

Techno **CNC** Systems

Computer Numerical Control for Windows
Techno CNC Interface V2015.1 ®

User's Guide

The screenshot displays the Techno CNC Interface V2015.1 software. The main window is divided into several sections:

- Top Left:** A 2D plot showing the toolpath for a part, with X and Y axes ranging from 0.0 to 1.2.
- Top Right:** A 3D view of the part being machined, showing the toolpath in blue and the part in red.
- Bottom Left:** A list of G-code lines, with line 5525 highlighted. The list includes coordinates for X, Y, Z, and A/B axes.
- Bottom Center:** A Spindle RPM control panel with a vertical scale from -3000 to 3000, a current reading of 2600, and a percentage indicator at 53%. It includes buttons for Spindle (On) and Coolant (Off).
- Bottom Right:** A control panel with buttons for G.Code, Jog, Point, Home, MDI, Probe, and Cycles. It also features a 'Jump To Line' dialog, 'Faster' and 'Slower' buttons, a 'Program Feedrate' of 25.0, a '% Override' of 31.3, and a 'Start' button.
- Right Side:** A parameter table for the current program (G54) and machine settings.

Program G54		Machine	
X	0.0752	X	2.0752
Y	0.1114	Y	1.1114
Z	0.2169	Z	-2.2831
A	0.0000	A	0.0000
B	90.0000	B	0.0000

Relative		Dist To Go	
X	0.0752	X	0.0000
Y	0.1114	Y	0.0000
Z	0.2169	Z	0.0000
A	0.0000	A	0.0000
B	90.0000	B	0.0000

Part Scale: 2.00000000
 Edit Accept Cancel

Tool: #2 - 1/16 Ball End Mill Length...
 Offset: X 0.0000 Y 0.0000 Z 0.0000
 Comp: Positive (G43) Diam: 0.0625
 Cutting Time: 2.501 Replace

Connected

Techno CNC Systems, LLC.

29 Trade Zone Drive
 Ronkonkoma, NY 11779
 Phone (631) 648-7481 ♦ Fax (631) 983-4701
 www.technocnc.com

Revised 04/13/2015
 © 1997-2012 WPI, Inc.

Table of Contents

1. GETTING STARTED	8
THANK YOU	8
PRODUCT SUPPORT.....	8
NEW FEATURES.....	9
SYSTEM REQUIREMENTS	12
INSTALLING THE TECHNO CNC SOFTWARE	12
INSTALLING THE USB DRIVER	13
<i>Windows XP</i>	13
<i>Windows Vista</i>	15
<i>Windows 7</i>	19
<i>Windows 8</i>	22
COMMAND-LINE OPERATION	22
SAFETY AND USAGE GUIDELINES.....	24
ABOUT THIS MANUAL	25
2. MAIN SCREEN FEATURES	27
THE MAIN SCREEN.....	27
PULL-DOWN MENU BAR	28
<i>File Menu</i>	28
<i>Edit Menu</i>	30
<i>Configuration Menu</i>	31
<i>Controller Menu</i>	32
<i>View Menu</i>	34
<i>Coordinates Menu</i>	37
<i>Favorites Menu</i>	38
<i>Help Menu</i>	38
DRO BOX.....	39
<i>Coordinate Systems</i>	39
<i>Set Button</i>	40
<i>Popup Menus</i>	42
VIEWPORT BOX.....	45
CONTROL SELECTION BOX.....	48
<i>G-Code Control Panel</i>	48
<i>Jog Control Panel</i>	50
<i>Point Control Panel</i>	51
<i>Home Control Panel</i>	55
<i>Auxiliary Control Panel</i>	56
<i>MDI Control Panel</i>	57
<i>Probe Control Panel</i>	58
<i>Cycles Control Panel</i>	63
TOOL BOX	68
USER VARIABLES BOX	71
TEACH BOX.....	72
FIDUCIAL BOX.....	72
CONNECTION STATUS BOX.....	75
PROGRAM LISTING BOX	76
DIGITAL OUTPUT BOX.....	76
VARIABLE OUTPUT BOX.....	77
3. INITIAL SETUP	79
WINDOWS SETUP.....	79
SOFTWARE SETUP.....	79
<i>The Configuration Files</i>	79

