

Galil PlugIn for Mach4

Table of Contents

Overview:	2
Controller Compatibility:	3
Ethernet Controllers:	3
Installing the PlugIn:	4
Prerequisites:	4
Configuring the PlugIn:	5
Ethernet Controller Configuration:	6
General Tab, Ethernet:	6
PCI Controller Configuration:	7
General Tab, PCI:	7
Common Controller Configuration:	8
Motor Config Tab	10
Extended I/O Tab	12
Probing:	13
Credits:	14



Galil PlugIn for Mach4



Overview:

The Galil PlugIn controls the popular Galil Motion Controllers. The Galil is a very powerful motion controller that will control up to 8 axes and provides I/O capabilities as well. When used with Mach, it provides a means of controlling a closed loop servo system as well as a stepper system. Stepper motor and servos can even be mixed.

For servos, the Galil controllers provide an industry standard $\pm 10V$ motor command that works with most analogue servo amps. It is one of the few ways to run analogue amps with Mach.

For stepper motors, the Galil Controllers provide the standard pulse and direction interface. Unlike the parallel port stepper driver, the Galil controller can provide step pulse rates at up to 3 MHz.

The PlugIn works by taking the Mach step and direction signals that would normally go to the parallel ports and converting them into linear interpolated moves in the Galil command language. A spindle is also supported.

Galil axes can be mapped 1 to 1 with the Mach axes or mixed in any manner desired. Limit switches, home switches, general inputs, and general outputs are supported through the Mach "Ports and Pins" settings. And support for extended I/O is also provided.

Galil PlugIn for Mach4

Controller Compatibility:

Accelera Series

- DMC-40x0 Ethernet
- DMC-18x6 PCI BUS

Econo Series

- DMC-41x3 Ethernet

Rio Series

- RIO-471xx
- RIO-4720x
- RIO-47300

The Optima DMC-21x0 controllers may work, but none have been tested.

Ethernet Controllers:

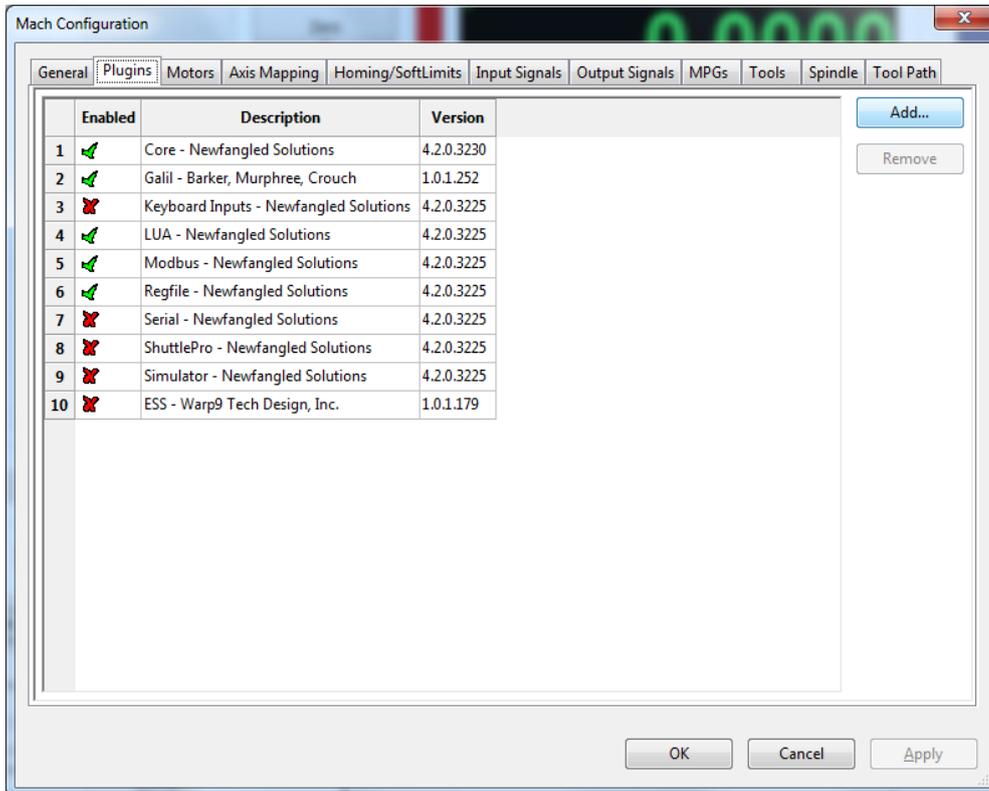
Galil Ethernet controllers have a few special items that need some consideration.

