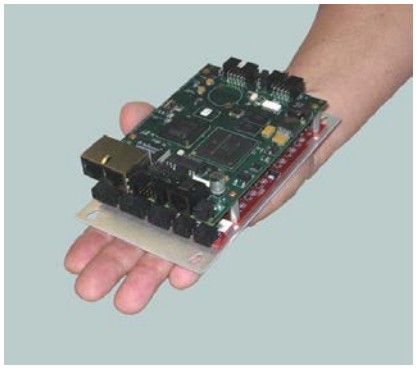
PRECISE AUTOMATION

Guidance Controller
Setup and Operation



Quick Start Guide

Version 4.2.0, January 1, 2019
PRELIMINARY RELEASE
P/N: 0000-DI-00010

Precise Automation Inc., 727 Filip Road, Los Altos, California 94024
www.preciseautomation.com

Document Content

Trademarks

Disclaimer

Precise Automation Inc.
727 Filip Road
Los Altos, California 94024
U.S.A.
www.preciseautomation.com

Warning Labels

The following warning and caution labels are utilized throughout this manual to convey critical information required for the safe and proper operation of the hardware and software. It is extremely important that all such labels are carefully read and complied with in full to prevent personal injury and damage to the equipment.

There are four levels of special alert notation used in this manual. In descending order of importance, they are:

DANGER: This indicates an imminently hazardous situation, which, if not avoided, will result in death or serious injury.

WARNING: This indicates a potentially hazardous situation, which, if not avoided, could result in serious injury or major damage to the equipment.

CAUTION: This indicates a situation, which, if not avoided, could result in minor injury or damage to the equipment.

NOTE: This provides supplementary information, emphasizes a point or procedure, or gives a tip for easier operation

Table Of Contents

Setup and Operation Quick Start Guide

Quick Start Guide Introduction

If you have purchased a complete robot system with an embedded Guidance Controller or have purchased a controller and would like an overview of how a complete system is used, this *Quick Start Guide* takes you through the steps to connect your controller to a computer, turn on the controller, move a robot in manual control, and generate a simple program to automatically drive a robot between several positions. Once you are familiar with these simple procedures, you will be ready to explore the full functionality and power of your system and the Guidance Controller by reading the detailed material provided in the *Precise Documentation Library*.

The Library includes a complete description of the controller's web interface, its native programming language (GPL), and the programming environment (GDE). The manuals for GPL and GDE are also available in PDF format as the *Guidance Programming Language, Introduction to GPL*, PN GPL0-DI-00010 and the *Guidance Development Environment, Introduction and Reference Manual*, PN GDE0-DI-00010.

If you are integrating the Guidance Controller to a new mechanical system for the first time, please see the "Software Setup" chapter in the "Controller Software" section of the *Precise Documentation Library* for instructions on software interfacing the controller to your mechanism. You should also obtain a copy of the "Guidance Configuration Utility" that is part of the "Guidance Development Suite" (GDS). This utility provides wizards that follow the Software Setup guide and take you step-by-step through the controller integration process.

DANGER: Before proceeding with this Guide, ensure that the following steps have been performed: (1) the robot has been properly mounted; (2) all required safety interlocks have been installed and tested; and (3) power has been connected. For Precise Automation robots, this information is provided in the robot's *"Hardware Introduction and Reference Manual"*.

Once you have verified that the steps outlined above have been completed, it's time to get started.

Interfacing Your Controller to a PC

The Guidance Controller includes all of the software necessary to operate the robot automatically in a standalone mode. However, a PC or other computer or tablet must be utilized as the graphical user interface (GUI) to manually operate and program the robot. For this Guide, the only application software required on the host computer is a standard web browser, although a licensed or unlicensed copy of the "Guidance Development Suite" (GDS) will be needed if the IP address of the controller has to be changed.

For the best results, it is suggested that you use the following equipment:

- A 500 MHz or faster PC running Windows XP or later
- Microsoft Internet Explorer version 8.x or later
- A 10/100 Ethernet interface and a standard Ethernet cable

To connect the robot's controller to a PC, perform the following steps:

» Connect the controller to a computer directly or via a hub or switch using the cable described above. If you are connecting to a robot or Guidance System, the cable should be plugged into the RJ45 on the Facilities Panel. If you are directly connecting to a controller, the cable can be plugged into any of the controller's RJ45 ports.

Turning On The Controller And Connecting

In general, two types of power must be supplied to the controller: (1) 24 VDC for powering the digital logic section (i.e. the CPU) and (2) motor voltage for driving the motors. For some systems, the motor power may also be low voltage and relatively safe. For other systems, the motor voltage may be as high as 340 VDC, which can be dangerous if improperly handled. When you first turn on the controller, only the low voltage is enabled. The controller must explicitly enable the motor voltage.

You can now turn on logic power for the controller by performing the following steps:

» Turn on the logic power for the controller. For systems like the PreciseFlex 400 robot, this is accomplished by supplying AC power to the IEC connector mounted in the robot's facilities panel and turning on the AC switch. This will supply the low voltage to the controller and AC power to the robot's motor power supply, but no power will be provided to the motors.

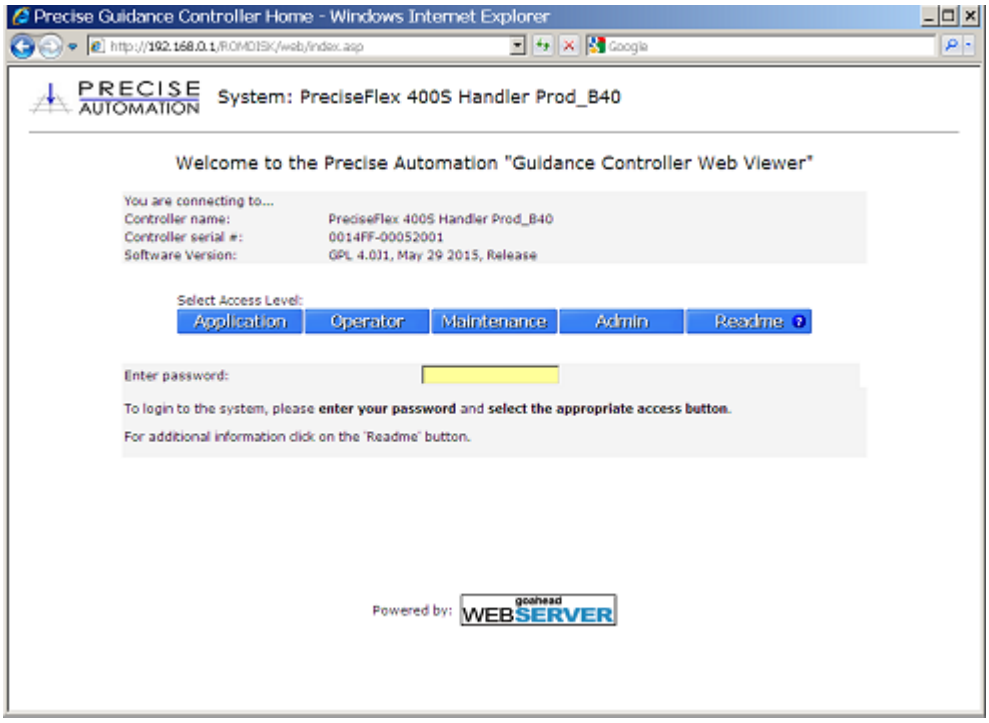
Once the low voltage is supplied, the controller executes its start up sequence. When this is completed, a green LED on the top board of the controller (if visible) or a LED mounted on the robot will blink at a rate of about once per second. This indicates that the controller has properly loaded and is ready to accept instructions. If the LED stops blinking, this indicates that the low voltage power to the controller has been turned off or that the system software has stopped operating.

After the controller has been successfully started, you can connect your computer to the controller.

» Start Internet Explorer (IE) or other browser on your computer.

» In the browser's address bar, enter the controller's Ethernet IP address. The default IP address is ***"192.168.0.1"***. If you have successfully connected to the controller, you will see the screen below.

» If you do not see the following screen, most likely the controller's and computer's IP addresses are incompatible. Go to the next sections for instructions on how to change the IP addresses, then repeat the previous instructions.



Initially, no passwords will be set and you can login to the system using the "Operator", "Maintenance" or "Admin" buttons. However, the “Operator” button will only allow access to a limited set of web pages, so it is best to login using either the “Maintenance” or the “Admin” button.

» Ignore the password field and press the “***Maintenance***” button to access the Precise Operator Interface.

Setting the Controller's IP Address

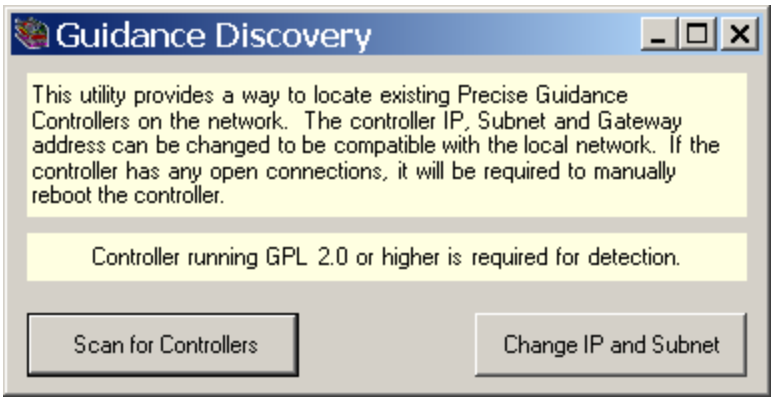
This section provides detailed instructions on how to change the IP address and subnet mask of a Guidance Controller. If you do not need to perform this operation, skip this section.

