitalsystem.com*** for more info on 7535. These boards connect directly to J4, J5, J11 and J12, and provide detachable screw terminals for easy wiring and maintenance.

The digital inputs are also used for emulated quadrature encoder for low speed applications, eg, MPG wheel for CNC control. For more details, see section 4.6 Single-Ended Encoder Inputs.

The DSPMC v2 board has the following electrical limits on its Digital I/O pins:

Digital Inputs	Digital Outputs
Input Voltage Range: 0 … 3.3 Volts High Level Threshold: 2.8 Volts Low Level Threshold: 0.8 Volts	Output Voltage Range: 0 … 3.3 Volts Max Output Current Per Output: 24mA

The Digital inputs and outputs on DSPMC-v2 (J4, J5, J11, J12, and J14) use the 3.3volts standard. The user should make sure that these I/O signals do not connect to a 5volts source. However, 5volts through a 4.7K or higher value resister can be connected to any input or output pin. A direct connection of these pins to 5volts (without a resister) will damage the unit.

The Vital Systems Opto-Isolated I/O boards 7535, and OPTO22 style modules e.g. G4ODC5 and G4IDC5, are compatible with DSPMC Digital I/O Ports.

J4 Pin Assignments:

Pin#	Function	Pin#	Function
1	Ground (Return)	20	Input 4
14	Output 0	8	Input 5
2	Output 1	21	Input 6
15	Output 2	9	Input 7
3	Output 3	22	Input 8
16	Output 4	10	Input 9
4	Output 5	23	Input 10
17	Output 6	11	Input 11
5	Output 7	24	Input 12
18	Input 0	12	Input 13
6	Input 1	25	Input 14
19	Input 2	13	Input 15
7	Input 3		

J5 Pin Assignments:

Pin #	Function	Pin#	Function
1	Ground (Return)	20	Input 20
14	Output 8	8	Input 21
2	Output 9	21	Input 22
15	Output 10	9	Input 23
3	Output 11	22	Input 24
16	Output 12	10	Input 25
4	Output 13	23	Input 26
17	Output 14	11	Input 27
5	Output 15	24	Input 28
18	Input 16	12	Input 29
6	Input 17	25	Input 30
19	Input 18	13	Input 31
7	Input 19		

J11 Pin Assignments:

Pin #	Function	Pin#	Function
1	Ground (Return)	20	Input 36
14	Output 16	8	Input 37
2	Output 17	21	Input 38
15	Output 18	9	Input 39
3	Output 19	22	Input 40
16	Output 20	10	Input 41
4	Output 21	23	Input 42
17	Output 22	11	Input 43
5	Output 23	24	Input 44
18	Input 32	12	Input 45
6	Input 33	25	Input 46
19	Input 34	13	Input 47
7	Input 35	26	+5V, 500mA

J12 Pin Assignments:

Pin #	Function	Pin#	Function
1	Ground (Return)	20	Input 52
14	Output 24	8	Input 53
2	Output 25	21	Input 54
15	Output 26	9	Input 55
3	Output 27	22	Input 56
16	Output 28	10	Input 57
4	Output 29	23	Input 58
17	Output 30	11	Input 59
5	Output 31	24	Input 60
18	Input 48	12	Input 61
6	Input 49	25	Input 62
19	Input 50	13	Input 63
7	Input 51	26	+5V, 500mA

4.5 Differential Quadrature Encoders on J6, J7, J3, and J8

J6 Pin Assignments:

Pin#	Function	Pin#	Function
1	Ch 0 A+	8	+5V 500mA
14	Ch 0 A-	21	Ground (Return)
2	Ch 0 B+	9	Ch 2 A+
15	Ch 0 B-	22	Ch 2 A-
3	Ch 0 Z+	10	Ch 2 B+
16	Ch 0 Z-	23	Ch 2 B-
4	+5V 500mA	11	Ch 2 Z+
17	Ground (Return)	24	Ch 2 Z-
5	Ch 1 A+	12	+5V 500mA
18	Ch 1 A-	25	Ground (Return)
6	Ch 1 B+	13	Reserved
19	Ch 1 B-		
7	Ch 1 Z+		
20	Ch 1 Z-		

J7 Pin Assignments:

Pin#	Function	Pin#	Function
1	Ch 3 A+	8	+5V 500mA
14	Ch 3 A-	21	Ground (Return)
2	Ch 3 B+	9	Ch 5 A+
15	Ch 3 B-	22	Ch 5 A-
3	Ch 3 Z+	10	Ch 5 B+
16	Ch 3 Z-	23	Ch 5 B-
4	+5V 500mA	11	Ch 5 Z+
17	Ground (Return)	24	Ch 5 Z-
5	Ch 4 A+	12	+5V 500mA
18	Ch 4 A-	25	Ground (Return)
6	Ch 4 B+	13	Reserved
19	Ch 4 B-		
7	Ch 4 Z+		
20	Ch 4 Z-		

J3 Pin Assignments:

Pin#	Function	Pin#	Function
1	Step 0	20	Direction 5
14	Step 1	**8**	**Encoder Channel 6 A+**
2	Step 2	**21**	**Encoder Channel 6 A-**
15	Step 3	**9**	**Encoder Channel 6 B+**
3	Step 4	**22**	**Encoder Channel 6 B-**
16	Step 5	**10**	**Encoder Channel 6 X+**
4	+5V, 500mA	**23**	**Encoder Channel 6 X-**
17	Ground (return)	**11**	**+5V 500mA**
5	Direction 0	**24**	**Ground (Return)**
18	Direction 1	12	Reserved
6	Direction 2	25	Reserved
19	Direction 3	13	Ground (Return)
7	Direction 4		

J8 Pin Assignments: J8 is a 10 pin header, which is accessible by opening the top cover.

Pin#	Function
4	**Encoder Channel 7 A+**
3	**Encoder Channel 7 A-**
6	**Encoder Channel 7 B+**
5	**Encoder Channel 7 B-**
8	**Encoder Channel 7 X+**
7	**Encoder Channel 7 X-**
2	**+5V 500mA**
1	**Ground (Return)**
9	Unused
10	Unused

4.6 Single-Ended Encoder Inputs

In addition to dedicated hardware encoder inputs, DSPMC board also provide three Single Ended encoder inputs in Digital Input J4 and J5 connector. These are simulated encoder inputs, therefore called, *SoftEncoder*, and are used for low speed applications like MPG. The following table lists the Digital I/O pins assigned to *SoftEncoders*. (Requires Firmware Rev 63 or newer):

SoftEncoder 0 : A+ On J4 Pin 24,
B+ On J4 Pin 12

SoftEncoder 1 : A+ On J4 Pin 25
B+ On J4 Pin 13

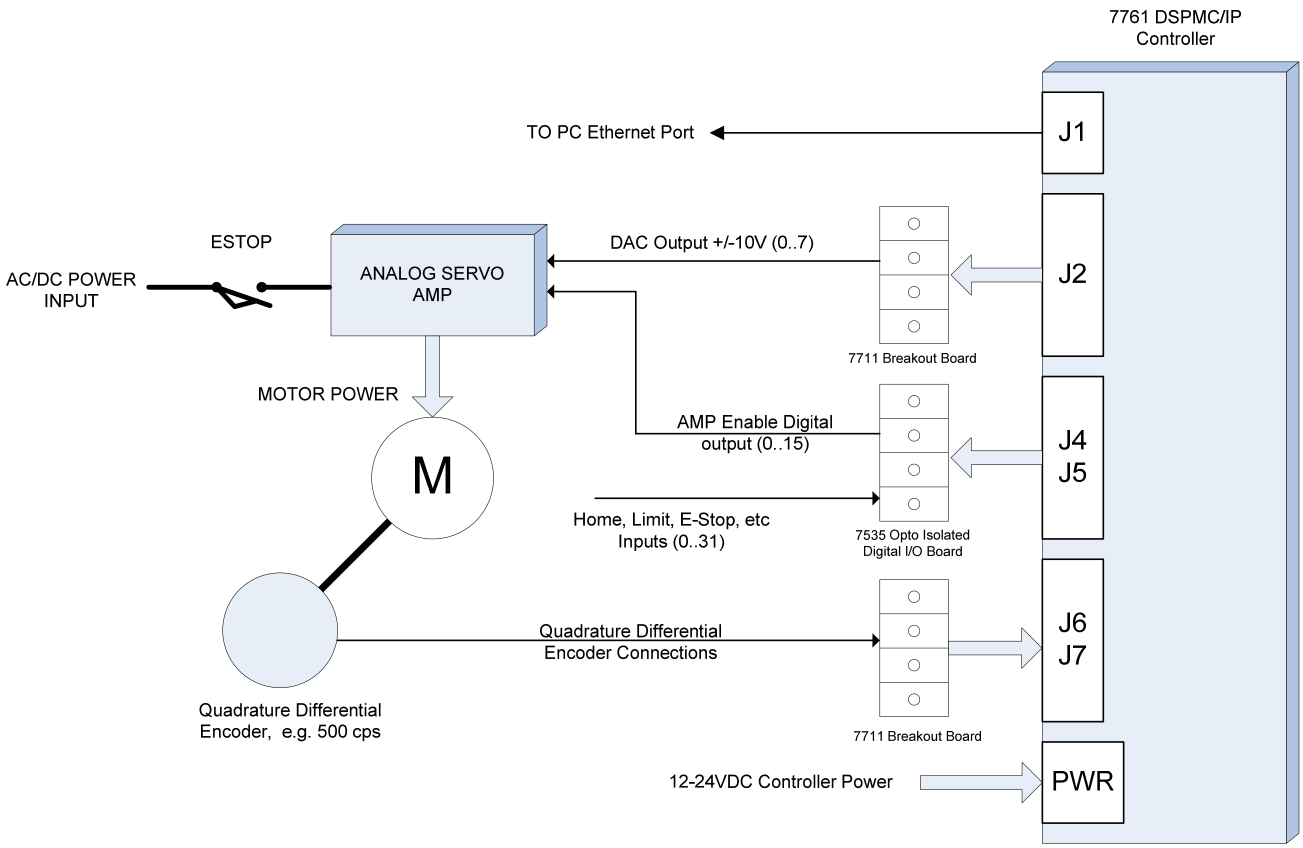
SoftEncoder 2 : A+ On J5 Pin 18
B+ On J5 Pin 6

SoftEncoders are normally used as MPG source. See Section 7.11.7 Manual Pulse Generation (MPG) for more information.

In addition, *SoftEncoders* are designed to avoid Jerks when MPG scale is changed by the PC software. To accomplish this, these encoder counters are implemented using floating point numbers (instead of whole integer numbers). They can be assigned decimal values (eg 3.092, 5.001, 64000.5 etc) which the hardware encoder counters are not capable of (they can only take whole integers).

5. Hardware Connections

The figure below shows a typical axis setup using Analog Servo amplifier and quadrature encoder feedback. The connectors on DSPMC board are all DB25 connectors. The user can wire up the system using the 7711 (or 7721) and 7535 breakout boards as shown in the figure. The Encoder Cable should be shielded with the shield properly grounded.



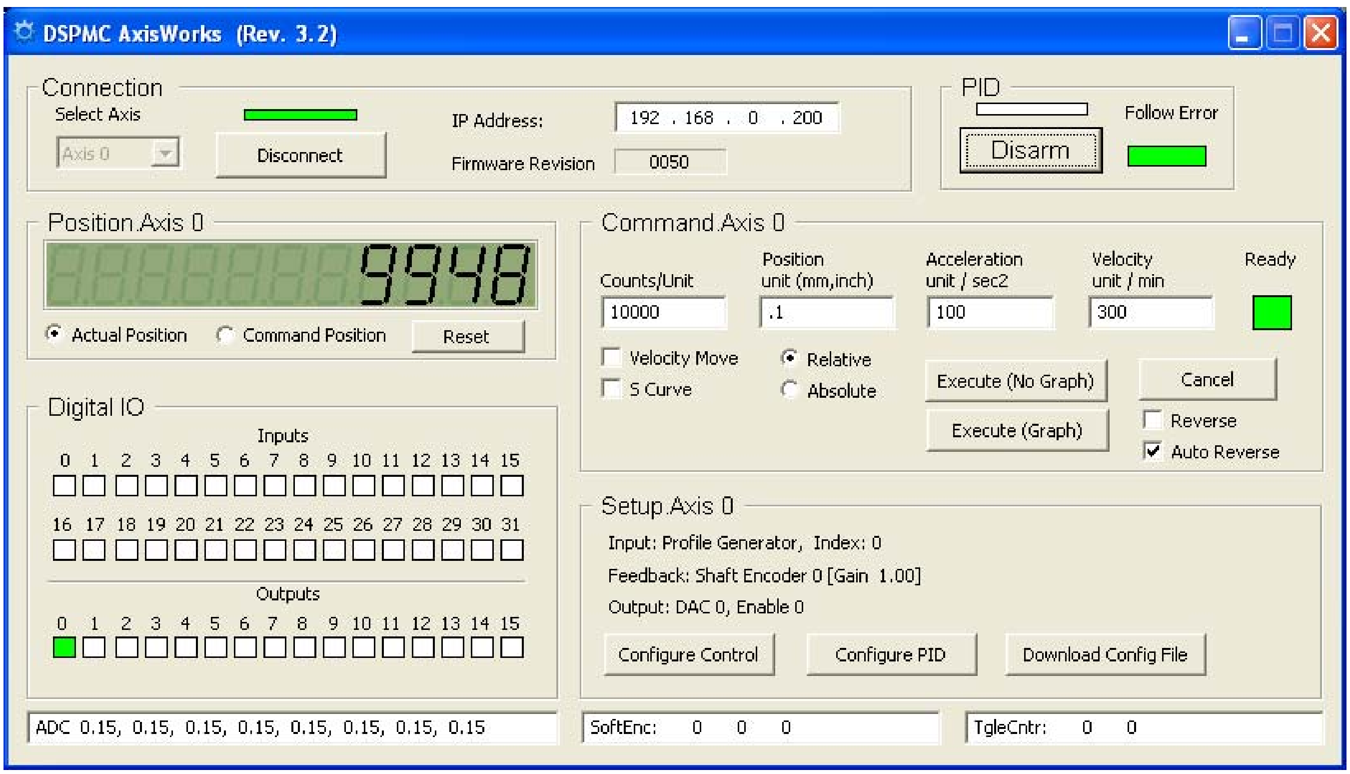
6. AxisWorks Software tool

The AxisWorks software tool allows the user to test the DSPMC installation. The user can exercise motion and I/O controls easily, without doing any custom programming. The tool lets the user configure the I/O and PID settings. I/O control configuration includes configuring the servo loop Input source and gain, Feedback source, output control and amp-enable output control etc. Once configured, the user can run the motion commands and can also tune the PID controls. PID configuration includes configuring each of the P, I, and D gains as well as set the error limits if any.