A good quality network is required. The Galil Controllers will answer just about any TCP/IP port and will open a “handle” when contacted by any remote machine. There is a limit of 8 handles, which can be used up rather quickly making your Galil controller unusable until it is reset. Obviously, a port sniffer running on the network will pretty much make a Galil Ethernet controller hose up pretty fast. So it is best to have the Galil controller on an isolated network along with your Mach machine.

Galil PlugIn for Mach4

Installing the PlugIn:

The PlugIn comes as an m4plug{x}. Where {x} is either w for Windows, l for Linux, or m for Mac. The plugin can be installed from the Mach 4 configuration dialog Plugins tab by clicking the "Add" button.



Prerequisites:

While the setup and configuration of the Galil controller is beyond the scope of this manual, a few things are worth mentioning.

If you are running servos, the PID filter values should be set in the controller prior to running the Galil PlugIn. This is known as tuning the servos. The full version of the Galil Suite software from Galil has a compliment of servo tuning tools that can be a big time saver. Manually tuning the servos is also possible via the KP, KI, and KD commands through a Galil terminal session.

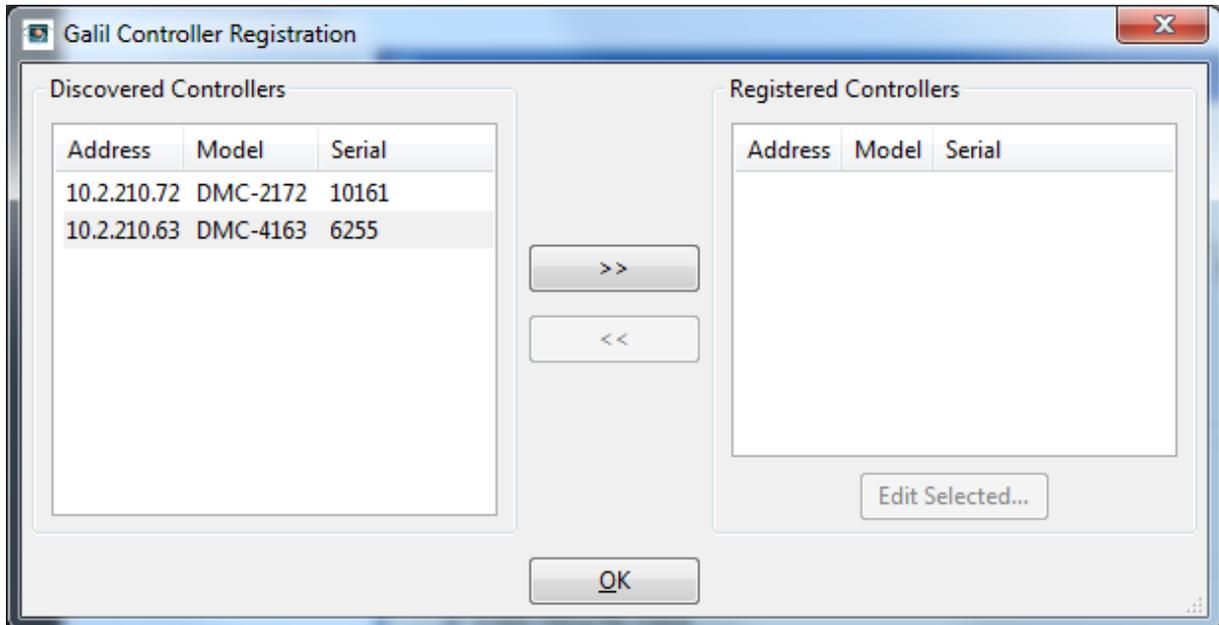
After tuning the servos with whatever method you choose, the KP, KI, and KD values should be saved to the controller with the BN command.

All home and limit switches should be connected and verified. You can use the Galil Suite software "Viewer" window to verify these switches.

Galil PlugIn for Mach4

Configuring the PlugIn:

Once the plugin is installed, it must be configured. With Mach in the disabled state, follow the menu path "Configure->Plugins->Galil". The Galil Controller Registration dialog will be shown with a list of discovered controllers.