To change the controller's IP address and subnet mask, the following are required:

- The Guidance Controller must be executing GPL version 2.0 or later.
- You must have a licensed or unlicensed copy of the "Guidance Development Suite" (GDS) installed on your PC. This software is distributed with some Precise products or can be downloaded from www.preciseautomation.com.

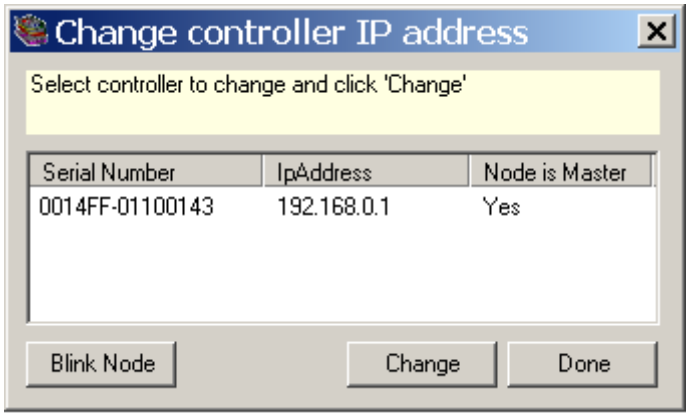
» Launch the Guidance Discovery application. This can typically be accomplished by selecting ***Start > Programs > Precise Automation > GDS > Guidance Discovery*** .

When this application begins execution, the following top-level window is displayed.

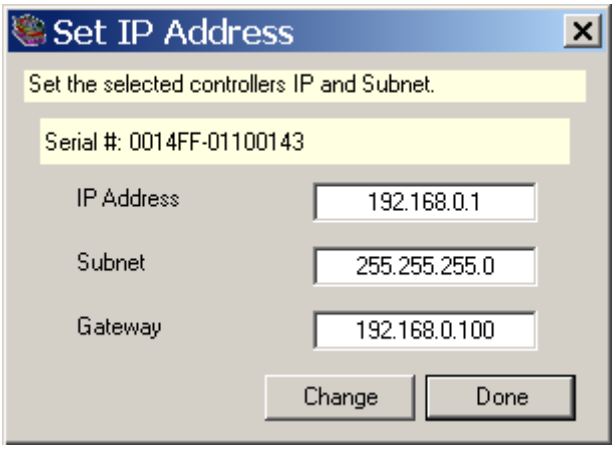


» To locate all Precise controllers connected to your PC, press the ***"Change IP and Subnet "*** button.

The following window will be displayed. Each located controller will be listed on a separate line.



» Click on the line detailing the controller of interest and press ***"Change"***. The window below will be displayed. If you are unsure of which controller you wish to change, press **"Blink Node"**. This will blink an LED on the controller or robot.



» Fill in the desired IP address, subnet mask and gateway address and press ***"Change".***

» Restart the controller in order for the change in the Ethernet IP address to take effect.

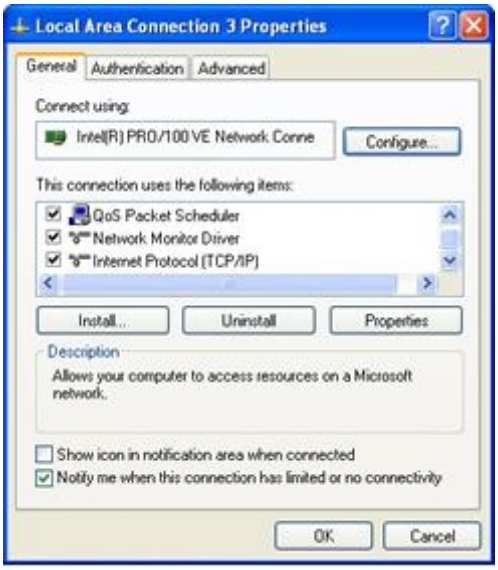
Setting the PC's IP Address in Windows XP

This section provides detailed instructions if you require assistance in configuring your PC's IP address and subnet mask to be compatible with the controller. If the IP addresses and subnet masks are already compatible, you can skip this section.

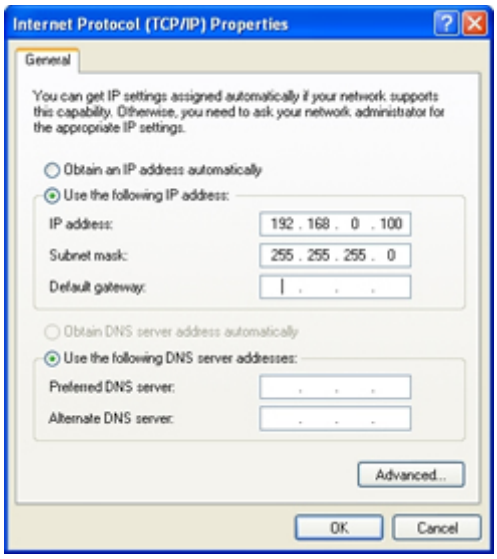
By default the Guidance Controller is factory configured with the IP address/subnet mask of 192.168.0.1 and 255.255.255.0. The PC utilized must be compatible with this IP information. For Windows XP users:

» Open the PC control panel by clicking ***Start > Settings >Network Connections***.

» Right click on the icon for the Ethernet card that is connected to the controller and select "***Properties***".

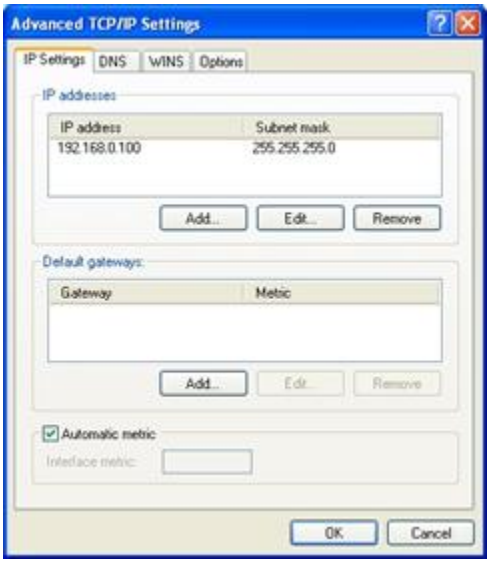


» Highlight "***Internet Protocol (TCP/IP)***" and click on the "***Properties***" button.



» If your PC is using DHCP and is not compatible with the controller, you will need to specify a compatible IP address/subnet mask. Click on the "***Use the following IP address***" radio button and enter IP information that is compatible. Once you have connected to the controller, you can change the IP address/subnet mask of the controller to be compatible with your network.

» If your PC is using a static IP address you do NOT need to change the IP address. Instead, click on the "***Advanced***" button.

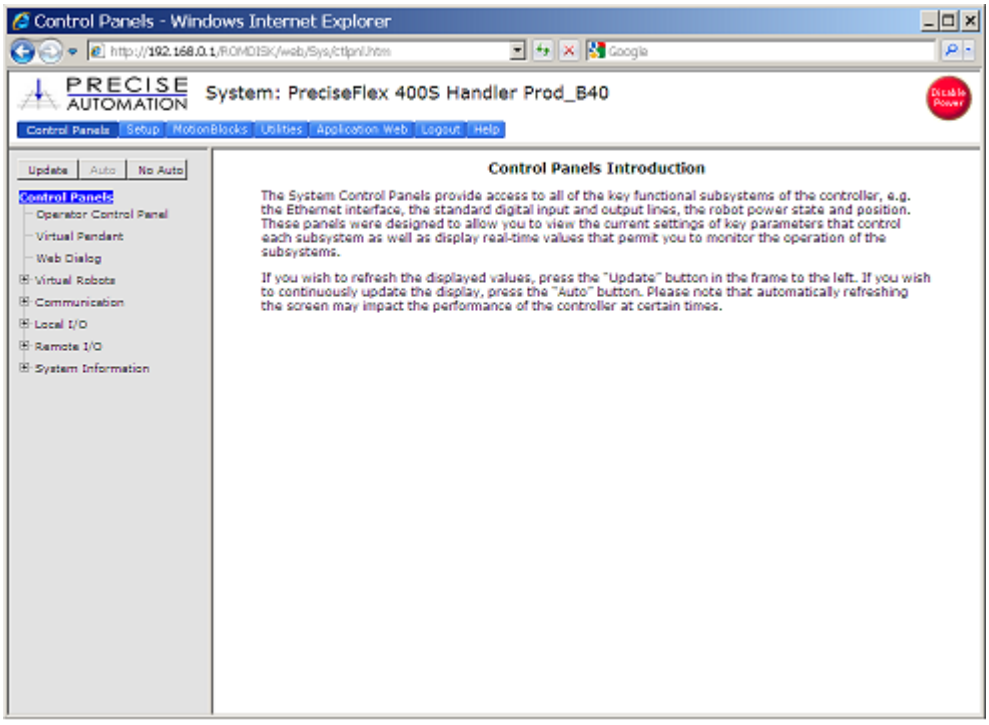


» Click on the "***Add***" button and enter in a compatible IP address and subnet mask. For example you could use '192.168.0.200' and '255.255.255.0'.

» Click the ***OK*** to activate the changes.

Standard Web Page Format

You should now be able to browse the web pages stored in the controller. When you enter the main web pages for the controller, you will be presented with the following page:



Before we introduce you to some of functions available on web interface, we will briefly describe the standard page layout and how to navigate about the web interface.

At the top of each page, you will see the name of the controller, the top-level navigation bar and a red “Disable Power” button.