<i>Configuration Wizard</i>	79
<i>Controller Settings</i>	80
<i>Communications Settings</i>	81
<i>Motor Signal Settings</i>	82
<i>Performance Settings</i>	86
<i>Camera Settings</i>	88
<i>General System Settings</i>	89
<i>Basic Definition Settings</i>	92
<i>Advanced Definition Settings</i>	93
<i>Rotary Axes Settings</i>	94
<i>Drive Parameters Settings</i>	98
<i>Backlash Compensation Settings</i>	103
<i>Basic Homing Settings</i>	104
<i>Advanced Homing Settings</i>	106
<i>Feedrate and/Ramping Settings</i>	107
<i>Motion Delay Settings</i>	113
<i>Reference Point Settings</i>	115
<i>Fixture Offset Settings</i>	115
<i>Threading Settings</i>	117
<i>Input Line Settings</i>	117
<i>Output Line Settings</i>	120
<i>Variable Output Settings</i>	123
<i>M-Code Definitions Settings</i>	125
<i>M-Code Execution Settings</i>	127
<i>G-Code Settings</i>	130
<i>User Variable Settings</i>	132
<i>Counter Settings</i>	134
<i>Cutter Compensation Settings</i>	135
<i>Import Settings</i>	135
<i>Cycle Settings</i>	136
<i>Tool Library Settings</i>	136
<i>Tool Life Management Settings</i>	138
<i>Tool Change Settings</i>	141
<i>Tools in Changer Settings</i>	145
<i>Tool Length Sensing Settings</i>	145
<i>Program Zero Sensing Settings</i>	147
<i>Touch Probe Settings</i>	148
<i>Main Screen Settings</i>	149
<i>Viewport Settings</i>	150
<i>Point List Settings</i>	152
<i>Jogging Settings</i>	153
<i>Pendant Settings</i>	154
<i>File Settings</i>	155
<i>Message Settings</i>	155
<i>Security Settings</i>	158
<i>General Preference Settings</i>	160
<i>Setting Machine Zero</i>	161
4. SYSTEM PROGRAMMING	163
OPENING A G-CODE PROGRAM	163
IMPORTING A DXF FILE	163
USING THE PROGRAM EDITOR	168
<i>File Menu</i>	169
<i>Edit Menu</i>	170
<i>Help Menu</i>	171
<i>Buttons</i>	171

G-CODES SUPPORTED.....	172
M-CODES SUPPORTED.....	173
OTHER COMMANDS SUPPORTED	174
ADVANCED KEYWORD COMMANDS AND FUNCTIONS	174
KEY PROGRAMMING CONCEPTS	175
<i>Mode</i>	175
<i>Absolute vs. Incremental</i>	177
G AND M-CODE REFERENCE	177
<i>G00 Rapid Tool Positioning</i>	177
<i>G01 Linear Interpolated Feedrate Move</i>	179
<i>G02 Clockwise Circular Feedrate Move</i>	180
<i>G03 Counter Clockwise Circular Feedrate Move</i>	184
<i>G04 Dwell</i>	185
<i>G17, G18, G19 Arc Plane Selection</i>	185
<i>G20, G21 Inch Units and Metric Units</i>	185
<i>G27 Home to Switches</i>	186
<i>G28, G30 Move to Reference Point</i>	186
<i>G29, G29.1 Return from Reference Point</i>	187
<i>G31 Seek Sensor</i>	188
<i>G40, G41, G42 Cutter Compensation</i>	189
<i>G50, G51 Scaling</i>	191
<i>G52 Local Coordinate System</i>	192
<i>G53, G53.1 Linear Move to Machine Coordinates</i>	193
<i>G54-59, G54.1, G92 Set Program Zero Commands</i>	193
<i>M06 Tool Change and T Select Tool Commands</i>	195
<i>G43, G44, G49 Tool Length/Geometry Compensation Commands</i>	195
<i>G73, G80, G81, G82, G83, G85, G98, G99 Drilling Canned Cycle Commands</i>	200
<i>G74, G84 Tapping Canned Cycle Commands</i>	204
<i>G76 Thread Cutting Canned Cycle Command</i>	207
<i>G90 Absolute Positioning Mode</i>	211
<i>G91 Incremental Positioning Mode</i>	211
<i>G120 Sense Tool Length</i>	211
<i>G140, G141 Engraving</i>	211
<i>G150 Increment Counter</i>	213
<i>G151 Set Counter</i>	214
<i>G160 Increment Tool Use Counter</i>	214
<i>G161 Set Tool Use Counter</i>	215
<i>G170 Find Fiducials</i>	215
<i>G179 Cancel Fiducial Correction</i>	215
<i>G180, G181 Safe Envelope Commands</i>	215
<i>G190 Set PWM/PFM Output</i>	216
<i>G210 Select Variable Output Port</i>	216
<i>M00 Program Pause</i>	217
<i>M01 Optional Program Pause</i>	217
<i>M30, M30.1 End of Program</i>	217
<i>M98, M99, M02 Subroutine Commands</i>	217
<i>M100, M101 Wait for Input Line</i>	220
<i>M03, M05, M07, M08, M09, M50, M51, MXX Auxiliary Device Control</i>	221
<i>F, G93, G94 Feedrate Commands</i>	221
<i>S Set Variable Output Port</i>	222
<i>Program Comments</i>	223
<i>Optional Line</i>	223
<i>BEGINGRID, ENDGRID</i>	223
ADVANCED PROGRAMMING REFERENCE.....	225
<i>Values</i>	225
<i>Variables</i>	225