The Galil plugin for Mach 4 can support multiple Galil controllers. The first registered controller is considered to be the master controller. It will be the only controller that will accept Mach movement commands. Any additional motion controllers can be registered after to provide additional I/O or movement capabilities via on board Galil programs. Rio controllers are supported as well.

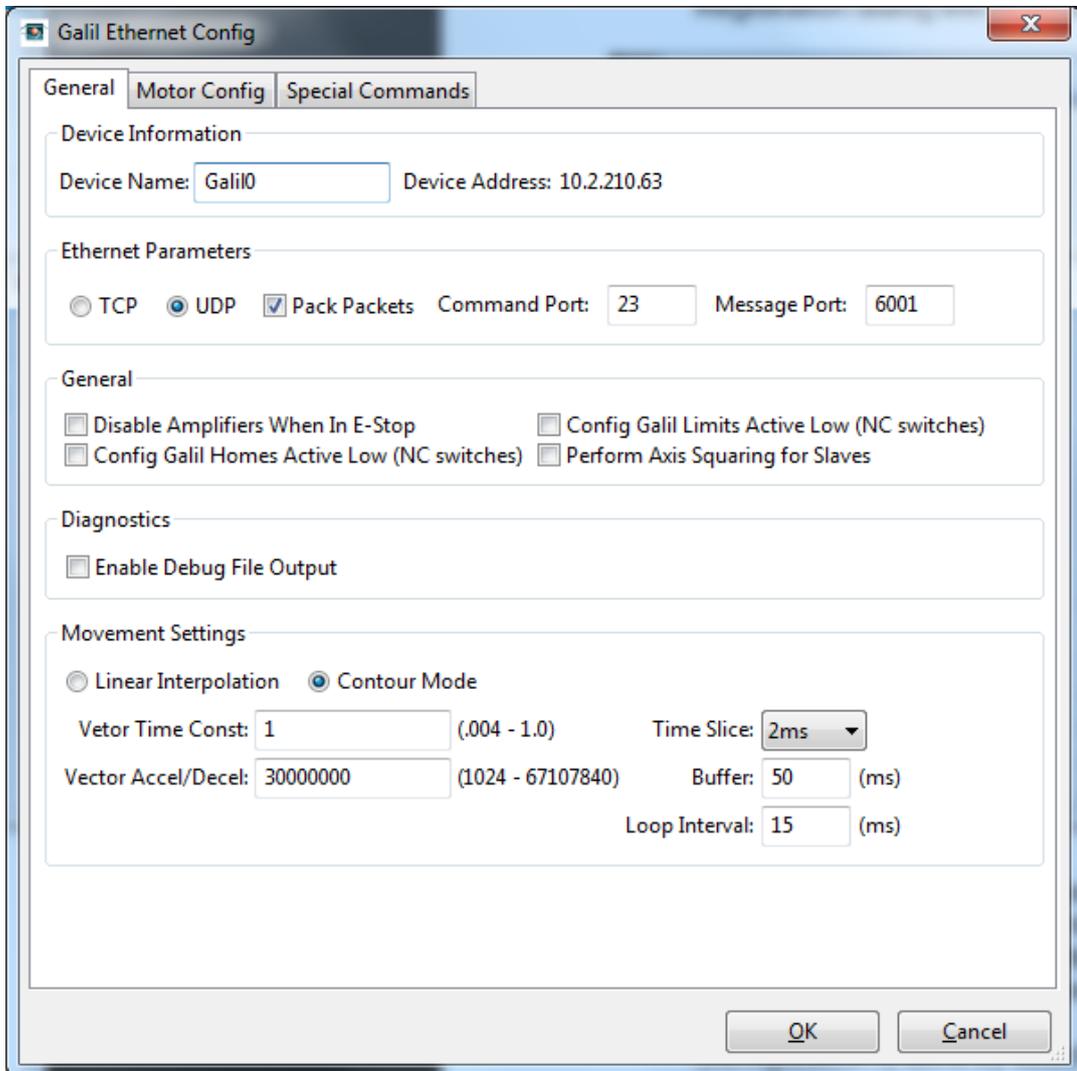
The list of discovered controllers is in the left hand pane. Select the desired controller and move it to the list of Registered Controllers in the right hand pane by pressing the ">>" button. Again, the first controller registered should be the master motion controller.

Configuration is done by selecting a registered controller and clicking the "Edit Selected..." button.