WARNING: The Disable Power button on each page is provided as a convenience for turning off power to the robot’s motors. However, this should not be confused with a hardwired E-Stop button. A hardwired E-Stop button directly connects to the hardware and guarantees that the motor power will be quickly disabled. The Disable Power button relies upon the proper operation of software to function.

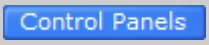
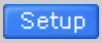
The navigation bar allows you to access all of the major groups of functions available via the web interface. For each group, the left panel displays a tree of specific topics that can be selected. Clicking on a topic displays the requested information in the bottom, right area of the page.

In the balance of this document, when we wish you to click a tab in the navigation bar, followed by a topic and subtopic in the topic tree, we will indicate this as follows:

Navigation button > Major topic > Minor topic

For example, ***Control Panels > System Information > System ID*** will display identification information on the controller hardware and software. Please try clicking on these now to become familiar with navigating the web interface.

The following briefly describes the contents of each of the tabs on the main navigation bar:

Main Navigation Bar Summary	
	These pages contains virtual control panels that are used on a day-to-day basis to operate the controller and monitor the system status. These include panels for starting and stopping programs, manually operating the robot, displaying and modifying the values of digital input and output signals, etc.
	The setup pages provide system configuration aids and access to all of the system parameters and dynamic system data values that can be monitored. Normally, this group of pages is only accessed to configure the controller and to perform maintenance operations.
	The MotionBlocks provide a very easy method for training the robot to execute a sequence of simple operations. MotionBlocks are taught by filling in a series of tables and require no programming experience. At the end of this guide, we will show you how to use this facility to automatically move the robot between several positions.
	The system utilities allow you to perform operations such as copying disk files to/from the controller's flash disk, upgrading the controller with new system software, and executing built in methods for collecting various types of dynamic information.
	This logs you out of the main web pages and takes you back to the initial login page.
	This contains links to other useful sites along with some very simple help information.

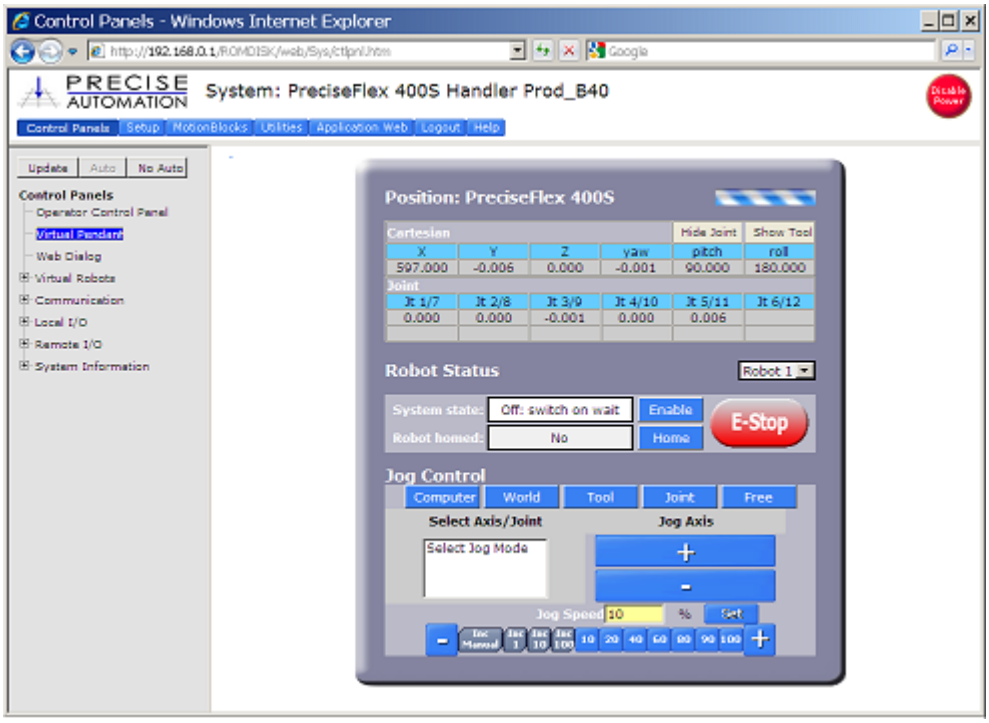
Now that you understand the general organization of the web interface, you should take a moment and navigate around the various pages and topics. If you would like more detailed information on the web interface, please see the section on the "Guidance Operator Interface" in the Documentation Library.

Enabling Motor Power and Homing the Robot

With a basic knowledge of the web operator interface, you can now proceed to turning on power to the robot and preparing it to run. To do so, please perform the following:

» Open the Virtual Manual Control Pendent (V_MCP) by pressing: ***Control Panels > Virtual Pendant.***

The following page will be displayed:



The top section of the V_MCP shows the current position of the robot in Cartesian coordinates followed by the position of each axis. Distances are displayed in millimeters and angles in degrees. Initially, the values will probably be zero since the system has not yet determined the absolute position of each axis. Above this section is a "barber pole" that rotates when the V_MCP is connected to the controller. The middle section of the V_MCP indicates the system status and has buttons for enabling and disabling motor power and homing the robot (more about this later). The bottom section allows you to manually move the robot in various modes.

You are now ready to enable power to the robot's motors. For some robots, the first time that power is enabled after the controller is restarted, the system will also “commutate” each motor. Commutation may move each axis a small distance in order to determine the alignment of the phases of the motor windings. For other robots, the commutation is performed as part of the homing operation.

» Click the V_MCP “***Enable***” button.

The “System state” will illustrate the steps being performed. After about 10-15 seconds, the state should transition to “On: auto mode wait”. This means that motor power is on and the system is waiting for an automatic execution mode to be initiated. You will also see the LED on the top of the controller or the LED of a Precise robot blink faster, at a rate of approximately 4 times per second, to indicate that motor power is enabled. If these indicators ever blink much faster than 4 times per second, it indicates that either the CPU or one of the amplifiers is running too hot and will be automatically shut down.

The final step before the robot can be utilized is to “home” each axis. For most robots, each axis must be homed in order to determine its position relative to a fixed reference. This permits each axis to be set to an “absolute position” that allows formerly taught programs to be reliably repeated.

The homing method varies from robot-to-robot and from axis-to-axis. Some axes are equipped with special homing sensors. Others simply reply upon a travel limit signal or hard stop to reestablish the absolute position. In the case of the PreciseFlex 400 robots, the four primary axes are equipped with Absolute Encoders that allow the absolute position to be determined without any movement of these axes.

» Press the V_MCP “***Home Robot***” button to start the homing sequence.

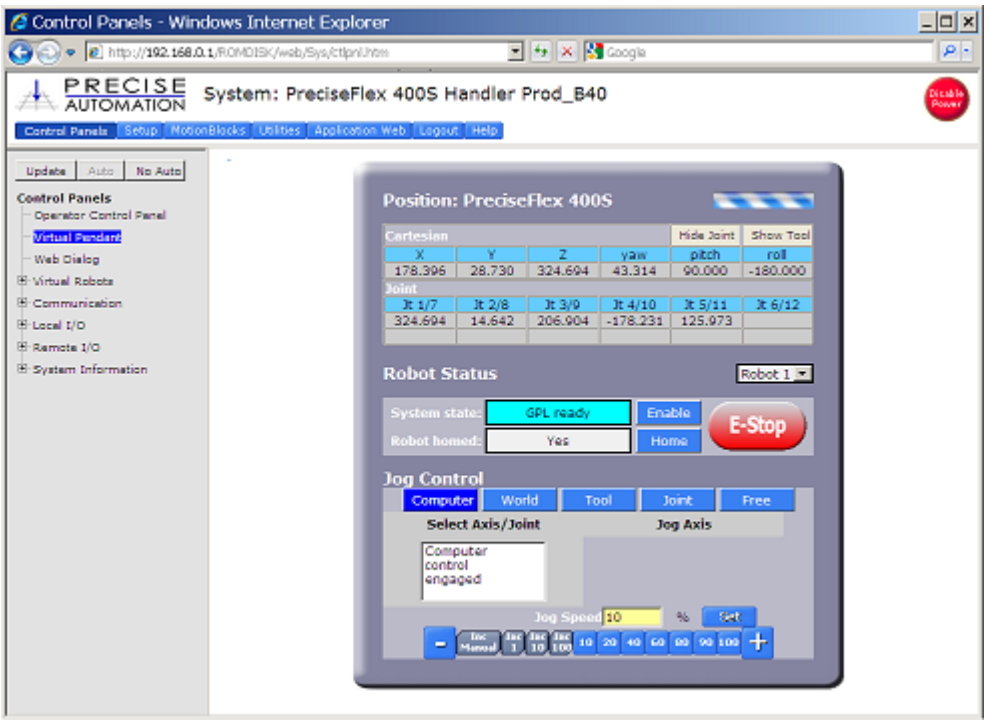
At the conclusion of the homing sequence, the “***Robot homed***” indicate will change to “***Yes***”.

The commutation and homing operations only have to be performed once after the system is restarted no matter how many times power to the robot is enabled and disabled.

The robot is now ready to be operated in manual or automatic mode.

Moving the Robot in Manual Control Mode

After robot power has been enabled, the motors have been commutated and the axes homed, the robot can be manually repositioned using the Virtual Manual Control Pendant. The V_MCP display (***Control Panels > Virtual Pendant***) will now look like the following:



Note that the "Robot Status" indicates that motor power is on and that the robot has been homed.