<i>Runtime Variables</i>	233
<i>Operators</i>	235
<i>Flow Of Control</i>	239
<i>File Access</i>	244
<i>G-Code Data Files</i>	245
<i>Feedrate Override Commands</i>	248
<i>Saving and Restoring the System State</i>	249
<i>Mathematical Functions</i>	251
<i>Other Functions</i>	257
5. TUTORIAL	259
STARTING TECHNO CNC.....	259
CONFIGURING TECHNO CNC.....	260
LOADING A G-CODE FILE.....	261
VIEWING THE TOOLPATH.....	262
ANIMATING THE G-CODE FILE.....	264
EDITING A G-CODE FILE.....	264
CONNECTING WITH THE SIGNAL GENERATOR.....	265
USING THE JOG CONTROLS.....	266
SETTING MACHINE ZERO.....	267
USING THE POINT MOVE.....	268
SETTING PROGRAM ZERO ON THE MACHINE TOOL.....	269
TESTING THE PROGRAM ON THE MACHINE TOOL.....	270
CUTTING THE PART.....	270
EXITING THE PROGRAM.....	271
TURNING OFF THE ELECTRONICS.....	271
6. REMOTE CONTROL INTERFACE	273
GETTING STARTED.....	273
FUNCTIONS.....	274
MESSAGES.....	275
7. GLOSSARY	277

1. Getting Started

Thank You

Thank you for purchasing Techno™ CNC, the innovative, powerful CNC control system for Windows® XP, Vista, 7 and 8. No control system is easier to set up and use. With intuitive controls and real time graphics, Techno CNC lets you make parts quickly and accurately on your machine tool.

We are extremely committed to the excellence and ongoing enhancement of Techno CNC. Feel free to call us with any comments or questions.

Product Support

We provide expert technical support for all of our products. We have many resources dedicated to helping you resolve your problems quickly. Please use these resources in the following order:

1. Web Site: support.technocnc.com

Our website has product specifications, documentation and a dedicated support section containing Trouble Shooting Guides and FAQ's, and a link to our user contact form: www.technocnc.com/support_mail.htm

2. Dealer Support:

If you purchased Techno CNC from a dealer or other machine tool manufacturer, please contact them as they will have the best knowledge of your complete system.