Note for Mach3 Users

With DSPMC Mach3 Plugin Release 3, the users do not need to use AxisWorks Software for tuning and other configuration settings. The new plugin provide easy to use intuitive configuration screens to fully configure DSPMC to work with Mach3 software. Please skip this chapter and go directly to 7. Mach3 Software Integration.

Upon double-clicking the DSPMC AxisWorks icon on the desktop, the initial window would look similar to the following. Each screen item is explained in detail as follows



6.1 User Interface

1 – Connection – Provides details for connecting AxisWorks to the DSPMC board

Before clicking on the 'Connect DSPMC' button, check whether the board has been powered on and network setup has been done as explained in *Network Connection Setup*.

Note that the IP Address and Firmware revision fields (which specify the DSPMC board's IP address and current loaded firmware revision) are left blank before the connection to the board is made.

A 'Red' bar above the 'Connect DSPMC' button indicates that the connection is not yet made. In order to initiate the connection, select an Axis (out of the 8 Axes from the drop down list). Then click on 'Connect DSPMC' button, this performs the actual connection to the board. , If connection is successful, the 'Red' bar will turn into 'Green' color.

2 – PID – Provides control for Arming (enable) and Disarming (disable) of PID loops globally. Also a color bar in red shows if a Following Error is tripped. If green, no error is detected.

3 – Position Axis – Displays actual position and command position for the selected Axis in raw encoder counts.

By selecting 'Command position', the display shows the value of the internal variable for the commanded position for the selected axis.

By selecting 'Actual position', the display shows the current value of the encoder counter for the selected axis.

Note that the actual position usually slightly deviates from the Command position when PID is enabled.

4 – Command Axis - Executes Move command based on User inputs for selected Axis.

Counts/Unit – Number of encoder counts in one Position Unit (e.g. mm, inches, etc). For example, if your preferred unit of measurement is inches, and if 40000 encoder counts move the axis 1 inch, you will put 40000 in this field. If you are going to work directly in encoder counts, then set this value to 1.

For CNC applications, this is calculated by encoder counts as follows:
Encoder Counts / Inch = Quadrature counts/rev x Gear ratio x lead-screw revs/inch.

For example
Encoder: 2000 Counts / Rev (Actual 500 CPR)
Gear Ratios: 2
Lead Screw: 5 Revs / Inch

Encoder Counts / Inch = 2000 x 2 x 5 = 20000

Position – Final position or displacement in terms of Position Units, e.g. 1.5, 10.093, etc.

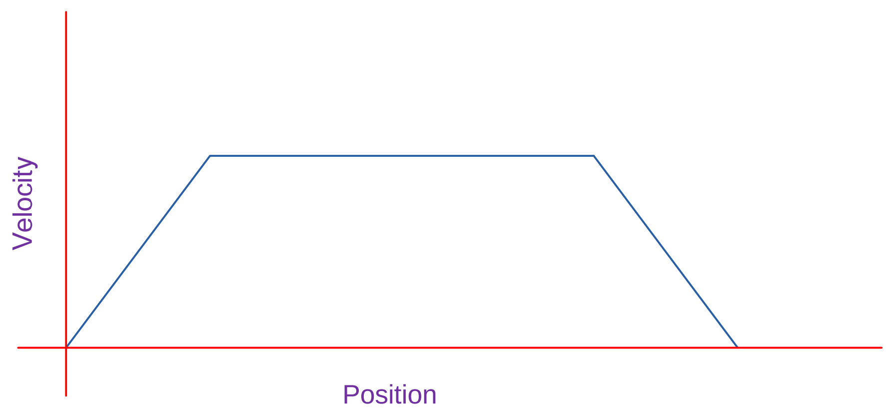
> ***Note:***
> Be careful not to use Acceleration and Velocity values greater than the machine is capable of, or you will cause a "Following Error". Start slow and work your way up. When you get an error that you can't tune out, back your settings down to 10% or so.

Acceleration – Acceleration value in terms of Position Units per second squared, e.g. inches/second^2, mm/sec^2 etc.

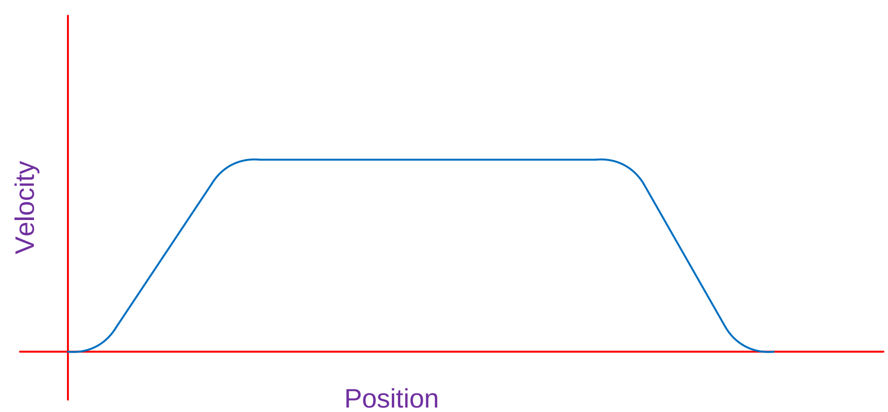
Velocity – Velocity value in terms of Position Units per minute, e.g. inches/minute, mm/mintue etc.

Relative and *Absolute* – These buttons indicate whether the value in the Position field is either the total distance to travel (relative) or the final position (absolute).

Velocity Move – This indicates that the move will not use the position data provided. This is useful when you need to run the axis for long time without worrying about the final position. To stop the axis, click on the cancel button
S-curve – This indicates that the motion profile will transition smoothly (no sudden change in acceleration). The motion profiles with or without S-Curve option are shown below:

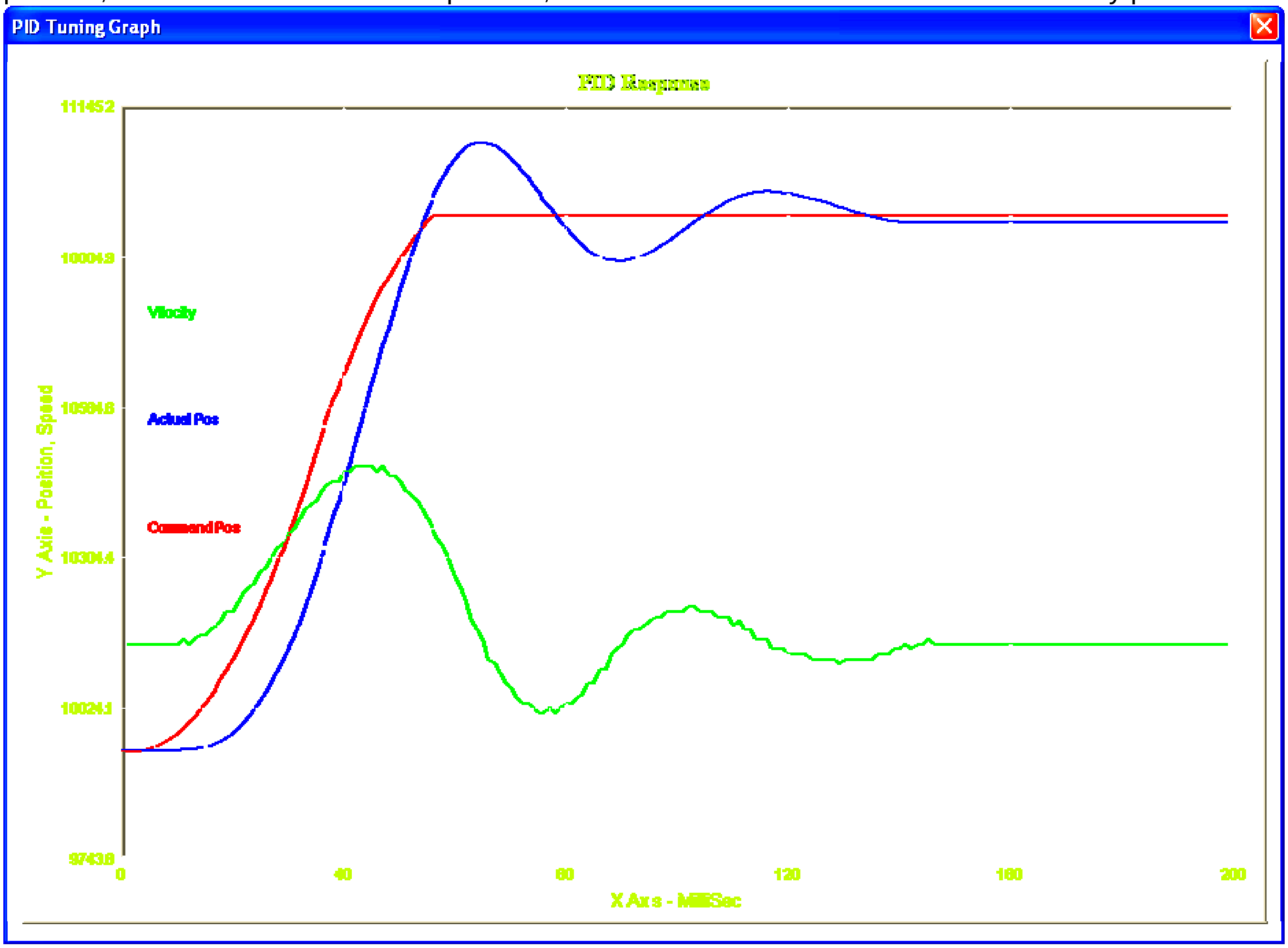


Trapezoidal Motion Profile



S-Curve Motion Profile

Pressing the 'Execute' button (with or without graph) starts the motion. User can press 'Cancel' button to cancel the motion execution anytime during the machine operation. Make sure you have setup the axis properly before moving the axis, as described in the following sections. If Execute with Graph is selected, the Graph window will appear, similar to shown below. The red line shows the commanded position, Blue line shows the actual position, while the Green line shows the actual velocity profile.



5 – Digital I/O – Displays the current states of digital Input and Output pins.

6 – Download Config File – Downloads the user specified configuration file (e.g. DSPMC.xml) to the board. Appendix A shows the format of this xml file. You can manually edit this file and re-download the configuration without going thru different windows.

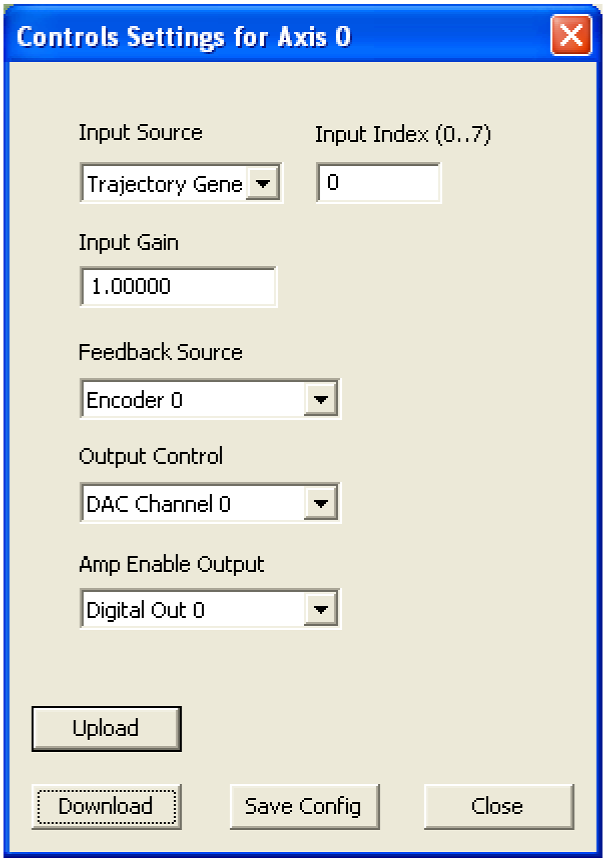
7 – Setup Axis – Setup axis controls (input, feedback, output etc) and PID configuration for the selected axis.

8 – Motion Profile Graphs – Displays actual versus programmed axis motion

The Current configuration values for 'Input', 'Feedback' and 'Output' are displayed for the selected axis. 'Configure Control' and 'Configure PID' provides dialog windows to the user to add/modify the configuration and PID settings.

6.2 Controls Configuration

The '**Configure Control**' button displays following dialog window:



Input source is the command input to the PID controller for the axis. This can be any one of the following:

- PC Host Software (this configuration is required by some PC software, e.g. Mach3)
- Trajectory Generator
- Any Encoder input
- Any analog input

Important Note:
You must select "Trajectory Generator" as the Input Source when using ***AxisWorks*** to execute motion on DSPMC Controller.

Input Index is the index of the command position source. This number normally match the axis number. In case for slave axis, it is equal to the master axis number.

Input Gain is the multiplier before the source data is used, e.g. if encoder 0 is the source, the counter value is multiplied by Gain and then used as the input to the PID controller for the selected axis.

Feedback source to the PID controller can be from any Encoder input or any Analog inputs (A/D channel 0-7)

Output control selects the DAC channels (DAC 0-7) that will receive the PID controller output.

Amp Enable Output selects the output pin used to enable the external servo amplifier. User can select any of the 16 digital output pins (Digital out 0-15) as the enable pin for the selected axis.