Using the V_MCP “Jog Control”, you can now manually move the robot. The following briefly describes the manual control modes:

Manual Control Modes (Jog) Summary	
Computer	Disables Jog Control and permits a user program to take control of the robot and automatically move its axes. This mode should always be selected when you have finished moving the robot in jog mode.
World	Moves all of the axes of the robot in a coordinated fashion such that the gripper travels in a straight-line path along or rotates about the X, Y, or Z axes of the robot’s base reference frame.
Tool	Similar to World mode except that the reference frame is determined by the robot’s gripper orientation. For example, if the gripper is aligned with a hole that is at a skewed angle, moving down and up in Tool Z will move the gripper into and out of the hole.
Joint	Moves individual joints one at a time. Even if a joint is beyond its limit stops, this mode can be used to drive the joint back into the working range.
Free	Allows one or more joints to be taken out of position control mode to permit the axis to move freely. For light-weight robots, this allows the operator to grip an axis and manually repositioned it.

To move a single joint in manual mode, do the following:

» Press V_MCP ***Jog Control > Joint.***

» Select a joint to move, e.g. ***Joint 1***

» Select a slow speed initially, e.g. "***10***"

» Move the axis in the positive or negative direction by holding down the "**+**" or "**-**" keys.

The World and Tool modes operate in the same fashion. The Free mode is a bit different in that multiple joints can be freed at once and the speed setting is not used. For this mode, each time you select an axis and press the "+" the axis is freed until you press the "-" key to place the axis back into position control mode.

If the jog speed is set to 5% or lower, the robot will move a discrete increment and then stop rather than move continuously. If you release the "+" or "-" key and press it again, the robot will move another increment. This is very convenient for fine positioning the robot. The V_MCP includes buttons for setting several increment sizes.

Try to move the robot in each of the manual modes and at various speeds to get a feel for the virtual pendant. Please note the following:

- For some robots, Free mode may be disabled for some or all of the axes to avoid a dangerous situation or potential damage to the robot.
- For some PC's and network environments, you may notice that the robot occasionally stops momentarily. This can be due to other applications running on the PC or excessive network traffic.

When you are done practicing with the manual control, press the Computer mode key. This disables Jog Control and permits the automatic execution programs to take control of the robot.

Hardware Manual Control Pendant

If your system is equipped with a Precise Hardware Manual Control Pendant (H_MCP), it can be used to operate the system in a manner very similar to the Virtual Manual Control Pendant. The advantages of the H_MCP are that it can be carried close to the robot (if adequate safety interlocks exist) and it has a hardware E-Stop button that is directly connected to the controller's dual hardware E-Stop circuits. Both the V_MCP and the H_MCP can be used with the same controller, although only one MCP can be in manual control mode at any time as a safety precaution.

For proper operation, the H_MCP must be connected to the controller's Remote Front Panel Interface either directly or through a front panel.

In its background mode, the H_MCP's display indicates the power status of the system or the manual control mode currently in effect. In addition, the LCD can display the current position of the selected robot, error log and system ID data, and user prompts.

The keypad is divided into several sections whose keys are described in the following tables. The top row of keys control the major operational functions of the H_MCP.

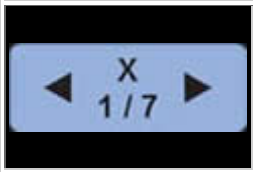
Hardware MCP - Major Mode Keys	
MANUAL	When motor power is enabled, this key selects the manual control mode: Computer, World, Tool, Joint or Free. Each time this key is pressed, it cycles to the next mode. If the selected robot contains more than 6 axes, this key must be pressed twice for Joint and Free modes to cycle through the first 6 axes and the second 6 axes. The three LED's in the blue box connected to this key are interpreted as follows:

LED 1	LED 2	LED 3	MODE
0	0	0	Computer
X	0	0	World
0	X	0	Tool
X	X	0	Joint
0	0	X	Free

POWER	Enables and disables motor power. The LED in the corner blinks when power is coming up and is solid on when power is enabled.

	Initiates execution of the homing sequence for all robots. The LED in the corner blinks during the homing sequence and is solid on when power is enabled and all robots are homed.
	Switches between GPL application control of the LCD display and keypad and standard H_MCP operation. The LED in the corner is on when an application is in control and blinks when an application wants to gain control. If an application dialog is being displayed and a manual control key is pressed or an error occurs, the MCP switches back to it standard control mode. Pressing the APP key switches between application control and the standard operation of the MCP.

When the H_MCP is in one of the manual control modes, the keys described in the following table control the movement of the robot. The selected mode, speed and robot number are displayed on the LCD.

Manual Control Mode Keys	
	For systems with multiple robots configured, this key cycles through the robot being controlled.
	For World, Tool and Joint modes, these keys decrease and increase the speed. As with the V_MCP, below 5% speed, the motion switches from continuous movement to discrete steps.
	For World, Tool and Joint modes, there are six keys (X/1/7 to ThetaZ/6/12) that move the selected axis in a positive or negative direction. Only one axis can be moved at a time. For Free mode, these keys lock/unlock selected axes. Multiple axes can be simultaneously unlocked.

The following keys present information in the LCD display. These information displays supersede the background data (i.e. power sequence state and manual control mode). To terminate these displays, the "QUIT" key should be pressed.

Operator Display Functions	
	Dynamically displays the robot position in Cartesian coordinates: X, Y, Z, Yaw, Pitch and Roll. For systems with multiple robots, the location of the robot selected for manual control is displayed.
	Dynamically displays the robot axes positions in millimeters or degrees. For robots with more than 6 axes, press this button a second time to display the second set of 6 axes. For systems with multiple robots, the location of the robot selected for manual control is displayed.
	Displays the most recent entry in the system Error Log. The "PREV" and "NEXT" keys can be pressed to sequence through entries in the log. If the "DEL" key is pressed, all entries in the Error Log are cleared.

? F4	Displays system identification information. If this button is pressed a second time, the controller's Ethernet IP address and mask are displayed along with the IP address for the associated machine vision server.

In the remainder of this guide, the V_MCP will be referred since it is available on all controllers, although the H_MCP can be used as well.

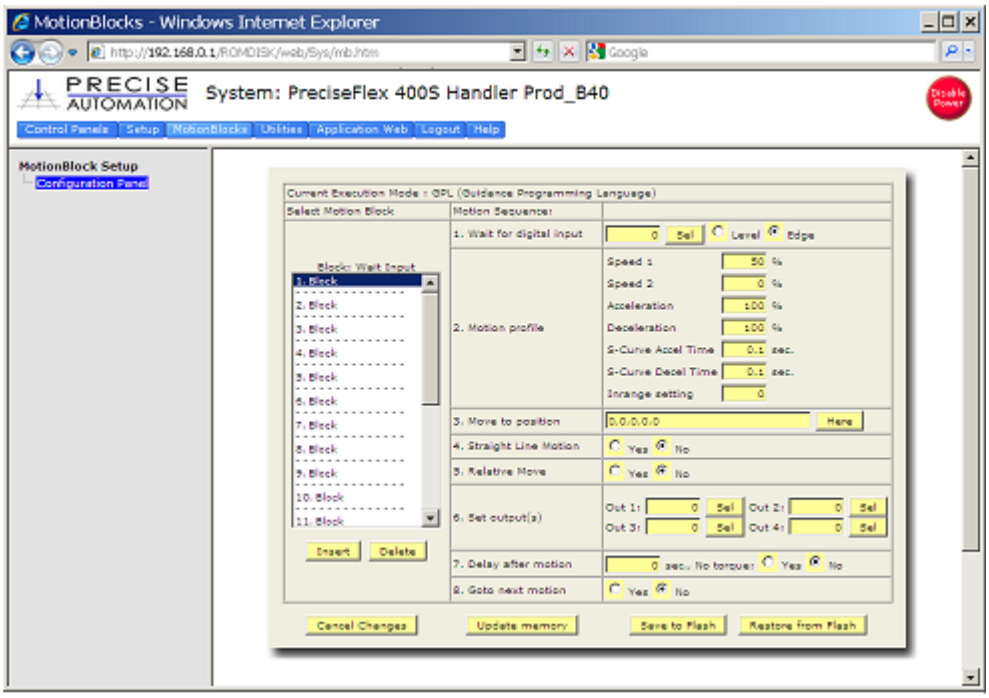
Teaching a Simple MotionBlocks Program

There are three methods for programming the Guidance Controller: (1) DIO MotionBlocks, (2) the embedded Guidance Programming Language (GPL), or (3) a standard PC language that is connected to the controller via TCP/IP.

MotionBlocks is the easiest method for programming a robot and is intended for simple applications such as machine loading. Users of MotionBlocks do not require any software training. They simply teach up to 20 robot locations and fill in a form for each motion via the web based Operator Interface.

In this guide, we will show you how to program a simple MotionBlocks application and execute it. Let's start by examining the MotionBlock data editing form. (For more information on GPL or the remote programming interface, please see the Documentation Library.)

» Open the MotionBlocks editing page, ***MotionBlocks > MotionBlock Setup > Configuration Panel***. The web page should look as follows:



The left section of the editing form ("Select Motion Block") allows you to select a MotionBlock to edit. Once a block is selected, its data is displayed on the right ("Motion Sequence"). Each MotionBlock generates a single motion to a specified destination.

The system executes MotionBlocks by scanning the list of blocks from top to bottom until it finds a block that has an associated digital input channel. This digital input serves as a trigger for initiating the execution of the MotionBlock. The trigger can be defined as an edge (i.e. a transition from off-to-on or on-to-off) or a high or low level. If the trigger condition is satisfied, the motion contained in the MotionBlock is executed. At the completion of the motion, the system scans the list again from top to bottom looking for the next motion to execute. Scanning and motion execution continues until this method of execution is terminated.