3. E-mail: support@technocnc.com

E-mail is the most organized way to convey your issues to our support staff. In your e-mail, please state your problem completely. Include your Techno version, the processor and speed of your computer, your version of Windows and your Signal Generator serial number. Attach your Setup and Tooling files (usually found in a folder named c:\Techno data) and, when appropriate, the G-Code file with which you are having problems. Alternatively, you can attach a single Techno support file generated by the Build Support File command on the Help menu. The support file is in ZIP format and contains all relevant files needed by technical support to resolve your issue. Please see "Help Menu" in the Main Screen Features section for more details.

4. Phone/Fax Support

If e-mail is unavailable to you, please call one of our telephone support numbers. We will normally respond to your call within 24 hours.

Techno CNC Systems Office:

Phone: (631) 648-7481 (8:30 AM-5:00 PM, EST, M-F)

Fax: (631) 983-4701

New Features

Techno 4 has many improvements and new features. Some of the new features in Techno 4 are:

- Full 5-axis support (simultaneous or indexing toolpaths)
- Tool life management
- User variables (editable on main screen, used in G-Code files)
- Run G-Code file backwards
- Foreign language support (Spanish, Portuguese)
- Support for USB cameras
- Fiducial correction system
- New controls for teaching a series of points
- Load and run large files much more efficiently (loads faster, uses much less RAM)
- Canned cycles for tapping holes (G74/G84)
- G-Codes for engraving, including support for custom-designed fonts and multiple engraving spindles (G140/G141)
- Branch G-Code based on state of input lines (using IF-THEN blocks)
- Added AND and OR keywords to advanced G-Code commands
- Added math functions to G-Code: sin, cos, tan, asin, acos, atan, pow, mod, sqrt, abs, round, rounddown, roundup, ln, exp, log and adp
- Added G-Code function to set feedrate override percent (SetFO)
- Added G-Code commands to write information to a file (CreateFile, OpenFile, WriteLine)
- Added option for G-Code file to read points from separate data file

- Cycles control panel provides simple CAM dialogs for
 - Pocketing (rectangle, circle, polygons)
 - Facing
 - Engraving (any TrueType font)
 - Drilling/punching/tapping holes
 - Threading
 - Any custom cycle added to the system as a plug-in (written in any .NET language)
- Option for output lines always displayed on main screen
- Option for main screen buttons that run G-Code macros
- Spindle speed slider active while file is running
- Very accurate runtime estimate
- Configuration wizard
- Independently home multiple motors driving a single axis
- Ability to jog an axis while G-Code is running
- More information displayed when file exceeds machine envelope (display G-Code line number, coordinate, distance beyond envelope)
- New Drive Parameters configuration panel with detailed settings per drive mechanism type
- Recent Setup & Tooling files on File menu
- Advanced contouring (smoothes tangential line-arc transitions and multiple lines)
- Automatic smoothing of linear move sequences in point-to-point G-Code files (G200/G201)
- Support for restricting motion to a safe working envelope defined within the machine envelope
- Configurable automatic delays at corners
- Input lines can trigger loading & running G-Code files
- Improved firmware updating scheme
- Support for Pulse-Width Modulation and Pulse-Frequency Modulation control (slider on main screen or G190 command)

-
- Support for multiple analog, PWM, and PFM outputs (select via G210)
 - Viewport improvements
 - Mouse wheel support
 - Right mouse click menu
 - Clear Trace menu command
 - Zoom window command
 - Direction arrows
 - M-Codes automatically executed
 - At the beginning/end of motion
 - At the beginning/end of ramping
 - Before/after homing each axis
 - When specific input lines are tripped
 - When connecting/disconnecting with the Signal Generator
 - DXF import improvements
 - Smart application of cutter compensation
 - Smart cutting order (cut innermost features first)
 - New options for assigning program zero location relative to DXF coordinate system
 - G90 vs. G91 style G-Code
 - Option to combine DXF layers into single layer
 - Optional return to program zero at end of file
 - More options for inserting additional G-Code
 - Option to turn off most controls and displays on main screen
 - Single-direction 3D scanning (previously only supported zigzag pattern)
 - Homing switches can be anywhere in the machine envelope (don't have to be at boundaries)
 - Support more wiring options for home/limit switches
 - Support for hotwire and wire EDM machines
 - Counters available in G-Code (e.g. to count parts made, G150/G151)
 - Option to automatically raise Z axis after feed hold