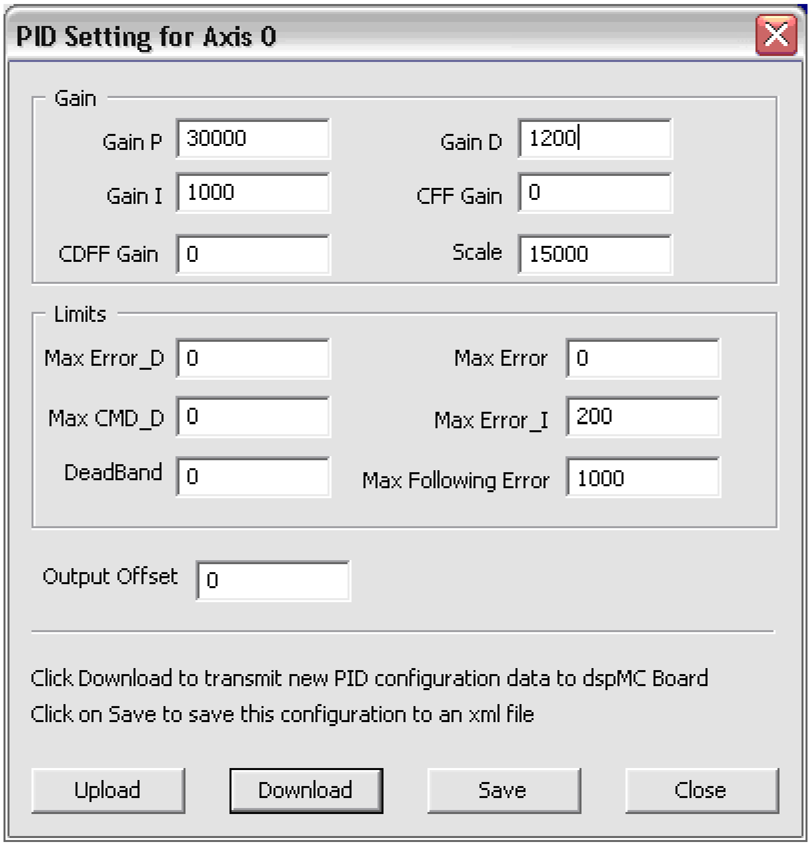
Note that an 'undefined' value is displayed for these fields when not configured.

The ***Upload*** button reads the current values from the DSPMC board and displays them on this window. If you make a change and like to see the original value you can click on Upload button to get the current programmed values inside the DSPMC.

Clicking the ***'Download'*** button send the configuration to the DSPMC board. Note that these configurations are stored on volatile memory inside the board, i.e., the information will need to be re-downloaded when power is cycled on the controller. To avoid re-entering the information, you can also save the configuration to the DSPMC.xml file by clicking on the '***Save Config***' button. Once saved, the file can be downloaded to the DSPMC by clicking on the '***Download Config File***' button on the main window.

6.3 PID Filter Configuration

The ‘**PID Settings**’ button displays the following dialog window:



Important Note:

Make sure not to use 0 in the Max-Following-Error field. Always use a non-zero positive value. If it is set to 0, the motor can move at max uncontrolled speed (in a run-away situation), which can be extremely dangerous.

User can input the P, I, and D coefficients as well as error limits and scale.

Scale is used as a divider for all the setting in the PID gain box. This makes it possible for the software to use smaller manageable numbers and still get the same effect. An example is if you put a gain of 4000 and a scale of 100, it is the same as a scale of 1 and a gain of 400,000.

Don’t be concerned if on the first time you don’t get any motion as the numbers are trial and error. Using P to D ratio of 25 to 1 works real well. If you have no experience, try these numbers:
P = 4000, I = 0, D = 100000, Max following error 1000, Dead-band = 3, Scale = 100 and all others to 0. If your servo drives are tuned real well, Deadband can be set at zero but start at 3 or 5, higher if your servo’s buzz at a standstill.

Command Feed Forward (CFF) gain and Command Differential Feed Forward (CDFF) gain for PID loop control can also be entered here. These are definitely for the more advanced user but will allow for more precise movement.

User can also specify following limits…

Max_error - Maximum error limit. Leave it at 0.
Max_Error_D - Maximum Derivative Error for Derivate gain,
Max_CMD_D - Maximum Command Error for CDFF gain,
Max_Error_I - Maximum Integral Error for the integral gain,

Deadband – a range of position around the commanded position where the PID is not active (when armed), e.g. if current command position is 1000 count, and Deadband is 10, the PID will be inactive between 990 and 1010 count.

Max Following Error – Maximum deviation allowed between command and actual, above that, the PID controller shuts down and need to be re-enabled manually. If 0, PID will never shutdown which can be extremely dangerous in a run-away motor condition. So always use a positive value in this field. This field can be back calculated from the maximum velocity of the axis, e.g. 600 000 count/sec max velocity divide by 1000 gives 600 counts per millisecond. So to achieve 600K count/sec speed, the max following error should be 600 or more. The actual value may be lot more than that based on how tight the PID tuning is, and the mechanical characteristics of the axis.

Output Offset - Sets a constant bias to the PID output. This is useful when tuning Z axis, where motor has to apply more pressure in one direction than the other.

Clicking on '***Download***' sends the PID configuration data to the DSPMC board. The new PID configuration can also be saved to an XML file by clicking '***Save***' button. Once saved, there is no need to enter these values every time you run AxisWorks. You can click on the '***Download Config File***' button on the main window to send the configuration to the DSPMC controller.

If you changed any field and like to revert back or see the current PID settings inside the DSPMC, click on the '***Upload***' button to read back the data from the controller.

7. Mach3 Software Integration

The Mach3 Software is an off-the-shelf Milling and Lathe machine control software. User can download the trail version of the software from www.machsupport.com

The DSPMC board can be integrated with Mach3 to form a high performance machining center. The DSPMC Software Tools provide the necessary drivers and configuration files to interface with Mach3 software. If you have installed the software tools as explained in the software installations section, you already have all the necessary drivers.

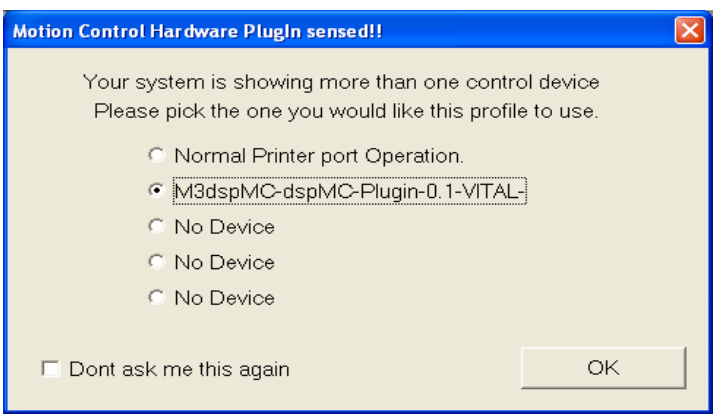
This document assumes that user is familiar with the usage of Mach3 software. This chapter describes the mapping of Mach3 internal software signals to the DSPMC connector.

With DSPMC Mach3 Plugin Release 3, the users do not need to use AxisWorks Software for tuning and other configuration settings. The new plugin provide easy to use and very intuitive configuration screens to fully configure DSPMC to work with Mach3 software.

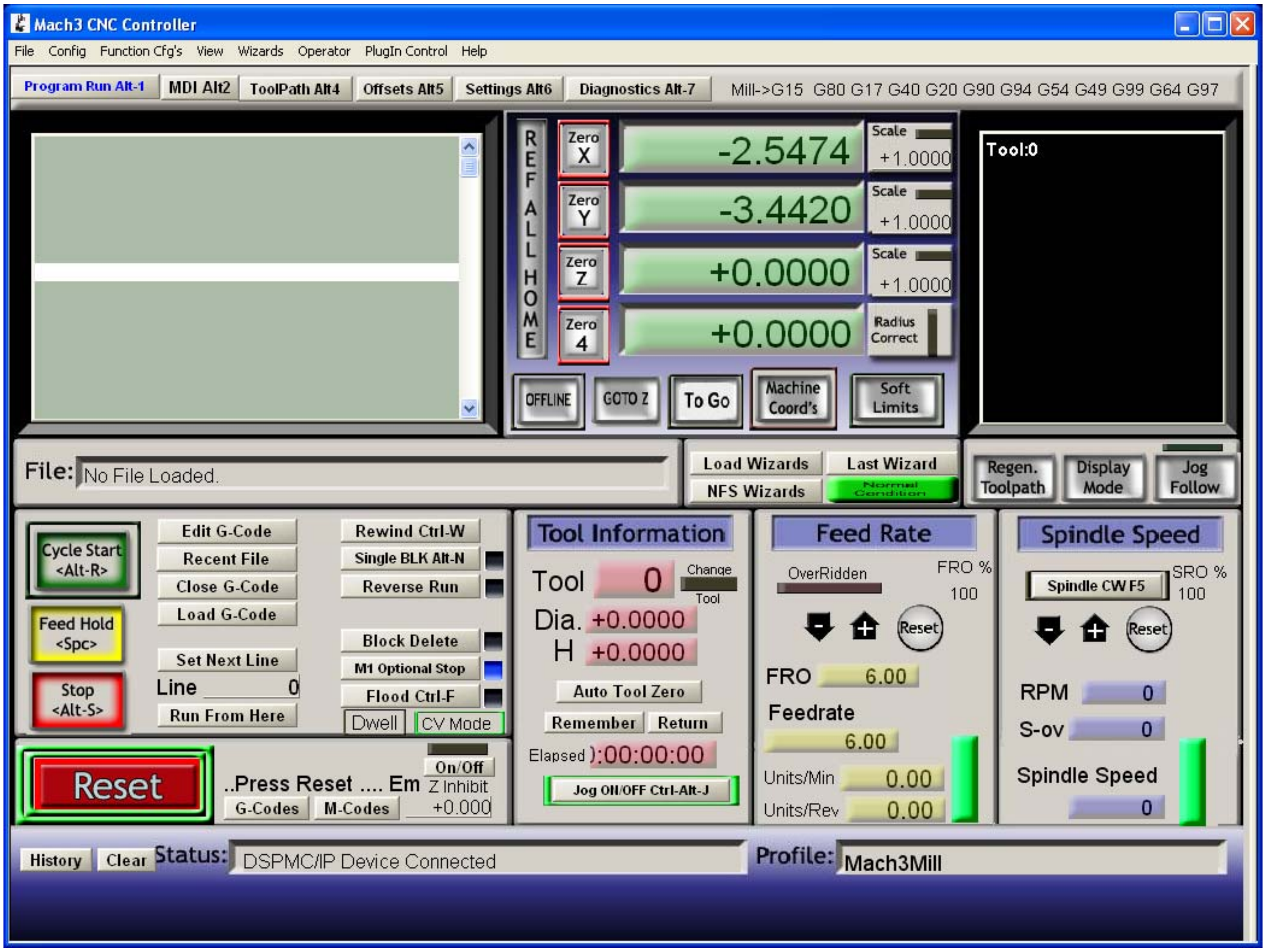
The general Mach3 software operation remains mostly the same when using DSPMC plugin.

7.1 Starting Mach3 with DSPMC

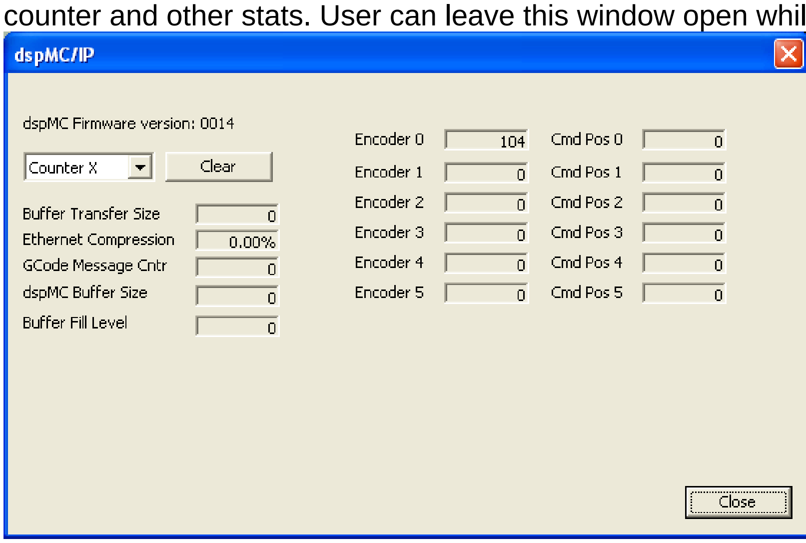
To launch Mach3 with DSPMC plugin, double-click on the Mach3 Mill or Lathe software icon on the desktop as you would normally run using parallel port. It shows the following dialog box with the option to select M3dspMC plugin. Make sure this plugin is selected and click 'OK'.



Make sure the DSPMC is powered up and connected to the Ethernet network. The Mach3 software shows up as follows with a message 'DSPMC/IP Device Connected' in the Status bar.



Click on the Menu item 'PlugIn Control', and then click on the item 'VITAL DSPMC/IP Window" which displays the following screen indicating that the DSPMC board is connected with current states of counter and other stats. User can leave this window open while running Mach3.



This screen is mostly used for diagnostics purposes.