As an convenience, several blocks can be grouped together to execute a sequence of motions in response to a single trigger signal. In this case, only the first block in the sequence has a non-zero trigger value.

The following table briefly describes the parameters contained within a MotionBlock. For more detailed information, please see the section on Automatic Execution methods in the Documentation Library.

MotionBlock Parameter	Description
Wait for digital input	Channel number of digital input trigger. Depending upon the setting of **Edge** or **Level**, positive numbers indicate that the MotionBlock is initiated on the upward going edge of the signal or the signal being **TRUE**, respectively. Negative numbers indicate triggering on the downward going edge of the signal or a **FALSE** signal. The hardware digital input channels range from 10001 to 18000 and the software digital input/output channels range from 20001 to 21000.
Motion profile	These values define the speed, acceleration and deceleration of the motion as a percentage of the nominal robot speed, acceleration and deceleration. The S-Curve accel and decel times specify the time required to ramp the acceleration or deceleration to their specified levels. These values are often set to 0.1 seconds to smooth out the motion. If the **Inrange** value is negative, sequential motions are blended. If this value is 0, the motion comes to a stop at its destination but does not wait for any position errors to be corrected before going on to the next motion. If **Inrange** is >0 and <= 100 the motion comes to a stop at its destination and the system requires an increasing level of accuracy in achieving the final position before going on to process the next motion. **Speed2** is a secondary speed specification that is useful for Cartesian (straight-line) motions when the destination is unpredictable. For MotionBlocks, this should almost always be set to 0, which ignores this parameter.
Move to position	Specifies the position of each axis of the robot in mm or degrees for the final destination of the motion. Pressing the **Here** button sets the position value equal to the current position of the robot.
Straight line motion	Set **Yes** if the robot's tool tip is to follow a straight line path or **No** if each axis is to be individually interpolated.
Relative move	Set **Yes** if the motion is relative to the previous motion.
Set outputs	The values of up to 4 digital output channels can be set when this motion completes or blending begins for the next motion, whichever comes first. The hardware digital output channels range from 1 to 8000 and the software digital input/output channels range from 20001 to 21000.
Delay after motion	If **Inrange** dictates that the motion should be stopped at the destination, this defines an optional dwell time before the next motion is executed.

Goto next motion	Set **Yes** if the next motion is to be combined with the current motion into a sequence of motions. This value is set to **No** for the last MotionBlock in a sequence to be executed in response to a single trigger signal.

We will now generate a simple MotionBlocks program. This program will move the robot to a simulated pickup location, lower the gripper, retract the gripper, move to a safe position and then repeat the cycle continuously.

To create this program, we start by defining the first motion. This motion will be to a position that is above the simulated pickup position. This position should be somewhere in your workspace that is clear and where the gripper can move down and up.

» Return to the Virtual MCP and position the robot above the simulated pickup position with the robot's gripper slightly retracted.

» Bring up the first MotionBlock for editing. ***MotionBlocks > MotionBlock Setup > Configuration Panel*** > ***1. Block***

» Fill in the following information in the Motion Sequence (right panel):

MotionBlock Parameter	Value to Set	Explanation
Wait for digital input	-20001 Level	Start if soft signal 20001 is off
Speed1	100	Full speed
Speed2	0	Secondary speed specification, ignored
Acceleration	100	Full acceleration
Deceleration	100	Full deceleration
S-Curve Accel Time	0.1	Smooth acceleration
S-Curve Decel Time	0.1	Smooth deceleration
Inrange setting	-1	Don't stop at end of motion
Move to position	Press "**Here**"	Record current robot position
Straight line motion	No	Move in joint mode
Relative move	No	Position in absolute coordinates
Set outputs	0,0,0,0	Don't modify any DOUT
Delay after motion	0 No	No delay at end of motion
Goto next motion	Yes	Continue to next motion block

» Press the "***Update memory***" button to save your new values. If you do not press this button, the new values will be lost if you switch to a new block or other web page.

We now wish to define a second MotionBlock that will automatically lower the gripper to the simulated pickup position. At the conclusion of this motion, the robot will stop and wait until the gripper roughly reaches the specified position before proceeding with the third motion.

» Return to the Virtual MCP and move the gripper of the robot down to the simulated pickup position.

» Bring up the second MotionBlock for editing. ***MotionBlocks > MotionBlock Setup > Configuration Panel*** > ***2. Block***

» Fill in the following information in the Motion Sequence (right panel):

MotionBlock Parameter	Value to Set	Explanation
Wait for digital input	0 Level	No trigger necessary since continuing
Speed1	50	Half speed
Speed2	0	Secondary speed specification, ignored
Acceleration	50	Half acceleration
Deceleration	50	Half deceleration
S-Curve Accel Time	0.1	Smooth acceleration
S-Curve Decel Time	0.1	Smooth deceleration
Inrange setting	10	Stop at end of motion, coarse tolerance
Move to position	Press "**Here**"	Record current robot position
Straight line motion	No	Move in joint mode
Relative move	No	Position in absolute coordinates
Set outputs	0,0,0,0	Don't modify any DOUT
Delay after motion	0 No	No delay at end of motion
Goto next motion	Yes	Continue to next motion block

» Press the "***Update memory***" button to save your new values.

For the third motion, we want to retract the gripper back to approximately the same height as the first position. Typically, such a motion is performed to ensure that we do no drag a gripped part across the work surface and to clear any obstacles. Please perform the following operations to define the third MotionBlock.

» Return to the Virtual MCP and retract the gripper of the robot so it is above the simulated pickup position and approximately at the same height as the first position.

» Bring up the third MotionBlock for editing. ***MotionBlocks > MotionBlock Setup > Configuration Panel*** > ***3. Block***

» Fill in the following information in the Motion Sequence (right panel):

MotionBlock Parameter	Value to Set	Explanation
Wait for digital input	0 Level	No trigger necessary since continuing

Speed1	50	Half speed
Speed2	0	Secondary speed specification, ignored
Acceleration	50	Half acceleration
Deceleration	50	Half deceleration
S-Curve Accel Time	0.1	Smooth acceleration
S-Curve Decel Time	0.1	Smooth deceleration
Inrange setting	-1	No need to stop after this motion
Move to position	Press "**Here**"	Record current robot position
Straight line motion	No	Move in joint mode
Relative move	No	Position in absolute coordinates
Set outputs	0,0,0,0	Don't modify any DOUT
Delay after motion	0 No	No delay at end of motion
Goto next motion	Yes	Continue to next motion block

» Press the "***Update memory***" button to save your new values.

Finally, we want to move the robot to a last, safe position. For the safe position, you should select a robot location that is at approximately the same height as the third position and away from the simulated pickup location. Please perform the following operations to define the final MotionBlock.

» Return to the Virtual MCP and move the robot away from the simulated pickup position, maintaining the same height as the first and third positions.

» Bring up the last MotionBlock for editing. ***MotionBlocks > MotionBlock Setup > Configuration Panel*** > ***4. Block***

» Fill in the following information in the Motion Sequence (right panel):

MotionBlock Parameter	Value to Set	Explanation
Wait for digital input	0 Level	No trigger necessary since continuing
Speed1	100	Full speed
Speed2	0	Secondary speed specification, ignored
Acceleration	100	Full acceleration
Deceleration	100	Full deceleration
S-Curve Accel Time	0.1	Smooth acceleration
S-Curve Decel Time	0.1	Smooth deceleration
Inrange setting	0	Stop at end of motion, no inrange checking
Move to position	Press "**Here**"	Record current robot position
Straight line motion	No	Move in joint mode
Relative move	No	Position in absolute coordinates

Set outputs	0,0,0,0	Don't modify any DOUT
Delay after motion	0 No	No delay at end of motion
Goto next motion	No	Final motion

» Press the "***Update memory***" button to save your new values.

The MotionBlocks are designed to store several sequences of motions each of which is triggered by a different digital input signal. To ensure that your system does not have other motions that might inadvertently be triggered when running this example, we should disable any other existing sequences.

You can easily see which MotionBlocks are primed for execution. The "Select Motion Block" section of the display lists each sequence. The "---" lines indicate groups of blocks. If a trigger can start a group, the signal number is listed to the right of "Block".

To disable all other sequences, please do the following:

» Open the MotionBlocks editing page, ***MotionBlocks > MotionBlock Setup > Configuration Panel***.

» Review the lists of MotionBlocks in the "Select Motion Block" scrolling list.

» For each sequence of blocks that has a trigger set (other than Block 1 that we just defined), click on the block in the list to bring up its data.

» Zero the "Wait for digital input" signal number to disable the block

» Press the "***Update memory***" button to save your new values.

Now that we have defined our complete MotionBlocks program, we should save the taught data to the flash disk to preserve the information if the controller is powered down.