- Support for G93/G94 (feedrate interpretation)
- Added safe move performed at specified feedrate (G29.1)
- New G-Code for tool length sensing (G120)
- Improved support for Fanuc G-Code
- Improvements to automatic tool change
- Main screen display units (English/metric) may be different than configuration units
- Support for 'hidden' axes (eg. rotary axis used in tool changer doesn't appear on main screen)
- .NET remote interface (e.g. load and run a file, move to a point, etc, from another program written in any .NET language)
- Support for 64-bit versions of Windows Vista, Windows 7 and Windows 8.

System Requirements

- PC with 1.0 GHz or faster processor.
- Microsoft Windows XP, Vista, 7 or 8, either 32 or 64 bit. Windows 95, 98, ME, NT and 2000 are not supported.
- At least 2 GB of RAM (4 GB preferred for running large G-code files).
- Hard drive with at least 60MB of space available.
- One available USB port (must be USB 2 or higher).
- High-performance graphics hardware to improve scrolling and zooming performance (recommended but not required).

Installing the Techno CNC Software

It's easy to install Techno CNC.

1. Close all applications.
2. Insert the Techno CD into the CD drive. On most PC's, the installation program will start automatically after a few seconds, and you can continue with step 3 below.
3. If your PC is configured not to start CD installations automatically, double-click the program icon labeled 'Setup.exe'.
4. Follow the on-screen instructions.

If you are currently using Techno CNC version 1, 2 or 3, note that Techno CNC 4 will be installed completely separate from your previous installation and will not affect it in any way.

Installing the USB Driver

You'll need to install the Techno USB driver so the software can communicate with the Signal Generator. Please follow these instructions to install the driver.

1. Make sure the Techno software is installed.
2. Connect the Signal Generator to the PC using an A-B USB cable.
3. Connect the power adapter to the Signal Generator.
4. Turn on the Signal Generator.

To finish the driver installation, follow the steps below for your version of Windows.

Windows XP

Windows will display the Found New Hardware wizard.



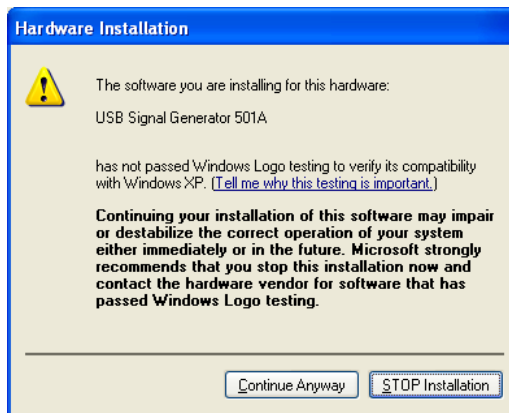
1. Choose 'No, not this time', then click Next.

Windows will display the following dialog box.



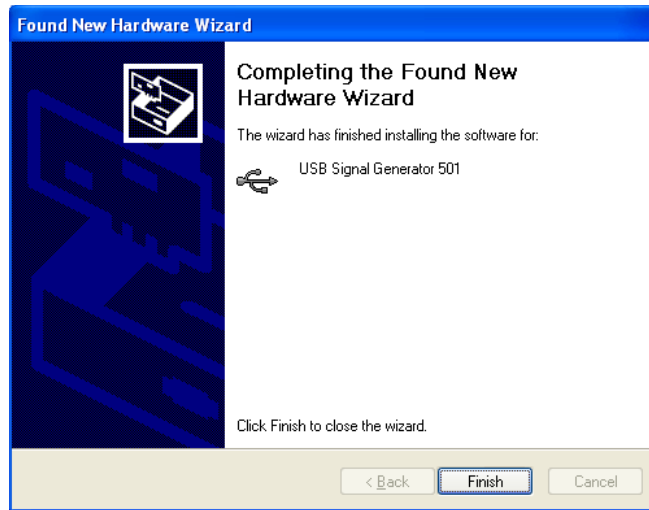
2. Choose 'Install the software automatically (Recommended)', then click Next.