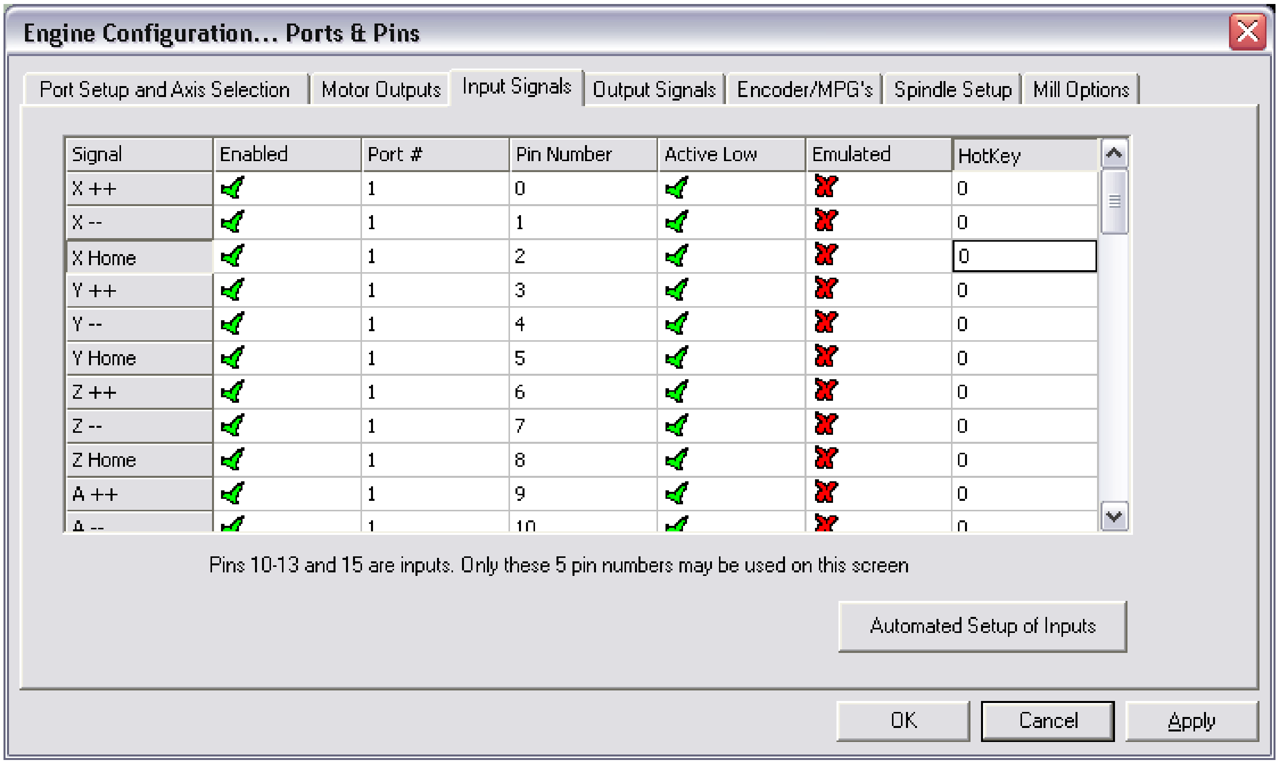
7.2 Mapping Mach Input Signals to DSPMC Digital Inputs

The following table shows the mapping from Mach3 input pin numbers to the actual digital input pin numbers available on the DSPMC board.

Mach3 *Input* ports and pins		**DSPMC J4 and J5 pin assignments**		**Breakout board 7535 pin assignments**	
Port #	**Pin Number**	**J4 Pin number**	**J5 Pin number**	**Board #1**	**Board #2**
1	0	**18**		0	
1	1	**6**		1	
1	2	**19**		2	
1	3	**7**		3	
1	4	**20**		4	
1	5	**8**		5	
1	6	**21**		6	
1	7	**9**		7	
1	8	**22**		8	
1	9	**10**		9	
1	10	**23**		10	
1	11	**11**		11	
1	12	**24**		12	
1	13	**12**		13	
1	14	**25**		14	
1	15	**13**		15	
1	16		**18**		0
1	17		**6**		1
1	18		**19**		2
1	19		**7**		3
1	20		**20**		4
1	21		**8**		5
1	22		**21**		6
1	23		**9**		7
1	24		**22**		8
1	25		**10**		9
1	26		**23**		10
1	27		**11**		11
1	28		**24**		12
1	29		**12**		13
1	30		**25**		14
1	31		**13**		15

Mach3 *Input* ports and pins		DSPMC J11 and J12 pin assignments		Breakout board 7535 pin assignments	
Port #	**Pin Number**	**J11 Pin number**	**J12 Pin number**	**Board #3**	**Board #4**
1	32	**18**		0	
1	33	**6**		1	
1	34	**19**		2	
1	35	**7**		3	
1	36	**20**		4	
1	37	**8**		5	
1	38	**21**		6	
1	39	**9**		7	
1	40	**22**		8	
1	41	**10**		9	
1	42	**23**		10	
1	43	**11**		11	
1	44	**24**		12	
1	45	**12**		13	
1	46	**25**		14	
1	47	**13**		15	
1	48		**18**		0
1	49		**6**		1
1	50		**19**		2
1	51		**7**		3
1	52		**20**		4
1	53		**8**		5
1	54		**21**		6
1	55		**9**		7
1	56		**22**		8
1	57		**10**		9
1	58		**23**		10
1	59		**11**		11
1	60		**24**		12
1	61		**12**		13
1	62		**25**		14
1	63		**13**		15

If using any of the Mach3 input signals, make sure the pins are 'Enabled' and set 'Active Low' as shown in the example figure below.



Ignore the line printed on the above window about pins 10-13 and 15 !@#$. This does not apply to DSPMC based system.

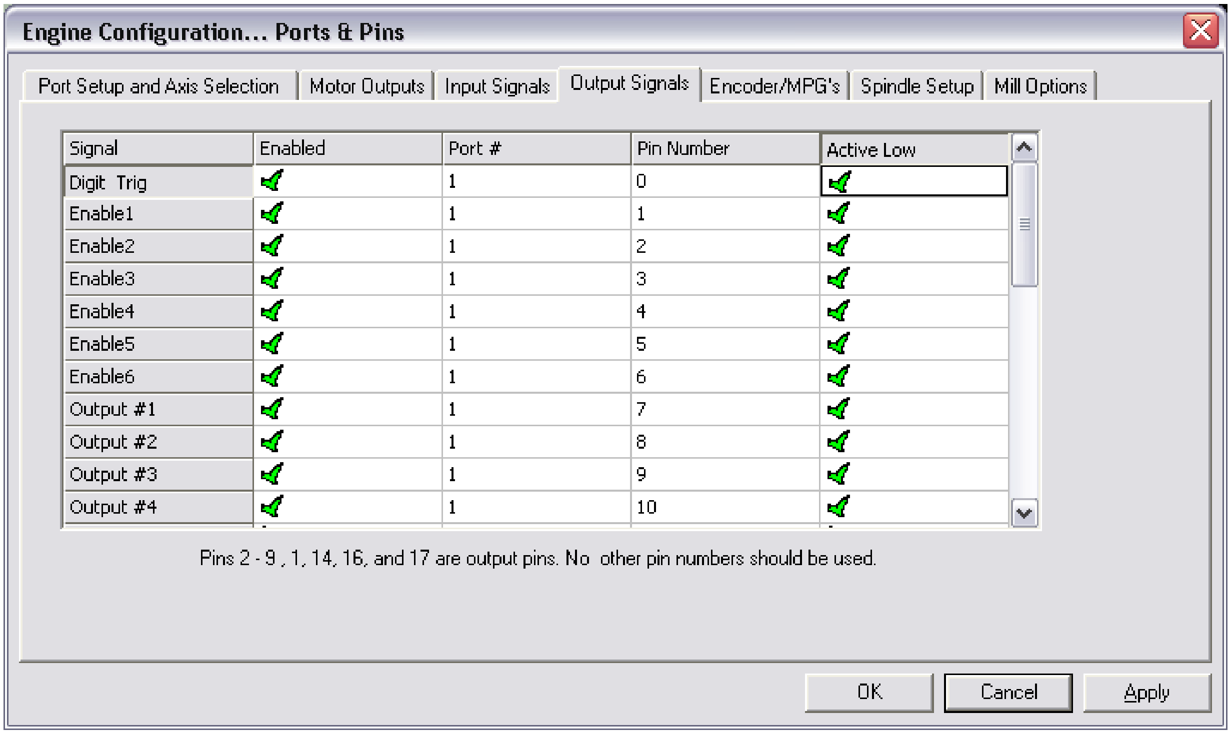
7.3 Mapping Mach3 output pins to DSPMC Digital Outputs

The following table shows the mapping from Mach3 output pin numbers to the actual digital output pin numbers available on the DSPMC board.

Mach3 *Output* ports and pins		**DSPMC J4 and J5 pin assignments**		**Breakout board 7535 pin assignments**	
Port #	**Pin Number**	**J4 Pin number**	**J5 Pin number**	**Board #1 On J4**	**Board #2 On J5**
1	0	**14**		0	
1	1	**2**		1	
1	2	**15**		2	
1	3	**3**		3	
1	4	**16**		4	
1	5	**4**		5	
1	6	**17**		6	
1	7	**5**		7	
1	8		**14**		0
1	9		**2**		1
1	10		**15**		2
1	11		**3**		3
1	12		**16**		4
1	13		**4**		5
1	14		**17**		6
1	15		**5**		7

Mach3 *Output* ports and pins		**DSPMC J11 and J12 pin assignments**		**Breakout board 7535 pin assignments**	
Port #	**Pin Number**	**J11 Pin number**	**J12 Pin number**	**Board #3 On J11**	**Board #4 On J12**
1	16	**14**		0	
1	17	**2**		1	
1	18	**15**		2	
1	19	**3**		3	
1	20	**16**		4	
1	21	**4**		5	
1	22	**17**		6	
1	23	**5**		7	
1	24		**14**		0
1	25		**2**		1
1	26		**15**		2
1	27		**3**		3
1	28		**16**		4
1	29		**4**		5
1	30		**17**		6
1	31		**5**		7

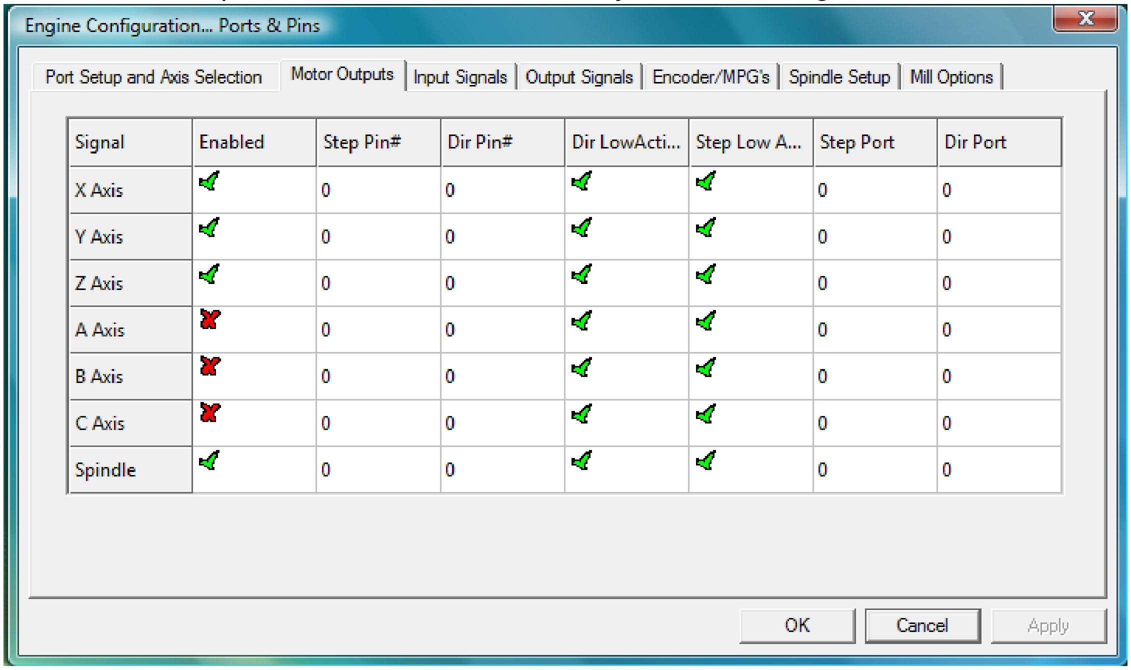
As with the input configuration, if using any of the Mach3 output signals, make sure the pins are 'Enabled' and set 'Active Low' as shown in the example figure below.



Ignore the line printed on the above window about pins 2-9,1,14… !@#$. This does not apply to DSPMC based system.

7.4 Motor outputs.

On the Motor Ouputs tab, enable the ones that you will be using, and set to active low.



7.5 Spindle Setup.

When using a VFD or other motor controlling device that uses 0-10v control the following steps are needed.   The selection of analog output channel is done via the Plugin config screen.

Make sure the spindle enable is checked and is set to active low.

Go to the Config tab and then spindle pulleys. Current pulley 1. For this example, set min speed to 0 and max speed to 100. Set Ratio 10. This will give a 0v output to the spindle at S0 (min speed) and a 10v at S100 (max speed).

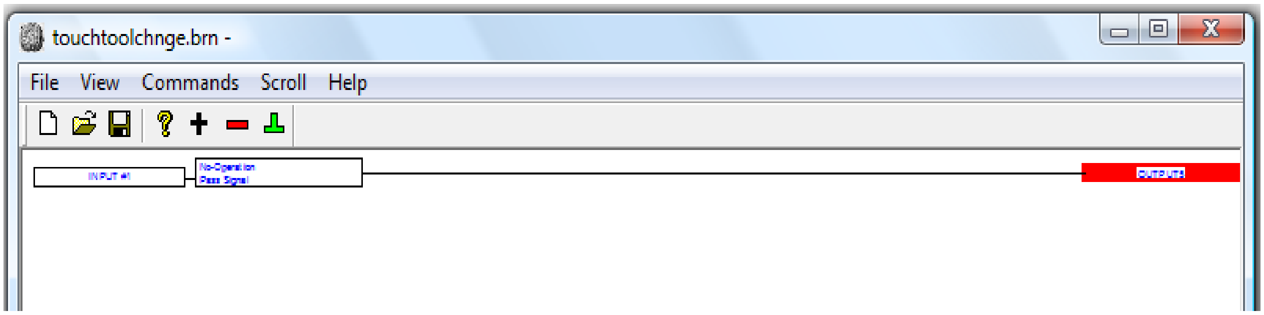
This setting is great for testing. Without the VFD/Drive hooked up you can test your output with a digital volt meter to make sure you are getting 0-10volts for 0 to max speed.

When it all works then put in min 0 and max gets set to the max speed of your machine, eg, 5000. This will allow you to program S in the G-code in actual rpm, ie 0 … 5000.

7.6 Getting beyond the basic input/output with Mach3.

When your are done with limit switches and other basic I/O you will probably want to have several switches on you control panel next to the e-stop such as feed hold, stop, g-Code rewind and other things. To get this added functionality you will need to learn how to write brains in Mach3. Brains are used to get access to all of the extra I/O and to work tool changers and just about anything you can think of.

Here is a pretty basic brain to map input 1 to output 5.



You would setup what wire goes to input 1 and output 5 under the ports and pins tab of Mach3. After a few tries you will get the hang of it.

For more information: go to www.machsupport.com and then to the video section and look for Brains.

7.7 Axis Homing and Direction

In the Config menu, select Homing/Limits. You will see the following window.