Galil PlugIn for Mach4

Ethernet Controller Configuration:

If an Ethernet controller is selected, you will be presented with a dialog window similar to the one below.



General Tab, Ethernet:

1. Device Information:

1.1. **Device Name:** When configuring a new controller, you must provide a device name. This is the name by which Mach will refer to this controller.

1.2. **Device Address:** The Ethernet address of the controller will be shown.

2. Ethernet Parameters:

2.1. **TCP:** Use the TCP protocol for communications.

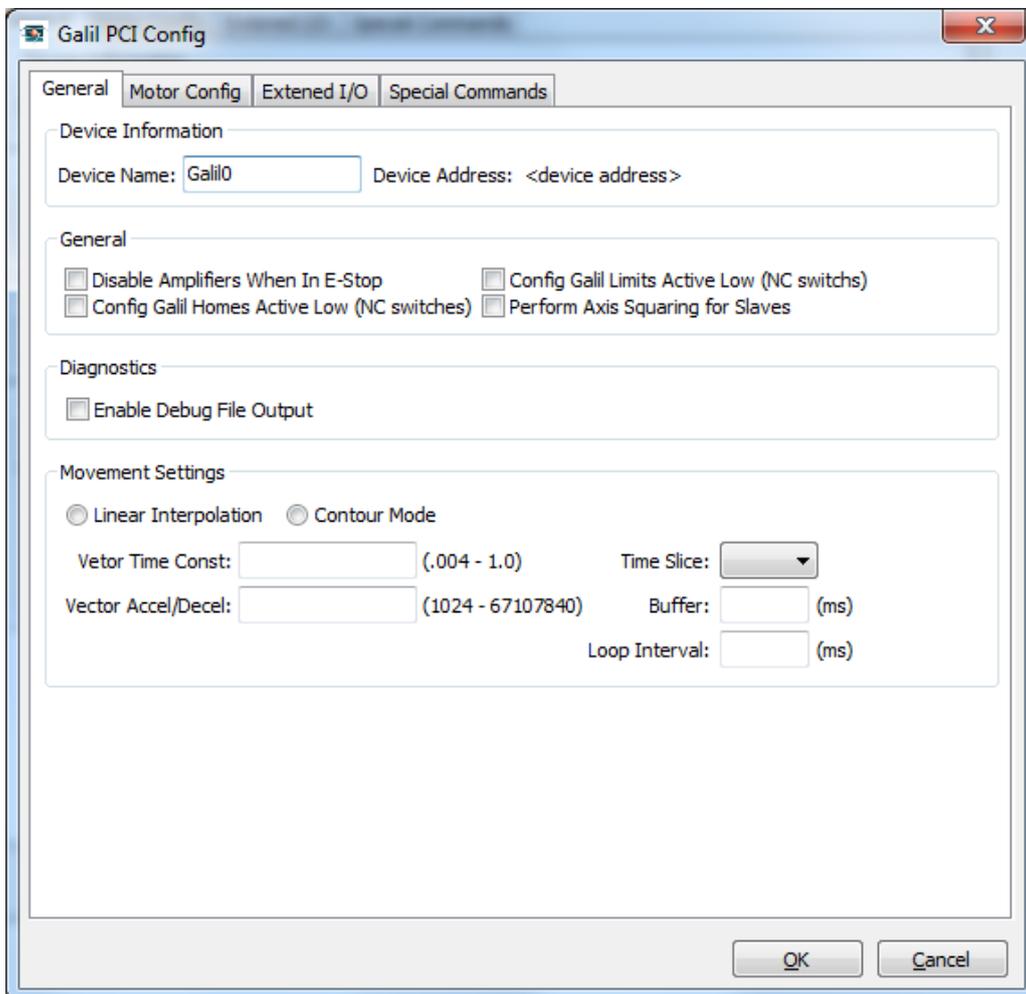
2.2. **UDP:** Use the UDP protocol for communications (faster).

Galil PlugIn for Mach4

- 2.3. **Pack Packets:** Insert multiple Galil commands into one protocol packet. Choose this if you are using UDP and will be running a granular time slice for best performance.
- 2.4. **Command Port:** The desired communications port (23 is the default).
- 2.5. **Message Port:** The desired port to use for unsolicited messages (6001 is the default).

PCI Controller Configuration:

If an Ethernet controller is selected, you will be presented with a dialog window similar to the one below.