Windows will display the following dialog box.



3. Click 'Continue Anyway.'

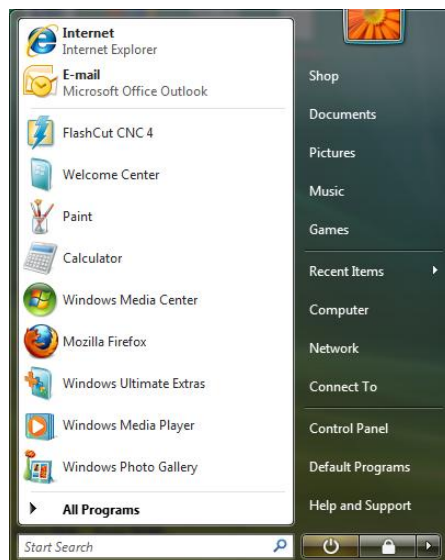
Windows will take a few moments to install the driver, then display this dialog box to tell you that driver installation is complete.



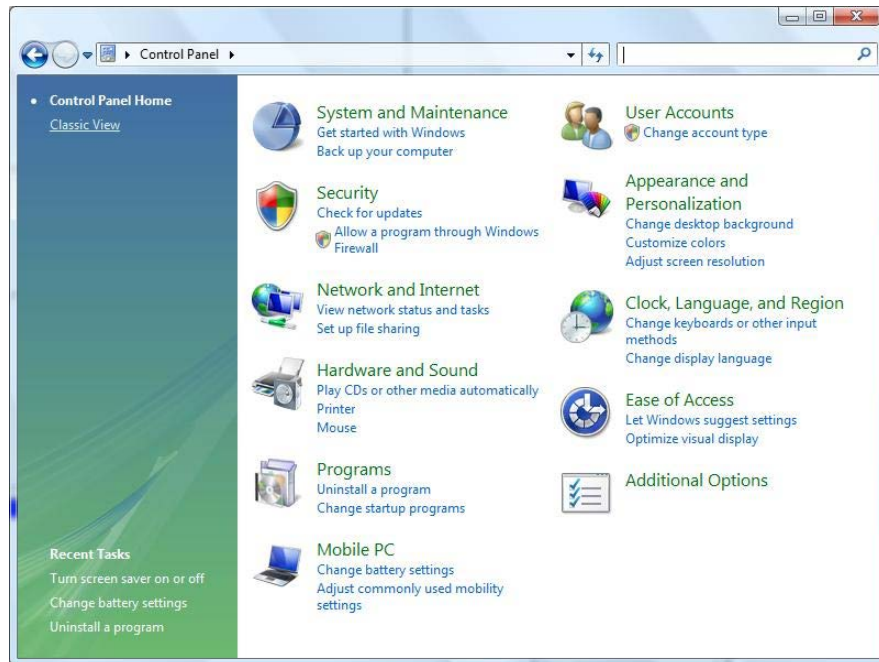
Windows Vista

Windows will automatically find and install the driver for the USB Signal Generator. If Windows is unable to locate and install the drivers, proceed with the following steps in order to properly install the driver.