Axis	Reversed	Soft Max	Soft Min	Slow Zone	Home Off.	Home Neg	Auto Zero	Speed %
X	✓	100.00	-100.00	1.00	1	✗	✓	20
Y	✗	100.00	-100.00	1.00	1	✓	✓	20
Z	✓	100.00	-100.00	1.00	.5	✗	✓	20
A	✗	100.00	-100.00	1.00	0.0000	✗	✓	20
B	✗	100.00	-100.00	1.00	0.0000	✗	✓	20
C	✗	100.00	-100.00	1.00	0.0000	✗	✓	20

To change the axis direction, click on the Reversed column for the axis you want to change the direction. A green check mark indicates the direction is now reversed.

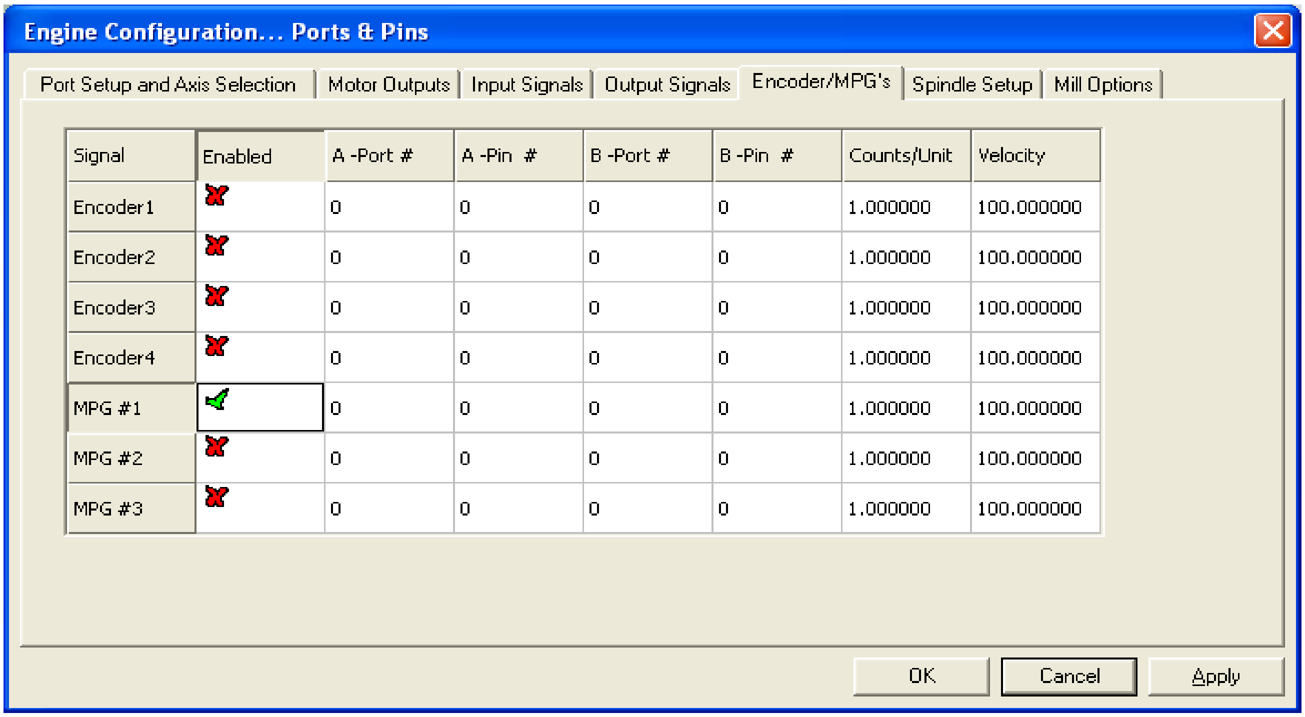
Homing Offset lets you define the home position co-ordinates. When homing sequence is complete, the axis machine position is set to this value. Home Neg, changes the default homing direction.

The DSPMC offers a number of different homing types for each axis. Please review section 7.12.3 Control Parameters to select the correct homing sequence.

If you are using Index-Pulse-Only Homing without a home sensor, you must assign an unused digital input in Mach3 as home sensor. Mach3 software requires a home sensor definition regardless of homing method.

7.8 Manual Pulse Generation - MPG

DSPMC allows using a quadrature encoder as a MPG source. The encoder is connected to the dedicated encoder inputs on J3, J6 and J7 (HardEncoder), as well as to J4 and J5 digital inputs (*SoftEncoder*). Users can configure MPG parameters as explained in section 7.11.7 Manual Pulse Generation (MPG).



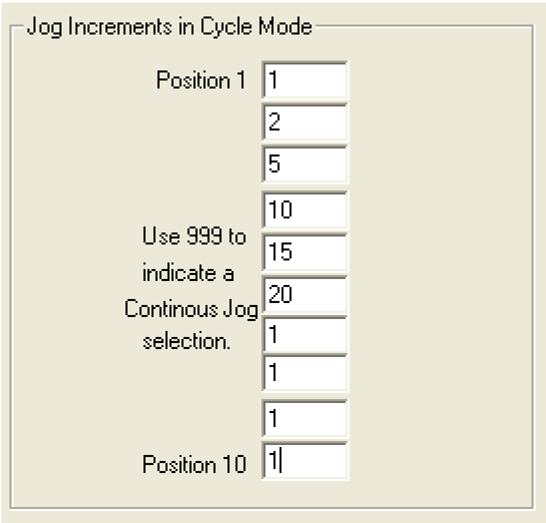
Signal	Enabled	A -Port #	A -Pin #	B -Port #	B -Pin #	Counts/Unit	Velocity
Encoder1	✗	0	0	0	0	1.000000	100.000000
Encoder2	✗	0	0	0	0	1.000000	100.000000
Encoder3	✗	0	0	0	0	1.000000	100.000000
Encoder4	✗	0	0	0	0	1.000000	100.000000
MPG #1	✓	0	0	0	0	1.000000	100.000000
MPG #2	✗	0	0	0	0	1.000000	100.000000
MPG #3	✗	0	0	0	0	1.000000	100.000000

To turn on MPG feature, make sure MPG #1 is checked green as shown above in the Ports and Pins window. Enter the Encoder resolution in the Counts/Unit field. The rest of the fields in this window are not used.

The *SoftEncoder* are available on digital inputs of connector J4 and J5. The pin assignments are as follows:

SoftEncoder 0 : A = J4 Pin 24, B = J4 Pin 12
SoftEncoder 1 : A = J4 Pin 25, B = J4 Pin 13
SoftEncoder 2 : A = J5 Pin 18, B = J5 Pin 6

You set the encoder multiplier in the General Config setting as shown below. You can use your own multiplier values in this window as well as use the standard .1, .01, .001, etc values.



When MPG mode is selected, and a G-Code file is run, the DSPMC Plugin will switch to jog mode automatically in order to run the file. Once the file is complete or stopped, the mode will revert back to MPG.

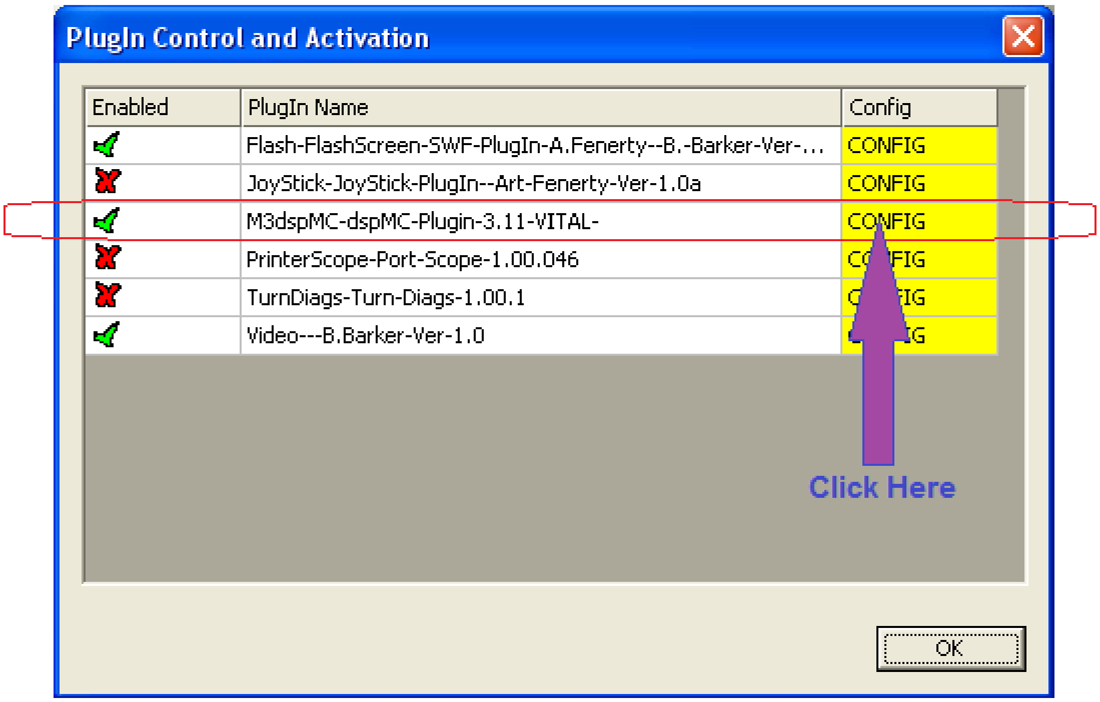
7.9 OEMDROs

The following table lists the OEMDROs used by the DSPMC plugin.

OEMDRO Index	Description
1320 thru 1327	Analog Input data from ADC channels 0..7. Available only with 7761-M model.
1330 thru 1332	Digital Input Encoders (SoftEncoder) on J4 and J5. **SoftEncoder 0** : A+ On J4 Pin 24, B+ On J4 Pin 12 **SoftEncoder 1** : A+ On J4 Pin 25 B+ On J4 Pin 13 **SoftEncoder 2** : A+ On J5 Pin 18 B+ On J5 Pin 6
1340	Threading RPM. This RPM is calculated based on the parameters defined in the Threading section of DSPMC Plugin Configuration, and is used by the Threading Logic.

7.10 DSPMC Plugin Configuration

DSPMC plugin configuration screens can be launched from Mach3 by navigating to Config -> Config Plugins and selecting M3dspMC-DSPMC-Plugin-xxx-VITAL- option.



Once DSPMC plugin configuration is launched you can see nine tabs, these are:

1. System Tab
2. Axis X(0)
3. Axis Y(1)
4. Axis Z(2)
5. Axis A(3)
6. Axis B(4)
7. Axis C(5)
8. Axis D(6)
9. Axis E(7)

Each Axis tab represents an axis to be controlled through DSPMC. By default, System tab will be selected, as shown in DSPMC Configuration window.

7.11 DSPMC Plugin Configuration System Tab

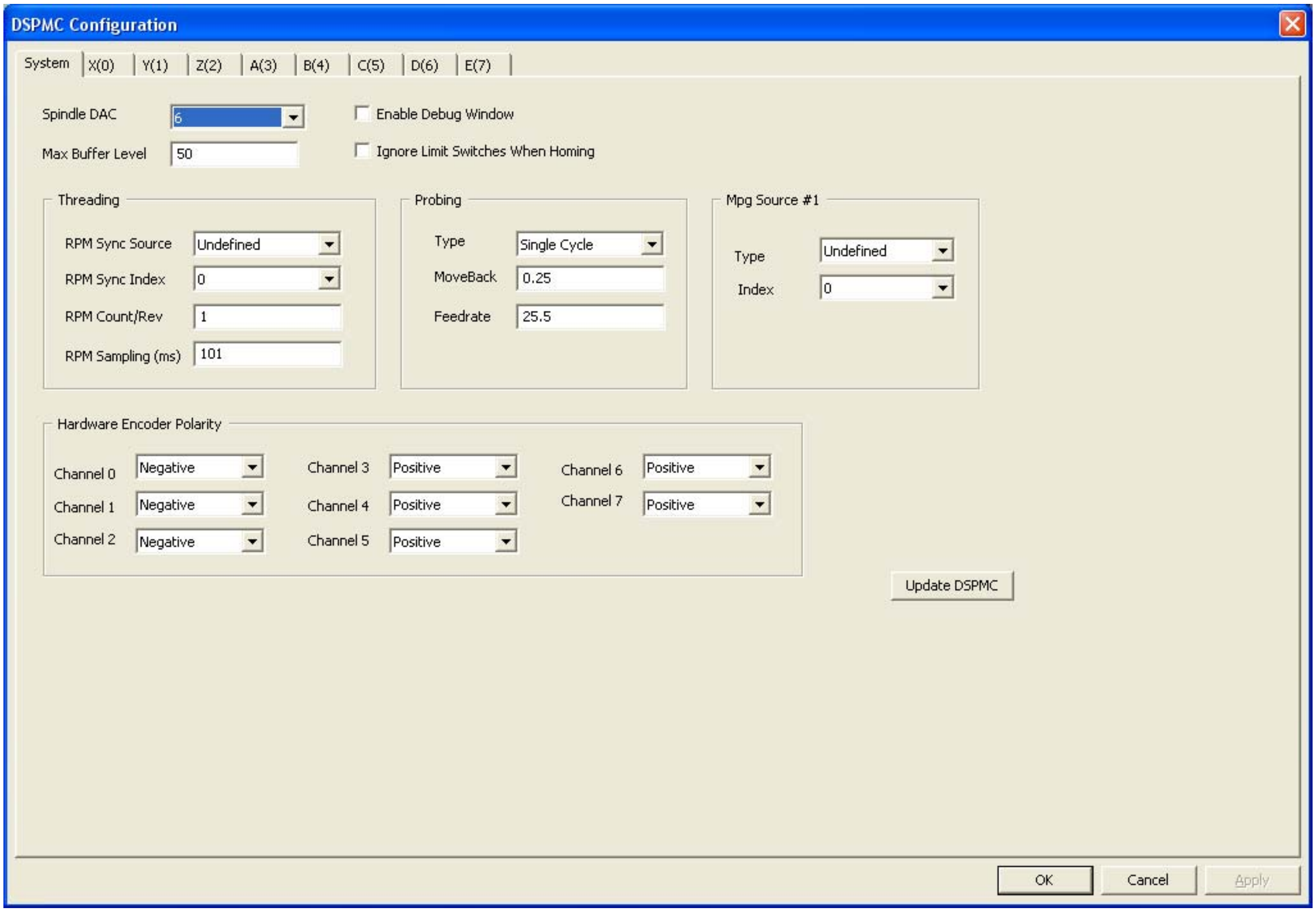


Figure: DSPMC Configuration window

In the system tab, you can set a number of configurations. Clicking on the Update DSPMC will transmit these settings to the DSPMC controller. Clicking OK will also transmit the data, and also save this data in selected mach3 profile (e.g. mach3mill, mach3turn etc)

In the following sub sections users can find detailed information about various configuration options that are provided under the system tab.