General Tab, PCI:

1. Device Information:

- 1.1. **Device Name:** When configuring a new controller, you must provide a device name. This is the name by which Mach will refer to this controller.
- 1.2. **Device Address:** The PCI address of the controller will be shown.

Galil PlugIn for Mach4

Common Controller Configuration:

1. General:

1.1. **Disable Amplifiers When in E-Stop:** Disables the amplifiers when Mach is in an e-stop condition.

1.2. **Config Galil Limits Active Low:** If you have normally closed switches on your limits, then this checkbox should be enabled.

The Galil lets you control the active state of the limit switches. This allows for both normally open and normally closed switches to be used. It is popular to wire the limit switches with normally closed switches for safety reasons. In the event of a cable cut, the machine will stop. A normally closed switched system will be active low. A normally open switched system will be active high.

Unfortunately, you cannot have a mixture of normally closed and normally open limit switches. And you cannot have a mixture of switch types for the homes switches either. You must choose one switch type for the limits and one switch type for the home switches. However, you could mix switch types if you wire them with relays that do the right thing on the machine. The important thing is that the switch state that the Galil sees must be consistent.

1.3. **Config Galil Homes Active Low:** Similarly, if your home switches are active low, enable this checkbox.

1.4. **Perform Axis Squaring for Slaves:** Gantry style machines that have a slaved axis require special homing routines. The default way the plugin homes a gantry axis is that it first homes the master and slave axes together. It will then disconnect the slave and home it to square the gantry. If you do not wish to square the gantry, then uncheck this checkbox.

2. Diagnostics:

2.1. **Enable Debug File Output:** Enabling this checkbox creates a GalilDebug-xxxxx.txt file in the PlugIn's directory where xxxxx is the serial number of the controller. This output may be useful for the developers if you are running into problems. Leave it unchecked for normal operation as it does slow the system down with lots of disk I/O.