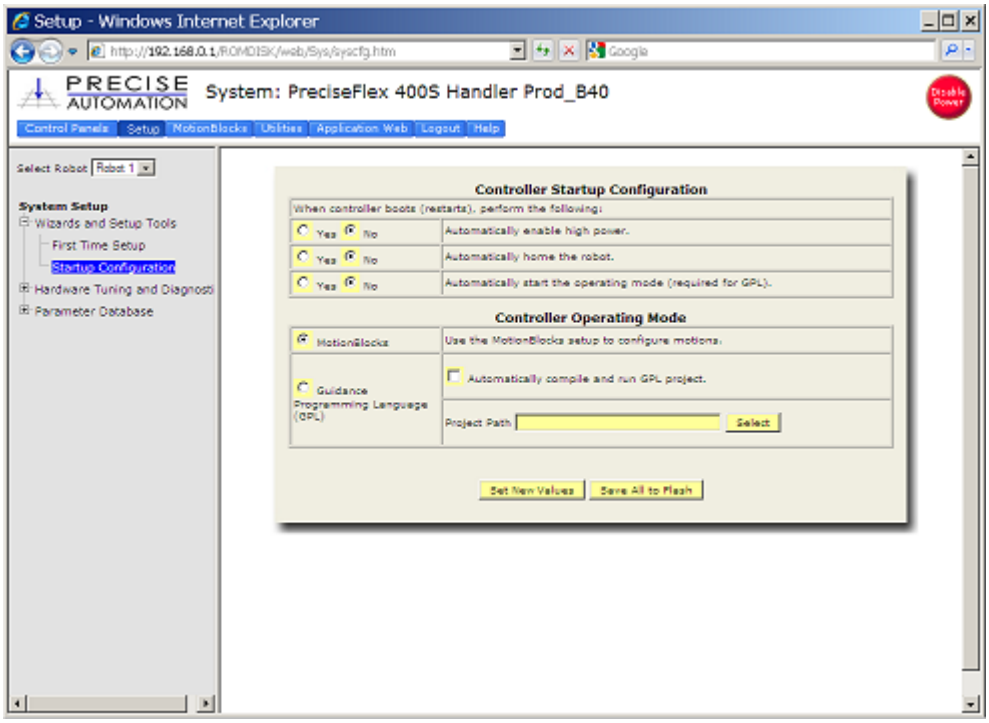
» Press the "***Save to Flash***" button at the bottom of the MotionBlocks editing panel. The button will flash for 10-30 seconds as the data is being written. **DO NOT TURN OFF YOUR CONTROLLER WHILE THE BUTTON IS BLINKING SINCE THIS MAY CORRUPT THE FLASH DISK**.

Congratulations, you have successfully written your first MotionBlocks program. You are now ready to automatically execute this sequence of instructions.

Executing a MotionBlocks Program

In order to execute a MotionBlocks program, the controller must be configured for MotionBlocks program execution instead of GPL.

» Open the Controller Startup Configuration page, ***Setup > Wizards and Setup Tools> Startup Configuration***. The web page should look as follows:



» Press the red "***Disable Power***" button at the top of the page. This is required because the changes that you will make are not permitted when power to the robot is enabled.

» In the "Controller Operating Mode" panel, click on the "***MotionBlocks***" radio button.

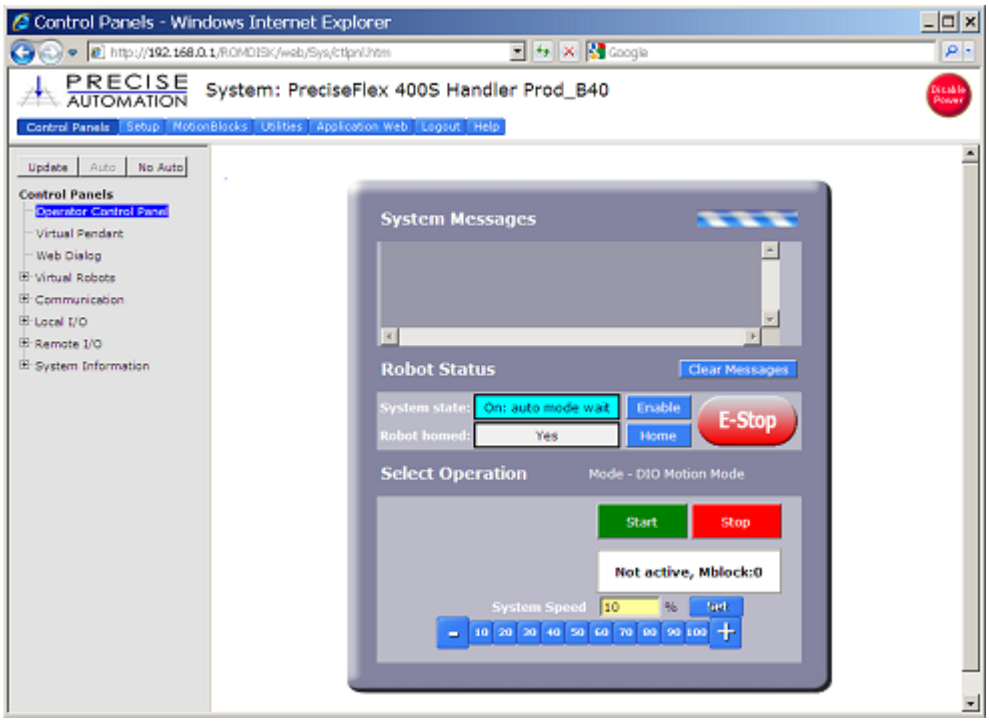
» Press the "***Set New Values***" button to store this setting into memory.

» Press the "***Save All to Flash***" button to store this setting in the flash disk. This ensures that this setting will remain in effect if the controller is restarted. The button will flash for 10-30 seconds as the data is being written. **DO NOT TURN OFF YOUR CONTROLLER WHILE THE BUTTON IS BLINKING SINCE THIS MAY CORRUPT THE FLASH DISK.**

The controller is now ready to execute the MotionBlocks program. Program execution is controlled via the Operator Control Panel. Please open this panel now.

» Open the Operator Control Panel, ***Control Panels > Operator Control Panel.***

» Press the "***Enable***" button to turn power to the robot back on. After power cycles through its various states, the web page should look as shown below.



If any error occurs, the error message, error code, and date and time when the error occurs will be displayed in the "System Messages" area of the Operator Control Panel. In addition, other general system messages that are not errors also appear in this panel. For example, at this time, you may have some messages displayed that indicate that you have cycled the power to the robot. Pressing the "Clear Messages" button clears the log of error messages. You can press this button at any time if you wish.

The "Robot Status" section is identical to the Virtual MCP. It shows the current operational state of the controller and has buttons for enabling and disabling robot power as well as for initiating the homing sequence.

The "Select Operation" is the panel that we want to focus on now. It provides the means for starting and stopping execution of the DIO MotionBlocks. In addition, it has a "System Speed" parameter that allows you to slow down the robot until you are comfortable that the program is correct. Please note, that this speed is independent of the speed in the Virtual MCP and the two should not be confused.

To execute your MotionBlocks program, please perform the following steps:

» Press the *"10"* button in the System Speed strip to operate the robot at 10% of the normal program speed

» When you are ready to have to robot move, press the "***Start***" button.

Congratulations! Your first robot program should now be executing and the robot should be moving slowly from position-to-position.

If you wish to stop the MotionBlocks at any time, please do the following:

» To stop the program at any time, press the "***Stop***" button.

» Alternately, you can press the red "***Disable Power***" button but power will have to be re-enabled when you wish to continue.

Once you have verified that the program is operating properly, you can gradually increase the speed by: pressing the "+" key, pressing one of the speed buttons or by typing a speed into the box with the yellow background. You can increase the speed as the robot is running.

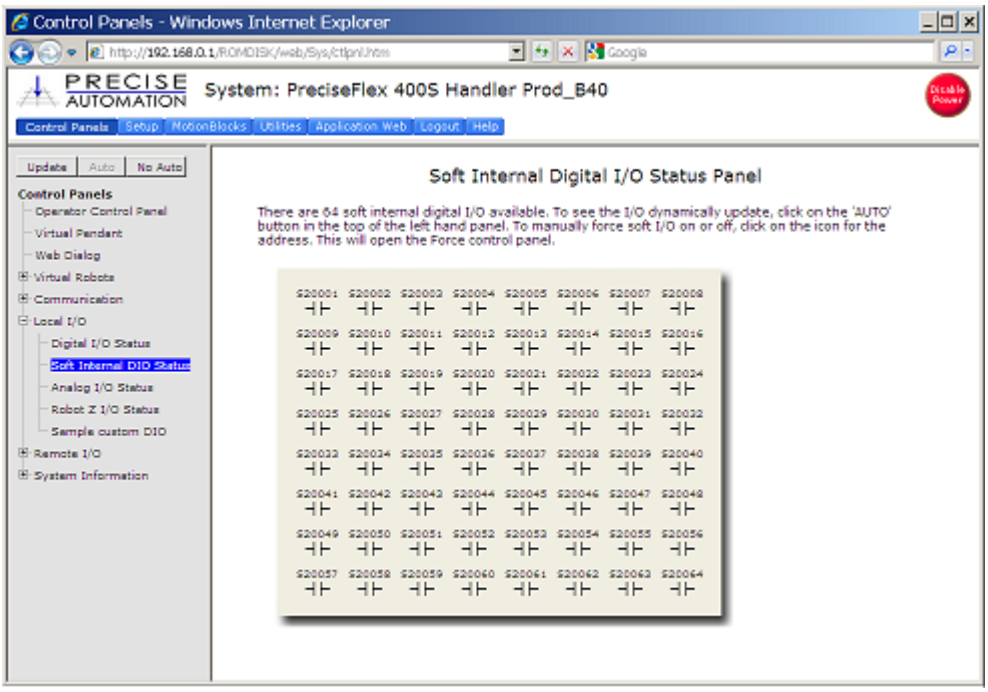
Additional Exercises

Once you have successfully executed your MotionBlocks program, you may wish to gain more experience by trying some of the following.

A. The execution of a MotionBlocks is really intended to be triggered by an external event that sets a digital input signal. In our example, we selected input signal -20001, level trigger. Since 20001 is an internal soft IO signal that is set to FALSE by default, this setting permitted the sample MotionBlocks to begin execution immediately.