7.11.1 Spindle DAC

The Spindle DAC configuration selects the analog output (DAC) for the spindle speed control. The DAC output voltage varies from 0 to 10volts. Any Analog output position can be selected from 0 to 7. Make sure the selected DAC output number is not used in any other axis's Control Output Index. If it is used, make sure the axis Control Output is select as "Undefined".

7.11.2 Max Buffer Level

This parameter defines how much command position buffering will be done inside the DSPMC controller. The total size of the buffer is 4096 points per axis. These points are consumed by the DSPMC at 1 KHZ. To get faster response time on feedrate changes, you may select a lower value, but the side effect is that if the PC software slows down and cannot sustain the motion data rate to the DSPMC, then the motion could be jerky.

The valid rage for this parameter is 1…100 percent.

7.11.3 Enable Debug Window

By checking this option users can enable debug window for debugging purpose. This option should be turned on only if directed by a factory personal.

7.11.4 Ignore Limit Switches when Homing

DSPMC allows Limit switches to be used as Home Sensors. This parameter should be checked to use Limit switches as Home sensor.

7.11.5 Threading

Following sub sections describes parameters to configure threading in DSPMC.

Threading RPM Synch Source

This parameter defines the encoder type for Spindle speed calculation and starting the threading cycle. The Index pulse from the encoder is used to launch the Z-Axis at the right time in order to position the tool correctly for Threading in every cycle. The RPM calculation is used to override the feedrate of the Z-Axis during the threading cycle.

Mach3 Software requires that the Spindle RPM to be constant before the threading G-Code command is executed. But, in reality, there is always some variation in actual RPM calculation. As the DSPMC plugin has no knowledge of which G-Code command is going to run, the plugin has no way to know when to keep the RPM value constant. To solve this problem, the plugin looks at ***OEMLED 1999*** to know when to pass the calculated RPM to Mach3 (so user/mach3 can see the actual spindle RPM), or to pass the fixed programmed (in G-Code) RPM value for Mach3 threading logic. The user can add a macro in the G-code program to turn on ***OEMLED 1999*** before any threading command, and to turn it off after the threading command, in order to work around this issue.

Two possible values for **RPM Sync Source** parameter are: "***HardEncoder***" and "***DigitalInput***".

Undefined: When this option is selected DSMPC will not enable threading and value of **RPM Synch Index** will be ignored.

When ***HardEncoder*** is selected, the spindle feedback encoder must be connected to one of the encoder inputs on J3, J6, J7 and J8 connectors. The encoder's differential A and B signals are used to calculate the RPM of the spindle, and Index pulse is used to trigger the threading cycle.

When ***DigitalInput*** is selected, the spindle feedback is generated by a single line pulse train. The pulse train is used to calculate the spindle RPM as well as used for Sync pulse to launch the threading cycle.

There are two dedicated inputs on J5 for the spindle pulse train. These are also called I/O Toggle Counters.

I/O Toggle Counter 0: Digital Input 18 (J5 Pin 19)
I/O Toggle Counter 1: Digital Input 19 (J5 Pin 7)

Threading RPM Synch Index
This parameter defines the encoder index for Spindle speed feedback. Below is the range for this index:
HardEncoder: index range is 0...7.
DigitalInput: 18(J5 Pin 19) and 19(J5 Pin 7)

Threading RPM Count/Rev
This parameter defines the encoder resolution in terms of count per revolution for Spindle speed feedback. For ***HardEncoder*** type encoder, the encoder resolution must is multiplied by 4. No multiplication is done when ***DigitalInput*** is selected.

Threading RPM Sampling (ms)
This parameter defines the timing window in milliseconds to add the encoder counts for RPM calculation. For slow pulse train (eg only few ticks per rev), this value should be high enough to accumulate enough counts to calculate RPM consistently. If the window time is too long, the system reaction time (regulation of Z-Axis feedrate) to changing RPM will be slow. A higher count/rev encoder will allow this window time to be very small, which will allow the system to react fast (regulate Z-Axis feedrate) if RPM changes. The range of this field is from 1 thru 10000 milliseconds.

7.11.6 Probing

This section defines parameter for CNC Probing feature. For the probing cycle, the axis, probing feedrate, and the probe switch are set by Mach3.

Probing Type:
This parameter defines the probing method:

SingleCycle: Axis starts the probing move. As soon as the probe switch is on, the current position is captured and the probing sequence is complete.

DualCycle: Axis starts a first probing move (called coarse move). As soon as the probe switch is on, the axis stops, and backs off distance specified in the **ProbingMoveBack** parameter. The axis then starts the second move, called fine move. The direction is same as the coarse move. The feedrate for the fine move is defined by the third parameter **Probing Feed Rate.**

Probing Move Back
This parameter defines the distance to move back to start the fine move. It is applicable only in the DualCycle mode.

Probing Feed Rate
This parameter defines the feedrate for the fine move. It is applicable only in the DualCycle mode.

7.11.7 Manual Pulse Generation (MPG)

MPG Source Type
This section defines MPG (Manual Pulse Generation) Quadrature encoder source. Both Differential and Single Ended Encoder types are supported. Differential encoder can be hooked up to any of the six encoder channel available on connectors J3, J6, J7 and J8. These encoders are defined as ***Hard Encoder***. Single Ended encoders (defined as ***Soft Encoder***) can be hooked up to the Digital Inputs available on Connector J5. An **Undefined Encoder** option causes DSMPC to ignore MPG Source values.

MPG Source Index
If Hard Encoder is selected in MPG Source Type, MPG Index denotes the hardware encoder index. If Soft Encoder is selected, MPG Index denotes available digital input options.

The pin assignments for available *SoftEncoder* are as follows

SoftEncoder 0 :	A On J4 Pin 24, B On J4 Pin 12
SoftEncoder 1 :	A On J4 Pin 25 B On J4 Pin 13
SoftEncoder 2 :	A On J5 Pin 18 B On J5 Pin 6

7.11.8 Hardware Encoder Polarity

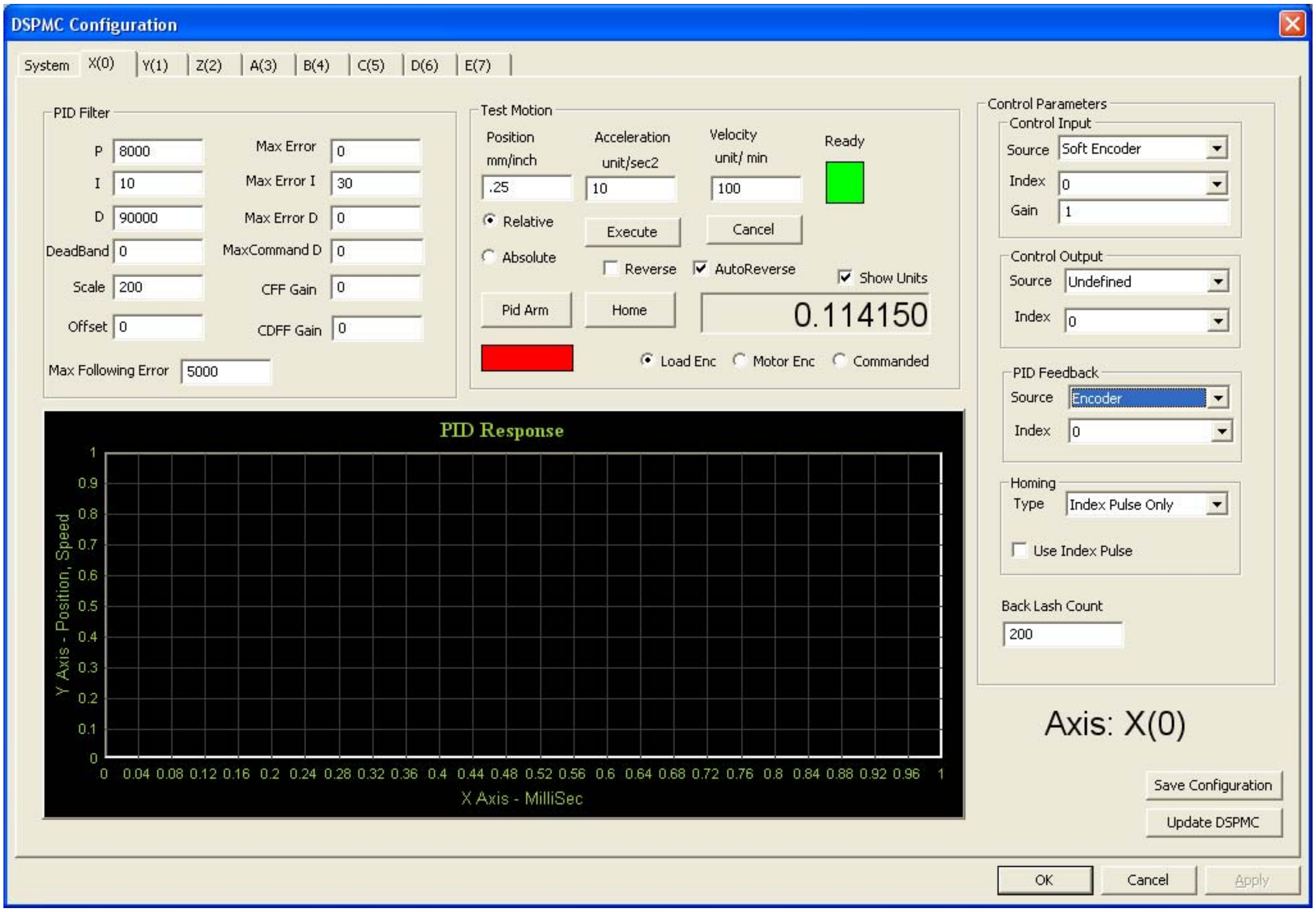
The ***Hardware Encoder Polarity*** field is used to reverse the direction of the encoder counters. If A/B signals are connected in reverse such that it does not match the PID control direction, the system will not be able to arm. To fix this issue, the hardware A and B signals can be reversed using this parameter.

Note that this encoder polarity setting only swaps the A and B signals to change the counter direction. The Index pulse signal polarity is not affected by this setting.

7.11.9 Update DSPMC Button (Under System Tab)

This button downloads the entire system configuration parameters to DSPMC. If you make any changes to Controls parameters, you must download the new settings by clicking on this button before ARMing the DSPMC or executing a test motion. To save data to your computer hard-drive, click OK. All configuration data from all pages is saved in the selected mach3 profile.

7.12 DSPMC Plugin Configuration Axis Tab



The Axis tabs provide configuration settings that are directly related to each axis. These tabs also provide motion testing features. There are two sets of parameters, PID parameters and the Controls parameters. When testing motion by pressing the 'Execute' button, the PID parameters are transmitted to DSPMC automatically before motion is launched. The Controls parameters must be transmitted to DSPMC manually by pressing 'Update DSPMC' button before PID is armed. Clicking on OK or the 'Save Configuration' buttons saves the entire configuration to the selected Mach3 profile.

7.12.1 PID Filter

These values define the co-efficient of PID filters for the selected axis. See Section 6.2 PID Filter Configuration for definition of these parameters. The PID filter runs at 5KHz for each axis.

More information on PID control is available at http://en.wikipedia.org/wiki/PID_control.

7.12.2 Test Motion

Test Motion options can be used by users to tune PID filter and configure Control parameters.

The Ready LED shows if the DSPMC is ready to accept motion command. If Ready LED is GREEN, it implies DSPMC is ready to accept new motion command. While executing a motion profile, the Ready

LED turns to RED and DSPMC cannot accept a new motion command until the current motion sequence is complete or cancelled.

Once the test motion is complete, you can see how closely the axis followed the commanded motion profile on the PID Response graph. You can tweak the PID parameters and execute the test motion to verify the behavior. By selecting AutoReverse check box, you can make the axis reverse the direction automatically in next Execute command and thus avoid the axis to keep on going in one direction during testing.

Position – Test motion final position or displacement in terms of Position Units, e.g. 1.5, 10.093, mm or inches etc.

Acceleration – Test motion acceleration value in terms of Units per second squared, e.g. inches/second^2, mm/sec^2 etc.

Velocity – Test motion velocity value in terms of Units per minute, e.g. inches/minute, mm/minute etc.

Relative* and *Absolute – These check boxes indicate whether the value in the Position field is either the distance to travel (relative) or the final position (absolute).

Execute Button– Transmits Execute-Motion command to DSPMC. In addition, it also downloads PID Filter parameters before starting the motion. User can press 'Cancel' button to cancel the motion execution anytime during the machine operation. Make sure you have downloaded the axis controls setting by clicking "Update DSPMC" before clicking on Execute.

The Ready LED shows if the current motion command is completed and DSPMC is ready for new motion command. New motion command can be launched by Execute button when the Ready LED is Green. If the LED goes to Red after click on Execute, but you do not observe any motion, the velocity or acceleration may be too low.

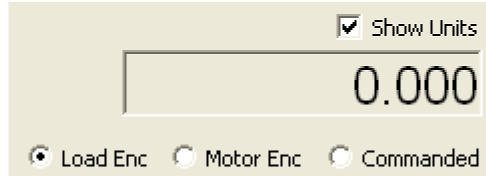
PID Arm Button – By clicking this button, the Plugin download's PID Filter parameters and arm or disarm the PID. If PID is armed, the LED below this button will turn to GREEN**,** otherwise it will be RED**.**

Home Button – Executes the Homing sequence based on selected Homing settings. Review section 7.12.3 Control Parameters to configure Mach3 Homing options for each axis before executing Homing.

Reverse - Checking this option will multiply parameter in the position box with -1 and thus direction of motion will be reversed.