1. Click Start and select Control Panel.

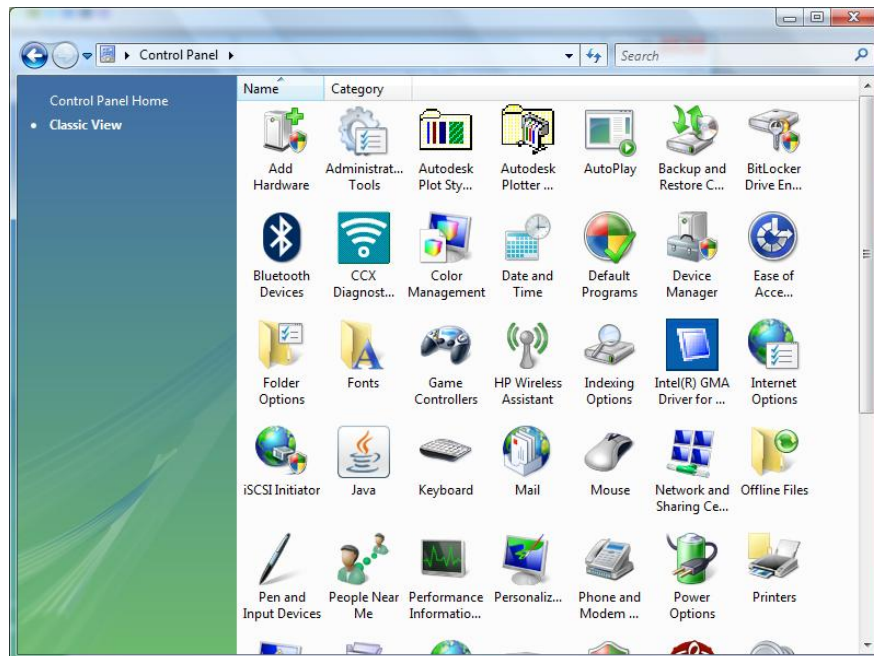


Windows will display the following dialog box.



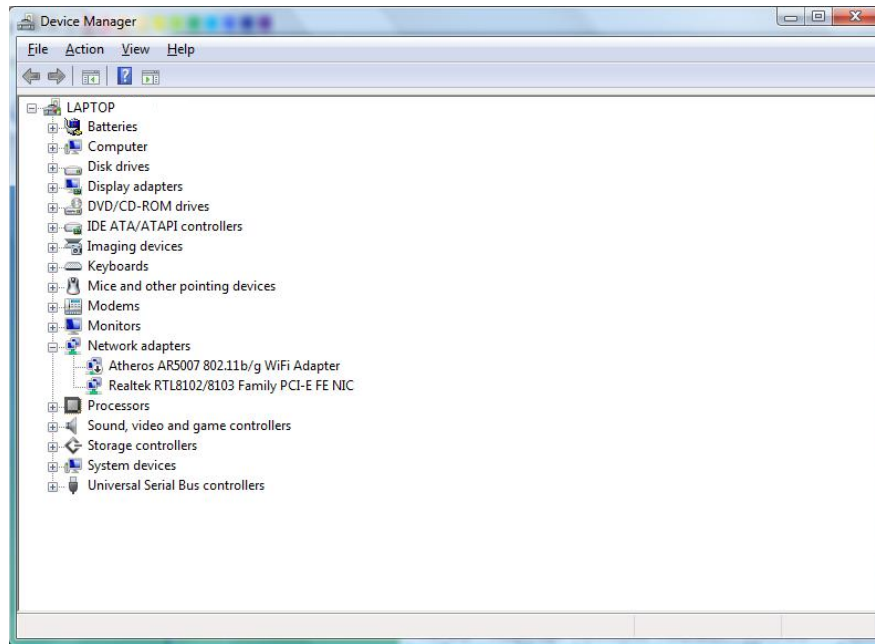
2. Choose Classic View at the top left corner of the dialog box.

Windows will display the following icons.



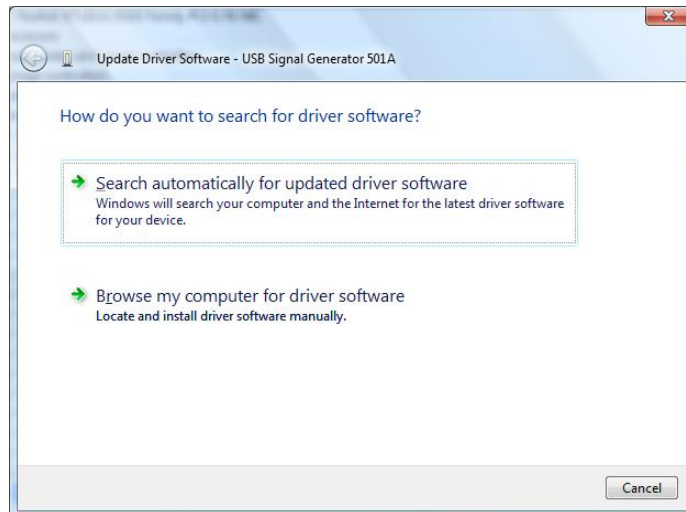
3. Choose Device Manager.