Galil PlugIn for Mach4

3. Movement Settings:

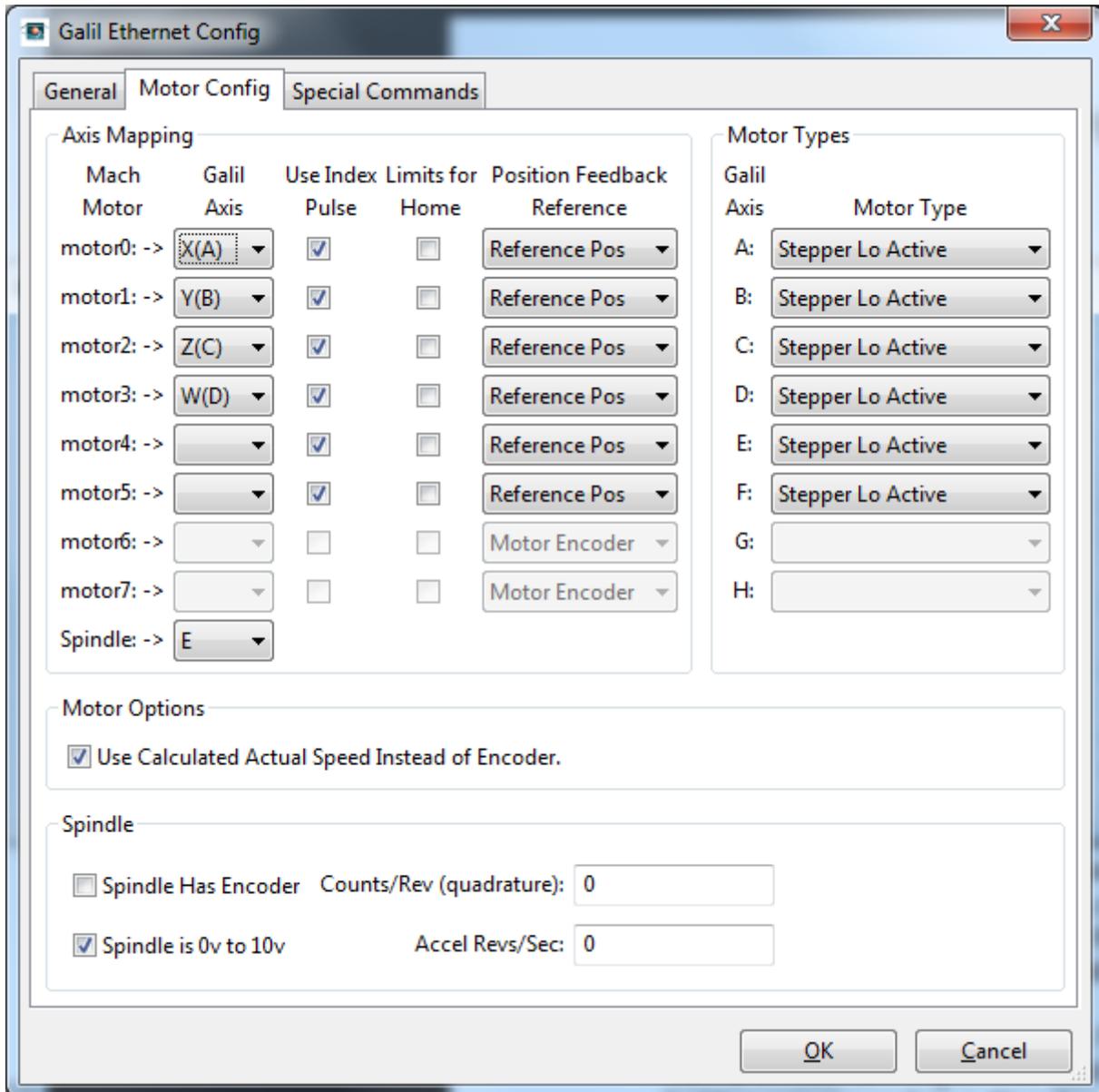
- 3.1. **Linear Interpolation:** The PlugIn is capable of using two modes of motion; linear interpolation and contour mode. Linear Interpolation works with all Galil controllers and thus, is the default.
- 3.2. **Contour Mode:** If have an Accelera controller, you will want to use this mode of motion as it is superior to Linear Interpolation with Mach.
- 3.3. **Vector Time Constant:** (Linear Interpolation mode only) For most situations, set the Vector Time Constant to 1. Setting this to a lower value will have the effect of smoothing the moves. The Vector Time Constant is a velocity filter. Setting the value progressively lower increases the filtering to attempt to maintain a more constant velocity over the move sequences.
- 3.4. **Vector Deceleration:** (Linear Interpolation mode only) The Galil is a very powerful motion controller in its own right and it is capable of performing its own trajectory planning. But Mach is doing the same thing! So the Vector Deceleration should be set to a range that allows Mach to take the lead in this department. Start with 30,000,000. That's thirty million!
- 3.5. **Time Slice:** This is the desired movement time slice. The lower the value, the more granular the system is. However, it also affects how much data is streamed to the controller. For example, a 2ms time slice sends twice as much data to the controller than a 4ms time slice. This value becomes important if the machine has a large working envelope and has to move at high speeds. A 12' x 4' router moving at 1000 inches per minute will generally need a smaller time slice over a 24" x 14" mill moving at 500 inches per minute to prevent faceting of large radiuses.

The available values for the time slice are determined by the type of controller being used. It is important to note that all PCs may not operate at the lowest time slice available to any particular controller. This is because the PC and Ethernet card may not be capable of supplying the Galil at the desired rate! For optimal performance, a modern PC with a server style Ethernet card should be used. If the PC/Ethernet card combination will not run the Galil smoothly, the only option is to increase the time slice value until the system runs smoothly.

- 3.6. **Buffer:** This is the amount of movement time that is buffered on the Galil. This setting is also dependent on the Mach PC and its' communications abilities. Generally, you will want a value here that is low enough to make things like feed holds seem instantaneous. Anything below 100 ms is acceptable to the majority of people. It is to be configured in conjunction with the Loop Interval parameter below. A general value is 3 times the Loop Interval.
- 3.7. **Loop Interval:** This is the interval at which the plugin tries to send data to the controller. A lower value here is means that the plugin will send data to the controller more often. It is also the interval in which the Galil will send feedback data to the plugin. 20 ms is a general good value. If 20 ms is chosen, a good buffer value would be 60 ms or higher. If 25 ms is chosen, a Buffer value of 75 ms or higher would be appropriate and so on.

Galil PlugIn for Mach4

Motor Config Tab



1. **Motor Mapping:** You can map motors in a different order than the standard 01234567 order that Mach provides. For instance, you may wish to drive your Mach motor 0 axis with the Galil G axis. Simply choose the Galil G axis in the dropdown list for the Mach motor 0.