With your sample MotionBlocks program executing, you can turn on signal 20001. This will stop execution of your sample program the next time that block 1 is executed. When you turn 20001 back off, your program will continue executing. To change the state of the trigger signal, do the following:

» Open the control page for the Soft internal IO, ***Control Panels > Local IO > Soft internal DIO Status***. The web page should look as follows:



This panel shows the current state of each of the internal soft digital IO signals. In the picture above, all of the soft signals are in their OFF (or FALSE) state. If you change the state of a signal on this panel, the change will take effect immediately. To change the state of 20001, do the following:

» Click on the desired signal, e.g. ***"S20001"*** in our example. A panel will display to the right that allows you to force the signal On or Off or not forced.

» Press the "***Force On***" button. The MotionBlocks program will stop moving the robot the next time that it attempts to execute the first block.

» Press the "***Force Off***" button and the MotionBlocks program will start moving the robot once again.

B. Try changing the sample MotionBlocks example to an "Edge" trigger. This will require that the soft signal transition from a Off to an On or an On to an Off state before the blocks are executed. Also, the sequence will only execute a single time until the signal goes through the same transition sequence.

C. Try adding additional MotionBlocks segments that are triggered by other soft signals. Then you can control the sequence that is executed by turning signals on and off in the "Soft internal IO" panel.

D. Try altering some of the other parameters in the MotionBlocks, such as the Speed, Inrange, etc. If you add +20002 as a "Set Output" signal to block 2 and a -20002 to the "Set Output" signal of block 4, you will see the value of this soft signal toggle as the motion is executed. You can press the "Auto" button in the left hand area of the Control Panels page to automatically refresh the displayed IO values.

With the experience you have gained, it's now time to plunge into all of the information provided in the *Precise Documentation Library* to learn more about your system. For example, in the library, you will find extensive information on the Guidance Programming Language (GPL). This is a full featured real-time language that is embedded in the controller. It provides capabilities well beyond that possible with the MotionBlocks.

Copying a GPL Project into the Flash Disk

The Guidance Programming Language (GPL) is a full-featured programming language. It closely resembles Microsoft Visual Basic.NET with the addition of built-in classes and objects to support general motion control. While programming in GPL is relatively easy, creating a GPL application does require a basic knowledge of programming and software development tools. So writing a new GPL Project is beyond the scope of this *Quick Start Guide*. However, if a GPL Project has been previously developed, it is straight-forward to load it and start its execution.

In this section, the steps for copying a GPL Project into a controller's flash disk are described. If the GPL Project that you wish to execute is already stored in the flash, skip the remainder of this section.

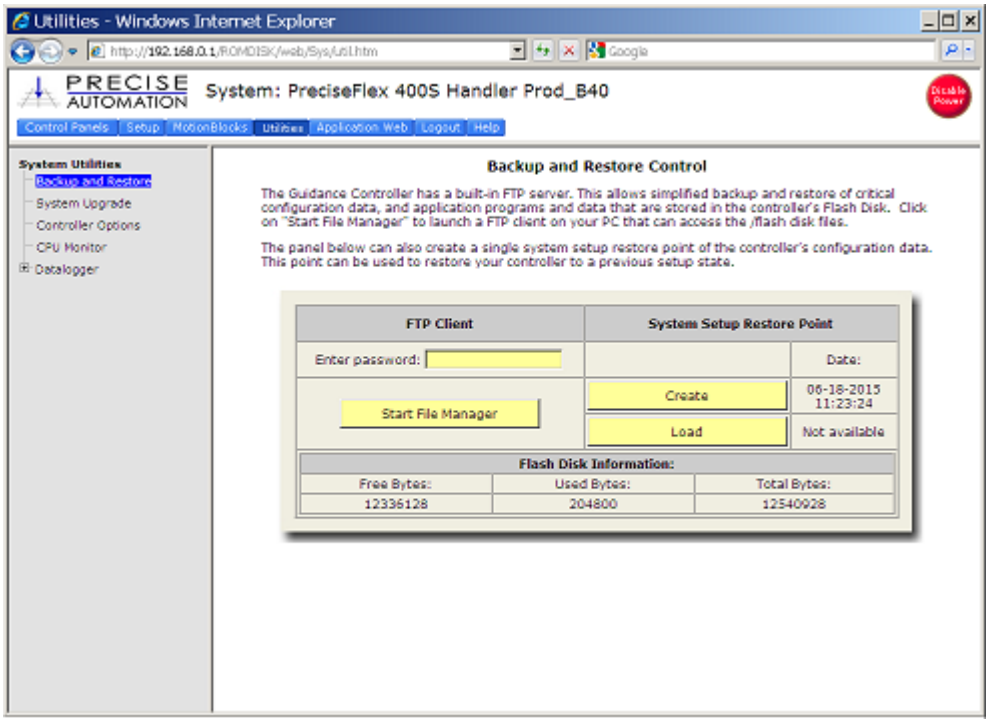
The controller includes a FTP server that permits the flash disk to be accessed by any FTP client software package. There are several common means of copying a Project into the flash:

- If you have purchased a copy of the Guidance Development Suite (GDS), you can utilize the Guidance Development Editor (GDE) to drag and drop a Project from a PC's hard drive into the flash disk.

- If you need to load several controllers with software, GDS includes a free Guidance Update Wizard that can be utilized.
- If you have any standard FTP client software package, you can access the controller by its IP address (typically 192.168.0.1) in "anonymous" mode with no password and navigate to the /flash/projects folder to copy Projects into the flash.
- Otherwise, you can utilize the FTP client that is built into most browsers to drag and drop Projects into the flash.

The last method of using the browser to copy a Project into a controller's flash will now be described.

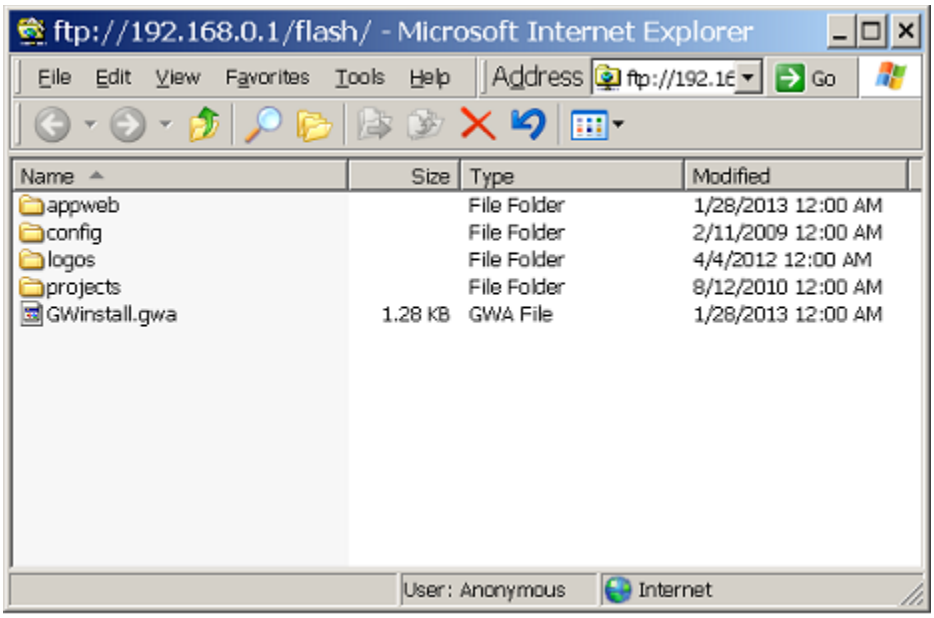
» Open the Controller Backup and Restore page, ***Utilities > System Utilities > Backup and Restore***. The web page should look as follows:



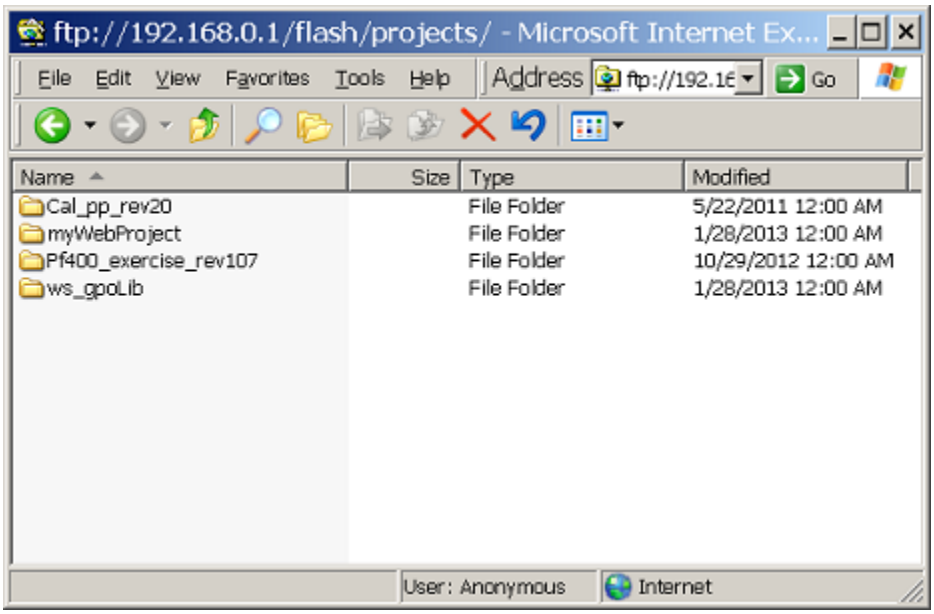
» Press the "***Start File Manager***" button in the right panel. This will force the browser to launch an FTP client.