Auto Reverse - Checking auto reverse option will toggle "reverse" option between two consecutive motion commands, thus the user do not have to manually reverse the direction of motion every time.

Axis Position Display (DRO) – Shows the position of the axis based on different settings as described below:



> **Show units -** When this option is selected, the data shown will be converted and shown in units (mm, inches etc), otherwise data will be displayed in raw encoder counts.
>
> **Commanded position -** Display shows the value of the internal variable for the commanded position for the selected axis.
>
> **Load Encoder -** Display shows the axis position derived from backlash count and selected feedback encoder.
>
> **Motor Encoder -** Display shows the current value of the axis position derived only from the encoder feedback.

Note that the actual position may slightly deviate from the Commanded position when PID is enabled.

7.12.3 Control Parameters

Control Input Source - Control Input Source defines the input type (or set-point) for the PID filter for a particular axis. This should be set to **MACHxx**. If the axis is not used, it must be disabled by selecting **undefined**.

Control Input Index - Defines the index of the PID input source. Normally this is equal to the axis number. For slave axis, it should be set to the number of the master axis.

Control Input Gain – The control input (Commanded) is multiplied by this number before applying to PID filter.

Control Output Source - Control Output Type defines the output for the PID filter for a particular axis. The possible values are:

> ***DAC:*** Use one of the analog outputs as the PID control output. This setting is used to drive a Servo amplifier that takes +/-10volt reference inputs.
>
> ***Stepper:*** Use one of the dedicated digital output pairs for the Step and Direction signals used in stepper drives. This feature is not currently available.
>
> ***Undefined:*** This setting is used to disable the axis and to ignore the control output index. If the axis is not used, the Control Output Source must be set to **undefined**.

Control Output Index - Defines the index of the PID Output.

PID Feedback Source - PID Feedback Source defines the feedback type for the PID filter for the selected axis. The possible values are:

> ***Encoder:*** Use one of the differential hardware encoder 0…7 as the PID feedback.
>
> ***A2D:*** Use one of the analog inputs as the PID feedback. This allows PID to be used for temperature and process control, in addition to motion control applications.

PID Feedback Index - Selects the index of the PID feedback source.

Homing Type - Defines homing sequence for each axis. Two types of homing sequence are supported:

1. **Home Sensor** (homing with or without Index Pulse)
2. **IndexPulseOnly** (Use only the Index pulse to Home)

For ***Home Sensor*** method, the axis moves in configured direction until home sensor is seen. It then moves in the opposite direction at 20% of initial speed until the sensor is not seen. At this point the home position is defined. If ***Use Index Pulse*** option is set, the axis then continues to move until Index pulse clears the position counter and indicate the home position.

For ***IndexPulseOnly***, the axis moves in the configured direction to locate the index pulse to home the axis. As soon as the index pulse is detected, it clears the position counter to indicate the home position and stops the axis.

Backlash Count – This field let you enter backlash in terms of encoder count. DSPMC uses this value to calculate virtual load position (mill table).

The following example shows how to calculate backlash counts:

Example Backlash = 0.010" on x axis
Encoder = 4000 counts per revolution (1000 count encoder)
Lead Screw = 10 revolution per inch

With 40000 counts per inch, the backlash in terms of encoder counts will be 40000 x 0.010 = 400

8. DSPMC Software Upgrade Tool

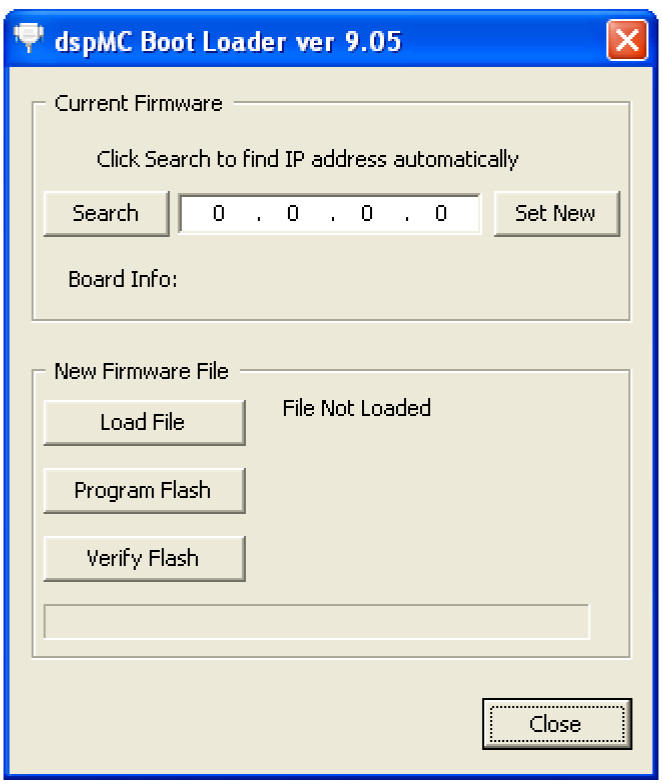
8.1 Upgrade the DSPMC Firmware

The Boot Loader software tool for DSPMC is used to re-program (Flash) the board software (Firmware). The latest Firmware file is available from the factory on request. The file is sent in compressed zip format, so you will need to unzip the file and extract the binary file before programming.

NOTE: Before programming the Firmware, close all programs, e.g. Mach3, that are communicating with the DSPMC board. This includes programs running on other computers that are connected to the DSPMC board over Ethernet. Also make sure the Servo Drives are powered down.

The following steps describe the procedure to upgrade the firmware.

1. Double Click the 'DSPMC Firmware Upgrade' icon that was installed during software installation. The following window appears:



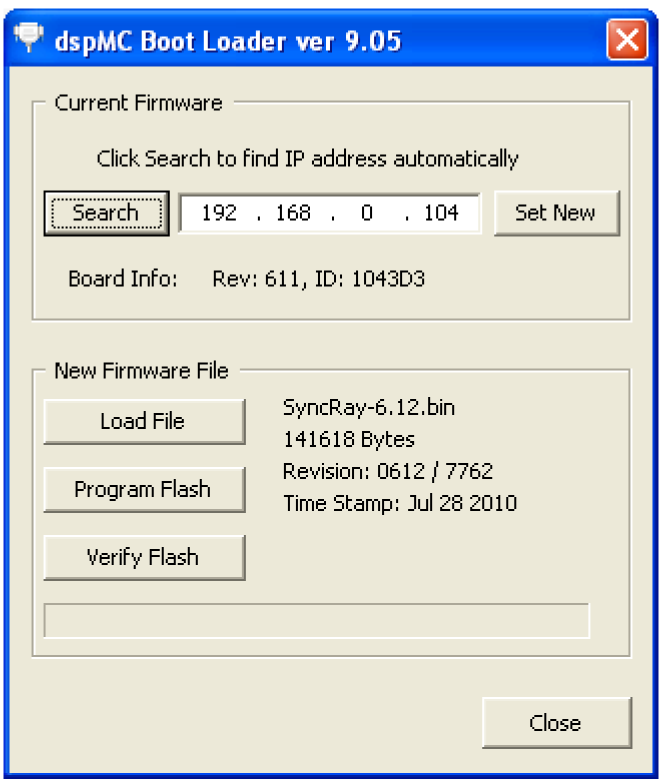
2. Click on the Search DSPMC button to setup the connection.

This window shows that the connection is correctly established. Click OK. If the connection cannot be established, please refer to Network Connection Setup section to setup communications with the device.

Once connected, the main window shows the current board information, ie, the firmware revision and the board serial ID. If the firmware file you received from factory has the same revision number, then you do not need to re-program the board.

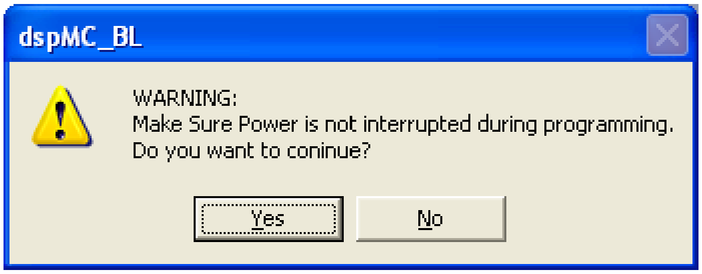
3. Click on the Load File button and search for the firmware (.bin) file that you received from factory.

Once the file is selected, a sample would look like this:

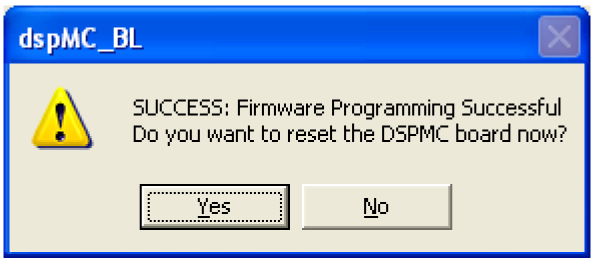


The version number and time-stamp of the firmware file you opened is shown on the main window. Please make sure this is the correct version you intend to flash.

4. Click on the Program Flash Button. Please make sure that the board power supply does not get interrupted during this process. Click on the yes button to start. If you are not sure, click 'No' and provide a stable power supply before you try programming again.



Once you click 'Yes', the software will start programming the board. The progress bar will show the programming progress. After the programming is done, the software will verify if the programming is successful. A 'SUCCESS' message is displayed if firmware programming and verification is successful as shown below.



The DSPMC unit must be restarted after a successful programming. You can automatically reboot the DSPMC by clicking on Yes. In case the Reboot command is not accepted, the unit must be restarted manually in order to execute the newly flashed program. This can be accomplished by turning off the power, and then turning back on again after 10 seconds.

If an error is detected during programming, a message box will show the location of the first data mismatch. You can try programming again by clicking on Program Flash Button to correct the error.

Note:
Note: If an error is detected during programming or verification, you can retry programming as many times as you like, as long as you do not turn off power to the DSPMC board. Once the board power is turned off, it will try to load the new firmware the next time you power up.

8.2 Changing the DSPMC Default IP Address

Once the device is found by clicking on the Search button, you can enter a new IP address in the address field and click on 'Set New'. The new address will be saved in the device and will take effect after you cycle the power on the unit. Most users should not need to change the default IP address.

Appendix A – DSPMC.xml Format

XML format of DSPMC configuration is shown below. Only axis 0 is shown. All other axes follow the same format. At the end of the listing, detailed description of each parameter is described.

> ***Note:***
> **Mach3 users do not need to edit this file. With Mach3 DSPMC Plugin Release 3, the dspMC.xml file is not required by the plugin.**

```
<?xml version="1.0" encoding="utf-8"?>
<!-- This file represents configuration database for DSPMC Motion Controller -->

<dspMCConfig>

        <EncoderPolarity>                       <!-- 1 = as is, -1 = reversed  -->
                <Channel_0> 1 </Channel_0>
                <Channel_1> 1 </Channel_1>
                <Channel_2> 1 </Channel_2>
                <Channel_3> 1 </Channel_3>
                <Channel_4> 1 </Channel_4>
                <Channel_5> 1 </Channel_5>
                <Channel_6> 1 </Channel_6>
                <Channel_7> 1 </Channel_7>
        </EncoderPolarity>

        <HomingIgnoreLimits>0</HomingIgnoreLimits>
                <!-- Options:    1 = Ignore Limit switches,
                                 0 = Do not Ignore Limit Switches during homing -->

        <Mach3Config>

                <DebugWindow> 1 </DebugWindow> <!-- 1 = Turn on debug window, 0 = Turn Off -->

                <Spindle>
                        <DAC>3</DAC>
                </Spindle>

                <MaxBufferLevel> 50 </MaxBufferLevel> <!-- upto 100% for 4096 points/axis -->

                <MpgSource1>
                        <Type>SoftEncoder</Type>  <!-- Options: SoftEncoder, HardEncoder-->
                        <Index>1</Index>
                </MpgSource1>

                <Threading>
                        <RPM_Sync_Source> HardEncoder </RPM_Sync_Source>
                                <!-- HardEncoder or DigitalInput -->
                        <RPM_Sync_Index> 3 </RPM_Sync_Index>
```

```
                <RPM_Count_Per_Rev> 2000 </RPM_Count_Per_Rev>
                <RPM_Sampling_Milli_Second> 50 </RPM_Sampling_Milli_Second>
                        <!-- 1 ms thru 10000 ms -->
        </Threading>

        <Probing>

                <ProbingType> DualCycle </ProbingType>
                        <!-- SingleCycle or DualCycle (coarse and fine moves) -->

                <ProbingMoveBack> 0.35 </ProbingMoveBack>
                        <!-- Distance in mm/inch to move back from touch point before starting
                        the second (fine) move -->

                <ProbingFeedRate> 15.5 </ProbingFeedRate>
                        <!-- Feedrate for the second cycle (fine move) in mm or inch per minute

        </Probing>

</Mach3Config>

<Axis0>
        <PID_Filter>
                <P>25000</P>
                <I>200</I>
                <D>1000000</D>
                <ffcGain>0</ffcGain>          <!-- Feed Forward Command Gain -->
                <ffcdGain>0</ffcdGain>        <!-- Feed Forward Command Derivative Gain -->
                <MaxError>0</MaxError>        <!-- Max Limit on Error (Command - Feedback) -->

                <MaxError_I>500</MaxError_I> <!-- Max Limit on Integral Error -->

                <MaxError_D>0</MaxError_D> <!-- Max Limit on Differential Error -->

                <MaxError_F>6000</MaxError_F> <!-- Max Limit on Following Error -->

                <Deadband>0</Deadband>  <!-- Range of Error where PID Loop
                                           does not respond -->
                <MaxCmd_D>0</MaxCmd_D>  <!-- Max Limit on Command Derivative -->

                <Scale>1500</Scale>           <!-- Divisor for PID Calculation -->

                <Offset>50</Offset>           <!-- Constant to be added after PID calculation
                                              and scalling -->
        </PID_Filter>


        <ControlInput> <!-- Source: TrajectoryPlanner, CommandPosition, Counter, A2D -->

                <Source>TrajectoryPlanner</Source>
                <Index>0</Index>
```

```
            <Gain>1.000000</Gain>
        </ControlInput>


        <PID_Feedback>
            <Source>Counter</Source>   <!-- Feedback: Counter, A2D -->
            <Index>0</Index>
            <Gain>1.000000</Gain>
        </PID_Feedback>

        <ControlOutput>

            <Type>DAC</Type>    <!-- Output: undefined,DAC, Stepper -->
            <Index>0</Index>
        </ControlOutput>

        <AmpEnableOutput>-1</AmpEnableOutput>
            <!-- Options: 0..31 output number, undefined = -1 -->

        <LimitSenseLow>-1</LimitSenseLow>
            <!-- Options: 0..63 input number, undefined = -1 -->

        <LimitSenseHigh>-1</LimitSenseHigh>
            <!-- Options: 0..63 input number, undefined = -1 -->

        <LimitSensePolarity>1</LimitSensePolarity>
            <!-- Valid numbers: 1, -1 -->

        <Homing>
            <Type>HomeSensor</Type>
                <!-- Options: HomeSensor, IndexPulseOnly -->
            <UseIndexPulse>1</UseIndexPulse>
                <!-- Options: 1 = use index pulse, 0 = donot use index pulse -->
            <SensorInput>-1</SensorInput>
                <!-- Options: 0..63 input number, undefined = -1 -->
            <SensorPolarity>1</SensorPolarity>
                <!-- Valid numbers: 1, -1 -->
        </Homing>
        <CountsPerUnit>40000</CountsPerUnit>  <!—Number of Encoder Count per unit -->
        <BackLashCounts>0</BackLashCounts> <!— Encoder Counts during backlash -->
    </Axis0>
```


Description of XML parameters:

A.1 Spindle DAC

The Spindle DAC configuration selects the analog output (DAC) for the spindle speed control. The DAC output voltage varies from 0 to 10volts. Any Analog output position can be selected from is 0 to 7.