The limit switches travel with the Galil axis, so in the example above, you would wire your limits to the Galil G axis but they would appear as Mach motor0 signals.

1.1. Mach Motor: This is how mach will refer to the mapped Galil axis.

1.2. Galil Axis: This list will be populated with the available Galil axes on the controller.

Galil PlugIn for Mach4

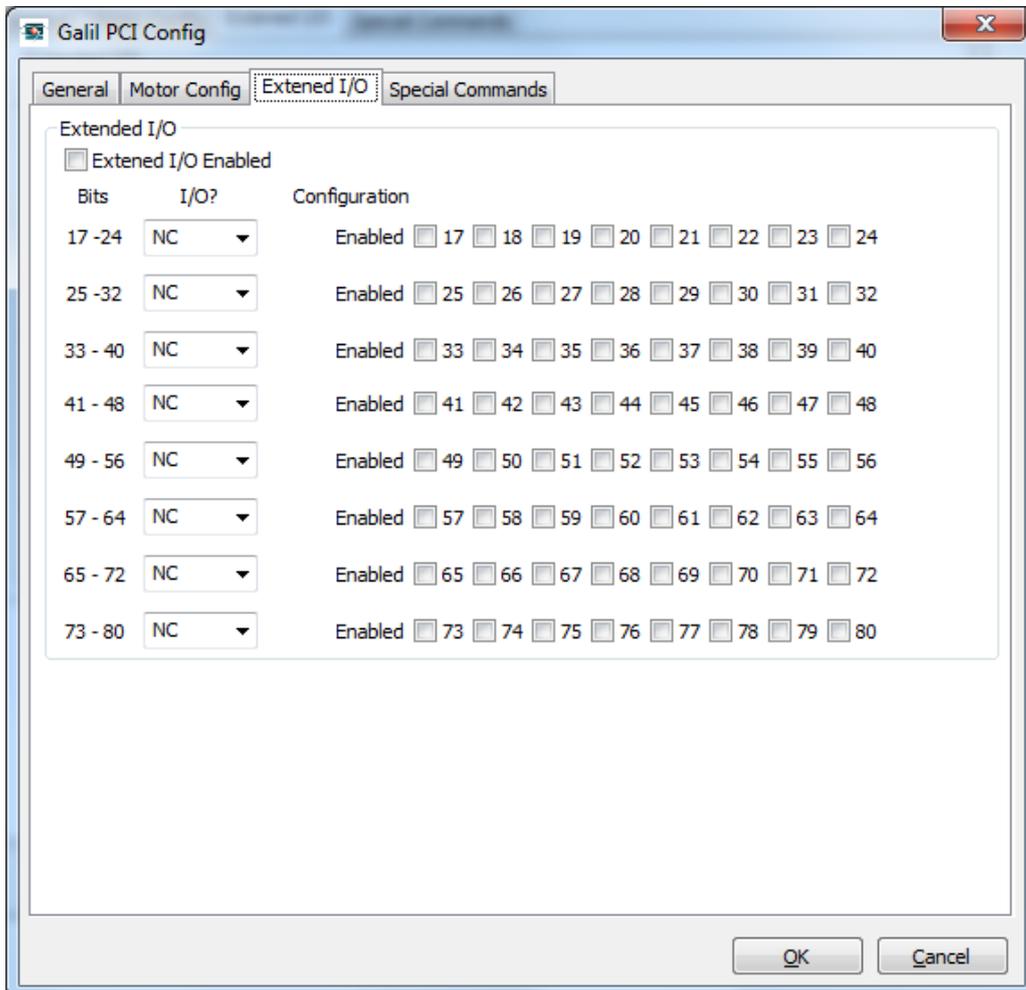
- 1.3. Use Index Pulse: If the encoder feedback is chosen, then the homing operation will use the index pulse from the encoder (if available) when this checkbox is checked.
- 1.4. Limits For Home: Some machines do not have separate homing switches and checking this checkbox will allow using the limit switches for the homing operation.
- 1.5. Position Feedback Reference:
Possible values are:
 - 1.5.1. Motor Encoder: For use with servo motors.
 - 1.5.2. Aux Encoder: For dual feedback systems.
 - 1.5.3. Reference Pos: For use with Stepper motors.
 - 1.5.4. User Var: For use with the Galil user data record variable.
- 1.6. The Mach Spindle can be mapped to any unused Galil axis, if desired.
2. Motor Types:
Each Galil axis motor type has to be defined. Possible values are:
 - 2.1. Servo Motor
 - 2.2. Servo Motor Reversed
 - 2.3. Stepper Hi Active
 - 2.4. Stepper Low Active
 - 2.5. Stepper Hi Active Reversed
 - 2.6. Stepper Low Active Reversed