» If your browser provides an intermediate window with instructions on how to bring up and open the flash in FTP, **follow the browser's instruction to open the flash disk in FTP**.

An FTP client window should now be displayed that presents the names of the top-level folders in the /flash disk. This window should look as follows.



» Click on the "***projects***" folder to navigate down to the area where all GPL Projects must be located. The FTP client window should now look as follows.



In most cases, each folder in the "projects" folder of the flash disk will contain a single complete GPL Project. A Project can consist of multiple files to help organize the software and its global data. In the picture above, "Cal_pp_rev20" is the standard GPL utility program that is executed to calibrate the zero positions of each axis of a robot.

In some cases, folders in "projects" contain libraries of routines that can be accessed by several GPL Projects. In the picture above, "ws_gpolib" is such a folder.

» Finally, to copy an existing GPL Project into the controller's flash disk, **simply drag and drop the GPL Project folder into the "projects" folder of the flash**

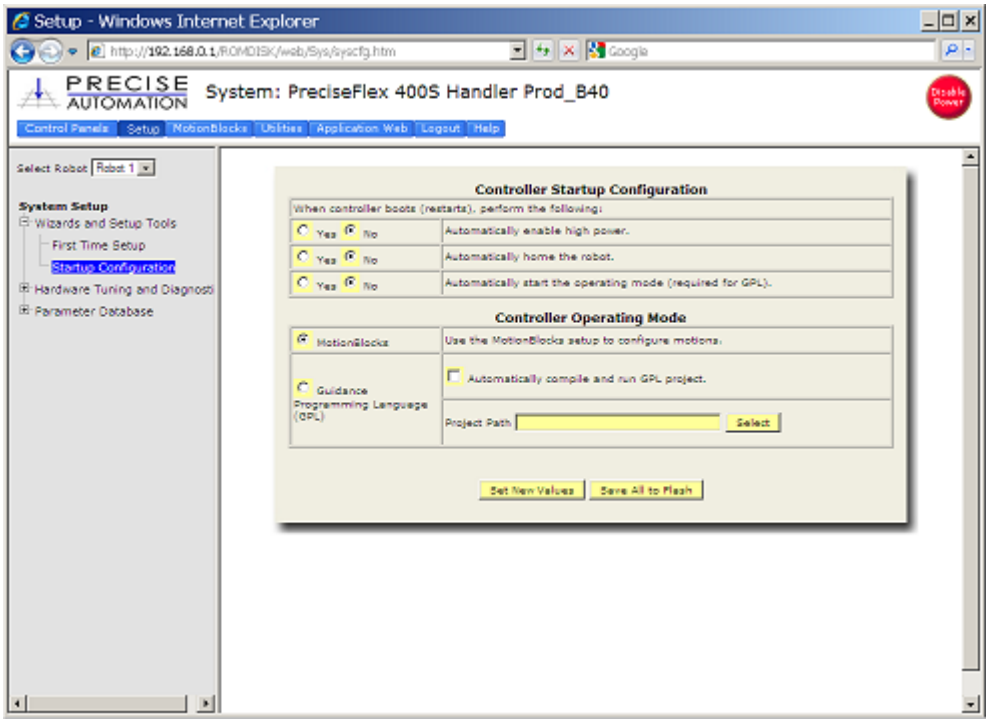
You have now successfully copied a GPL Project into the flash where it can be easily loaded into memory and executed.

Executing a GPL Project

Once a GPL Project has been copied into the controller's flash disk, execution of the Project is simple and is very similar to executing a MotionBlocks program. In this section, the steps for executing such a Project are described.

In order to execute a GPL Project, the controller must be configured for GPL execution instead of MotionBlocks.

» Open the Controller Startup Configuration page, ***Setup > Wizards and Setup Tools> Startup Configuration***. The web page should look as follows:



» Press the red "***Disable Power***" button at the top of the page. This is required because the changes that you will make are not permitted when power to the robot is enabled.

» In the "Controller Operating Mode" panel, click on the "***Guidance Programming***

Language (GPL)" radio button.

» Press the "***Set New Values***" button to store this setting into memory.

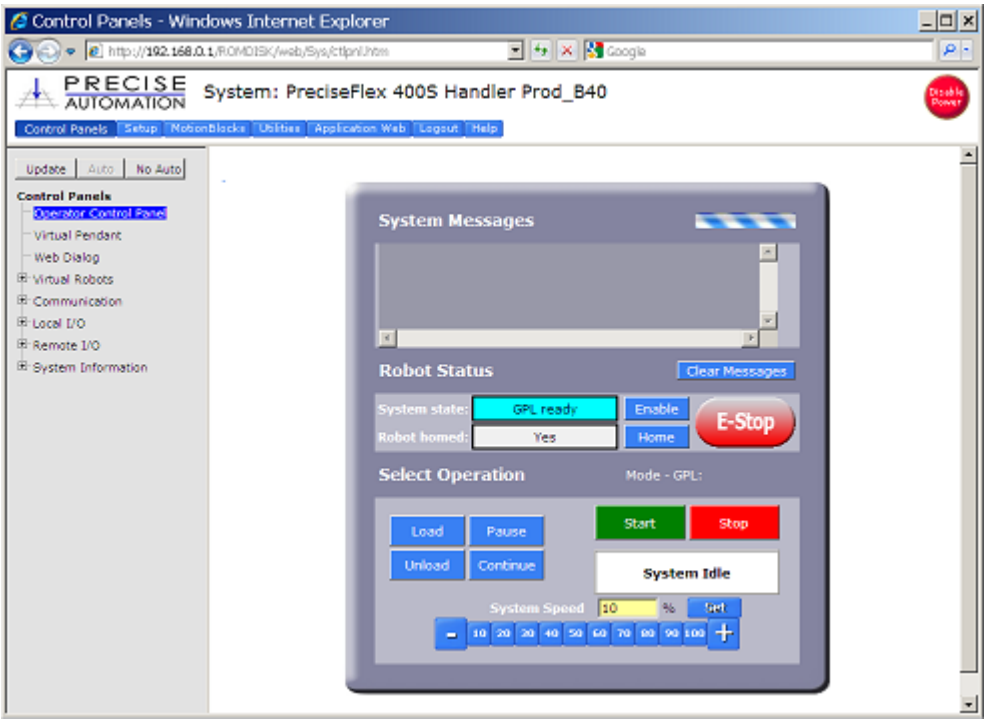
» Press the "***Save All to Flash***" button to store this setting in the flash disk. This ensures that this setting will remain in effect if the controller is restarted. The button will flash for 10-30 seconds as the data is being written. **DO NOT TURN OFF YOUR CONTROLLER WHILE THE BUTTON IS BLINKING SINCE THIS MAY CORRUPT THE FLASH DISK.**

You should note that when you selected GPL execution mode, the ***"Automatically start the operating mode"*** option was also set to ***Yes***. With these two settings, each time the robot's power is enabled, the controller is put into a state where it is ready to allow a GPL Project to take control of a robot.

As with MotionBlocks execution, in order for the controller to operate the robot in an automatic mode, the robot must have already been homed. Assuming that the robot is homed, the controller is now ready to execute a GPL Project. Execution is controlled via the Operator Control Panel.

» Open the Operator Control Panel, ***Control Panels > Operator Control Panel.***

» Press the "***Enable***" button to turn power to the robot on. After power cycles through its various states, the web page should look as shown below.



The Operator Control Panel for executing GPL Projects is very similar to that for executing MotionBlocks. The "System Messages" and the "Robot Status" are identical and operate in the same manner as previously described. Also the "Select Operation" is very similar and contains a "System Speed" that allows you to slow down the robot until you are confident that the GPL Project is correct.

The major difference is that the System State reads "GPL ready" and the operations are: "Start", "Stop", "Load", "Pause", "Unload", and "Continue". Unlike MotionBlocks that has a single set of data, your system can execute one of several GPL Projects.

To execute a GPL Project, perform the following steps:

» Press the *"10"* button in the System Speed strip to operate the robot at 10% of the normal program speed

» Press the ***"UnLoad"*** button. This ensures that no GPL project is currently selected for execution.

» Press the "***Load***" button. This displays a popup list of Projects that are in the flash disk and available for execution.

» In the popup display, click on the Project that you would like to execute and press "***Select***".

» When you are ready to execute the Project and have the robot move, press the "***Start***" button.

Congratulations! You have successfully started the execution of a GPL Project.

If you wish to stop the GPL Project at any time, please do the following:

» To stop the Project at any time, press the "***Stop***" button that is to the right of the Start button. The Project will terminate execution at the conclusion of the currently executing statement or motion. Once a Project has been "stopped", you must "start" the Project from the beginning.

» Alternately, you can press the ***"Pause"*** button. The Project will terminate execution at the conclusion of the currently executing statement or motion. A Project that has been "paused" can be "continued" from the point that it has been paused.

» Also, you can press the red "***Disable Power***" button. This will immediately halt any robot motion, but power will have to be re-enabled when you wish to continue

Once you have verified that the program is operating properly, you can gradually increase the speed by: pressing the "+" key, pressing one of the speed buttons or by typing a speed into the box with the yellow background. You can increase the speed as the robot is running.

This completes the instructions for executing a GPL Project that has been stored in the controller's flash disk. With this information, you can execute standard utility programs or complete applications that have been previously developed.

For more information on GPL, please see the *"Guidance Programming Language, Introduction to GPL"*. For information about developing GPL Projects, see the *"Guidance Development Environment, Introduction and Reference Manual"*. Both of these documents are provided as individual PDF files and as part of the *Precise Documentation Library*.