Make sure the selected DAC output number is not used anywhere else in the dspmc.xml file. If it is used, then enter **“Undefined”** in the ***ControlOutput Type*** section.

The following section defines DAC output 6 for spindle.

```
<Spindle>
    <DAC>6</DAC>
</Spindle>
```

A.2 MaxBufferLevel

This section defines how much command position buffering will be done inside the DSPMC controller. The total size of the buffer is 4096 points per axis. These points are consumed by the DSPMC at 1 KHZ. To get faster response time on feedrate changes, you may select a lower value, but the side effect is that if the PC software slows down and cannot sustain the motion data rate to the DSPMC, then the motion could be jerky.

The valid rage for this section is 1…100 percent.

eg,

```
<MaxBufferLevel> 50 </MaxBufferLevel>
```

A.3 BackLashCounts

The following example shows how to calculate backlash counts for x axis:

Example Backlash = 0.010" on x axis
Encoder = 4000 counts per revolution (1000 count encoder)
Lead Screw = 10 revolution per inch
Axis: Z (2)

With 40000 counts per inch, the backlash in terms of encoder counts will be 40000 x 0.010 = 400

Therefore in the dspmc.xml you will enter 400 for the selected axis as follows:

```
<Axis0>
…..
<BackLashCounts>400 </BackLashCounts>
…..
</Axis0>
```

A.4 RPM

This section is ignored by DSPMC.

A.5 PID Filter

These values define the co-efficient of PID filters for each axis. See Section 6.2 PID Filter Configuration for definition of these parameters. The PID filter is run at 5KHz for each axis.

More information on PID control is available at http://en.wikipedia.org/wiki/PID_control.

A.6 ControlInput Source

ControlInput Source defines the input type (or set-point) for the PID filter for a particular axis. The possible values are:

CommandPosition: This setting is used for PC applications that generate command position itself, e.g. Mach3. The DSPMC makes the axis follow these command positions. The command position data is provided to DSPMC at a rate of 1KHz for each axis.

TrajectoryPlanner: The DSPMC controller also has built in trajectory generator (or planner) for each axis. This allows motion profiles to be generated automatically, based on user provided distance, accel, and velocity settings.

Counter: Use a shaft encoder counter as the PID input.

A2D: Use one of the analog inputs as the PID input. This allows PID to be used for temperature and process control, in addition to motion control applications.

A.7 ControlInput Index

Defines the index of the PID input source. Normally this is equal to the axis number. For slave application, this defines the data index of the master.

A.8 ControlInput Gain

Defines the multiplier before applying the data to the PID input. E.g. if Analog Input 0 is the source, analog input 0 value is first multiplied by the Gain and then used in the PID calculation.

A.9 PID Feedback Source

PID Feedback Source defines the feedback type for the PID filter for a particular axis. The possible values are:

Counter: Use a shaft encoder counter as the PID feedback.

A2D: Use one of the analog inputs as the PID feedback. This allows PID to be used for temperature and process control, in addition to motion control applications.

A.10 PID Feedback Index

Defines the index of the PID feedback source.

A.11 PID Feedback Gain

Defines the multiplier before applying the data to the PID feedback. E.g. if Encoder Counter 0 is the feedback source, Encoder Counter 0 value is first multiplied by the Gain and then used in the PID calculation.

A.12 ControlOutput Type

ControlOutput Type defines the output for the PID filter for a particular axis. The possible values are:

> ***DAC:*** Use one of the analog outputs as the PID control output. This setting is used to drive Servo amplifiers that takes +/-10volt reference inputs.
>
> ***Stepper:*** Use one of the dedicated digital output pairs for the Step and Direction signals used in stepper drives.
>
> ***Undefined:*** This setting is used when the axis is disabled and the index is used on some other axis. For example, if DAC 6 is used for Spindle speed and DAC 6 is also used as ControlOuput for the axis, but the axis is not part of the system, then the user must define the ControlOutput type as undefined. If the axis is part of the system, then some other index should be used for Spindle or axis output.

A.13 ControlOutput Index

Defines the index of the PID Output type.

A.14 AmpEnableOutput

Defines the index of the digital output to be used as the enable signal for the output servo amplifier or drive. Valid range is 0 thru 31, and -1. A value of -1 means no digital output is used as Enable signal for this axis. Value of 0..31 defines the digital output index.

A.15 Homing

This section defines homing sequence for each axis. Two types of homing sequence are supported:

- **HomeSensor** (homing with or without Index Pulse)
- **IndexPulseOnly** (Use only the Index pulse to Home)

For *HomeSensor* method, the axis moves in configured direction until home sensor is seen. It then moves in the opposite direction at 20% of initial speed until the sensor is not seen. At this point the home position is defined. If *UseIndexPulse* option is set to 1, the axis then continue to move until Index pulse clears the position counter and indicate the home position.

For *IndexPulseOnly*, the axis moves in the configured direction to locate the index pulse to home the axis. As soon as the index pulse is detected, it clears the position counter to indicate the home position and stops the axis.

A.16 MPG Settings – MpgSource1

This section defines MPG (Manual Pulse Generation) Quadrature encoder source. Both Differential and Single Ended Encoder types are supported.   Differential encoder can be hooked up to any of the six encoder channel available on connectors J6 and J7. These encoders are defined as ***HardEncoder*** in the xml file. Single Ended encoders (defined as ***SoftEncoder***) can be hooked up to the Digital Inputs available on Connector J5.
The pin assignments for available *SoftEncoder* are as follows (***Require Firmware 63 or newer***):

SoftEncoder 0 : A On J4 Pin 24,
B On J4 Pin 12

SoftEncoder 1 : A On J4 Pin 25
B On J4 Pin 13
SoftEncoder 2 : A On J5 Pin 18
B On J5 Pin 6

Example 1: Connect a differential encoder for MPG on Encoder Channel 5 (J7)

```
<MpgSource1>
    <Type> HardEncoder </Type>
    <Index>5</Index>
</MpgSource1>
```

Example 2: Connect a single-ended encoder for MPG on J5 Digital Inputs, pin 18 and 6.

```
<MpgSource1>
    <Type> SoftEncoder </Type>
    <Index>2</Index>
</MpgSource1>
```

A.17 Threading

This section defines parameter for CNC threading. The following is an example configuration:

```
<Threading>
    <RPM_Sync_Source> HardEncoder </RPM_Sync_Source>
            <!-- HardEncoder or DigitalInput -->

    <RPM_Sync_Index> 1 </RPM_Sync_Index>

    <RPM_Count_Per_Rev> 2000 </RPM_Count_Per_Rev>

    <RPM_Sampling_Milli_Second> 100 </RPM_Sampling_Milli_Second>
            <!-- 1ms thru 10000ms-->
</Threading>
```

A.17.1 RPM_Sync_Source This parameter defines the encoder type for Spindle speed calculation and starting the threading cycle. The Index pulse from the encoder is used to launch the Z-Axis at the right time in order to position the tool correctly for Threading in every cycle. The RPM calculation is used to override the feedrate of the Z-Axis during the threading cycle.

The two possible values for **RPM_Sync_Source** parameter are: "***HardEncoder***" and "***DigitalInput***".

1. When ***HardEncoder*** is selected, the spindle feedback encoder must be connected to one of the encoder inputs on J6 and J7 connectors. The encoder's differential A and B signals are used to calculate the RPM of the spindle, and Index pulse is used to trigger the threading cycle.

2. When ***DigitalInput*** is selected, the spindle feedback is generated by a single line pulse train. The pulse train is used to calculate the spindle RPM as well as used for Sync pulse to launch the threading cycle.
There are two dedicated inputs on J5 for the spindle pulse train. These are also called I/O Toggle Counters.

    - I/O Toggle Counter 0: Digital Input 18 (J5 Pin 19)

- I/O Toggle Counter 1: Digital Input 19 (J5 Pin 7)

A.17.2 RPM_Sync_Index This parameter defines the encoder index for Spindle speed feedback. Below is the range for this index:
HardEncoder: index range is 0...5.
DigitalInput: index range is 0...1.

A.17.3 RPM_Count_Per_Rev This parameter defines the encoder resolution in terms of count per revolution for Spindle speed feedback. For ***HardEncoder*** type encoder, the encoder resolution must is multiplied by 4. No multiplication is done when ***DigitalInput*** is selected.

A.17.4 RPM_Sampling_Milli_Second This parameter defines the timing window in milliseconds to add the encoder counts for RPM calculation. For slow pulse train (eg only few ticks per rev), this value should be high enough to accumulate enough counts to calculate RPM consistently. If the window time is too long, the system reaction time (regulation of Z-Axis feedrate) to changing RPM will be slow. A higher count/rev encoder will allow this window time to be very small, which will allow the system to react fast (regulate Z-Axis feedrate) if RPM changes. The range of this field is from 1 thru 10000 milliseconds.

A.18 Probing

This section defines parameter for CNC Probing feature. For the probing cycle, the axis, probing feedrate, and the probe switch are set by the PC software, eg Mach3.

The following is an example configuration:

```
<Probing>
        <ProbingType> DualCycle </ProbingType>
                <!-- SingleCycle or DualCycle (coarse and fine moves) -->

        <ProbingMoveBack> 0.35 </ProbingMoveBack>
                <!-- Distance in mm/inch to move back from touch point before
                                                starting the second (fine) move -->

        <ProbingFeedRate> 15.5 </ProbingFeedRate>
                <!-- Feedrate for the second cycle (fine move) in mm or inch per minute -->
</Probing>
```

A.18.1 ProbingType This parameter defines the probing method:

> **SingleCycle**: Axis starts the probing move. As soon as the probe switch is on, the current position is captured and the probing sequence is complete.
>
> **DualCycle:** Axis starts a first probing move (called coarse move). As soon as the probe switch is on, the axis stops, and backs off distance specified in the **ProbingMoveBack** parameter. The axis then starts the second move, called fine move. The direction is same as the coarse move. The feedrate for the fine move is defined by the third parameter **ProbingFeedRate.**

A.18.2 ProbingMoveBack This parameter defines the distance to move back to start the fine move. It is applicable only in the DualCycle mode.

A.18.3 ProbingFeedRate This parameter defines the feedrate for the fine move. It is applicable only in the DualCycle mode.

A.19 Using Limit Switches as Home Sensors

When Homing using Limit switches, the ***HomingIgnoreLimits*** parameter should be set to 1 as follows:

< HomingIgnoreLimits > 1 </ HomingIgnoreLimits >

For Home sensors using dedicated inputs, this parameter should be set to 0, or removed from the xml file.

A.20 Hardware Encoder Counter Polarity

The ***EncoderPolarity*** parameter is used to reverse the direction of the encoder counters. If A/B signals are connected such that it does not match the PID control direction, the system will not be able to arm. To fix this issue, the hardware A and B signals can be reversed, or simply use this section to change the polarity in the software.

```
<EncoderPolarity>                <!-- 1 = as is, -1 = reversed  -->
    <Channel_0> 1 </Channel_0>
    <Channel_1> -1 </Channel_1>
    <Channel_2> 1 </Channel_2>
    <Channel_3> 1 </Channel_3>
    <Channel_4> 1 </Channel_4>
    <Channel_5> 1 </Channel_5>
    <Channel_6> 1 </Channel_6>
    <Channel_7> 1 </Channel_7>
</EncoderPolarity>
```

Note that this software encoder polarity setting only applies to the encoder counter. The Index pulse signal polarity is not affected by this setting.

License Agreement

Before using the DSPMC and accompanying software tools, please take a moment to go thru this License agreement. Any use of this hardware and software indicate your acceptance to this agreement.

It is the nature of all machine tools that they are dangerous devices. In order to be permitted to use DSPMC on any machine you must agree to the following license:

I agree that no-one other than the owner of this machine, will, under any circumstances be responsible, for the operation, safety, and use of this machine. I agree there is no situation under which I would consider Vital Systems, or any of its distributors to be responsible for any losses, damages, or other misfortunes suffered through the use of the DSPMC board and its software. I understand that the DSPMC board is very complex, and though the engineers make every effort to achieve a bug free environment, that I will hold no-one other than myself responsible for mistakes, errors, material loss, personal damages, secondary damages, faults or errors of any kind, caused by any circumstance, any bugs, or any undesired response by the board and its software while running my machine or device.

I fully accept all responsibility for the operation of this machine while under the control of DSPMC, and for its operation by others who may use the machine. It is my responsibility to warn any others who may operate any device under the control of DSPMC board of the limitations so imposed.

I fully accept the above statements, and I will comply at all times with standard operating procedures and safety requirements pertinent to my area or country, and will endeavor to ensure the safety of all operators, as well as anyone near or in the area of my machine.