It is important to note that the older Optima and Econo series controllers must also set the axis motor type with an on-board jumper.
3. Motor Options:
 - 3.1. Use Calculated Actual Speed Instead of Encoder: Check this if you are using steppers or wish Mach to calculate the motor velocities.
4. Spindle:
 - 4.1. **Spindle has Encoder:** If your spindle has an encoder and you wish to use it to allow the Galil controller to keep the spindle speed constant, then enable this checkbox.
 - 4.2. **Spindle is 0v to 10v:** If the spindle drive accepts a 0 to 10v command signal instead of a +-10v command signal, then enable this checkbox.
 - 4.3. **Counts/Rev:** This value is the number of counts per revolution for the spindle encoder. The value should be in quadrature counts. e.g. If your encoder has 512 counts per revolution, the Galil will see it as 2048 counts per revolution. In this example, 2048 is what you should enter in this field.
 - 4.4. **Accel Revs/Second:** This is the acceleration and deceleration value for the encoder based spindle in revolutions per second.

Galil PlugIn for Mach4

Extended I/O Tab

If the Galil controller supports Extended I/O, then the Extended I/O tab is available.



1. Extended I/O:
 - 1.1. **Extended I/O Enabled:** To enable extended I/O, this checkbox must be enabled.
 - 1.2. **I/O ?:** You must explicitly enable each block as either and Input or Output blocks, or leave it in the not configured state (NC).
 - 1.3. **Enabled:** Each I/O bit can be enabled or disabled depending on the state of its associated “Enabled” checkbox.

Galil PlugIn for Mach4

Probing:

The Galil PlugIn is capable of using a probe for various probing functions. This is accomplished using the Galil high speed position latch functions. Unfortunately, the latch inputs share general purpose inputs on the Galil ICM. Galil axes A, B, C, and D use inputs 1, 2, 3, and 4 and Galil axes E, F, G, and H use inputs 9, 10, 11, and 12 for the latch inputs.

IMPORTANT!!!! A probe should be wired to **ALL** latch inputs for any moving axis on your machine! For example, if you have a 3 axis mill and you are using Galil axes A, B, and C to drive them, then the probe's output should be wired to inputs 1, 2, and 3. This is because the G31 probe command will move in all 3 dimensions. If you have 3 axis mill with a 4th axis rotary driven from Galil axes A, B, C, and D, then the probe's output should be wired to inputs 1, 2, 3, and 4.

When setting up Mach for probing, it is imperative that the "Probe" signal is enabled in "Ports and Pin" on the "Input Signals" tab and the port and pin should be mapped to one of the Galil latch input pins. For example, if your 3 axis mill uses Galil axes A, B, and C to drive Mach axes X, Y, and Z, then the probe's output signal should be wired to Galil inputs 1, 2, and 3 and the Mach Probe signal should be enabled and mapped to port 1, and one of pins 25, 26, or 27. Which pin is mapped really does not matter since all of the inputs will be active upon a probe hit.

Galil PlugIn for Mach4

Credits:

Brian Barker: brianb@machsupport.com

Thanks go to Brian for feeling adventurous and writing the initial Galil PlugIn for Mach 3. Without Brian, the Galil PlugIn would not have materialized at all.

Jeff Birt:

Jeff added the Extended I/O support in the Mach 3 plugin. He has also been a lead tester and programmer for the PCI controller support in the Mach 4 plugin.

Kenny Crouch: kcrouch@cruises4fun.com

Kenny added the MPG support in the Mach 3 plugin. He has also been a lead tester and debugger for the Ethernet Accelera and Econo series controller in the Mach 4 plugin.

Steve Murphree: stevem@machsupport.com

Steve added the following in the Mach 3 plugin.

- The ability to control registered Galil controllers other than the first registered controller.
- Axis mapping.
- Contour Mode movement control.
- He originated this manual and continues to maintain version control of the plug-in software.
- Encoder controlled Spindle.
- Analog Input support.
- Custom Notify Messages
- Use Limits for Homes.

He is the lead programmer for the Mach 4 plugin.