Windows will display the following dialog box.



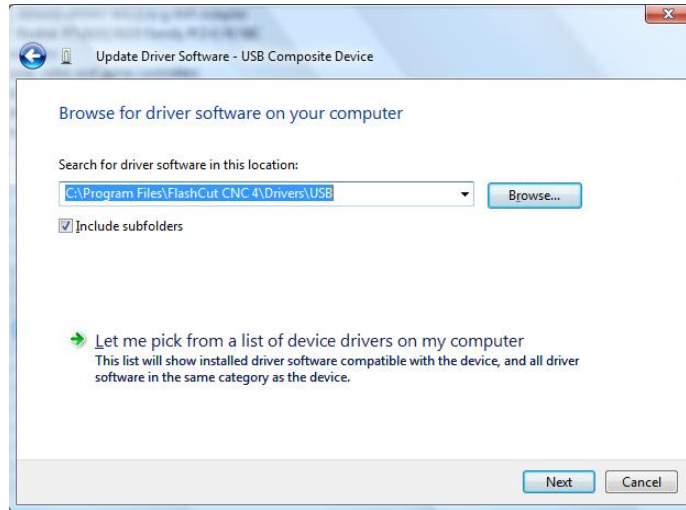
4. Expand the 'Universal Serial Bus Controllers' listing.
5. Right click on 'USB Signal Generator 501A' and select 'Update Drivers'.

Windows will display the following dialog box.



6. Choose 'Browse my computer for driver software'.

Windows will display the following dialog box.



7. Click Browse and navigate to the following folder:

32-Bit Windows Vista

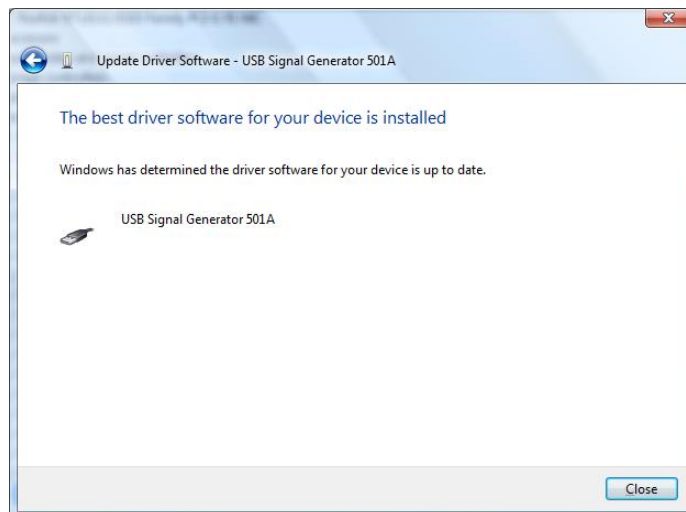
C:\Program Files\Techno CNC 4\Drivers\USB

64-Bit Windows Vista

C:\Program Files (x86)\Techno CNC 4\Drivers\USB

8. Click Next.

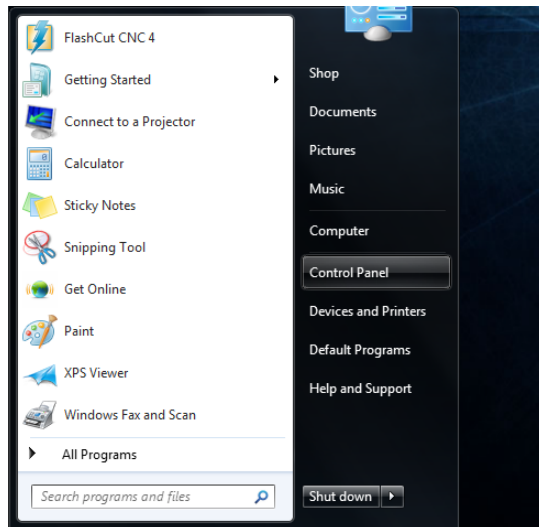
Windows will take a few moments to install the driver, then display this dialog box to tell you that driver installation is complete.



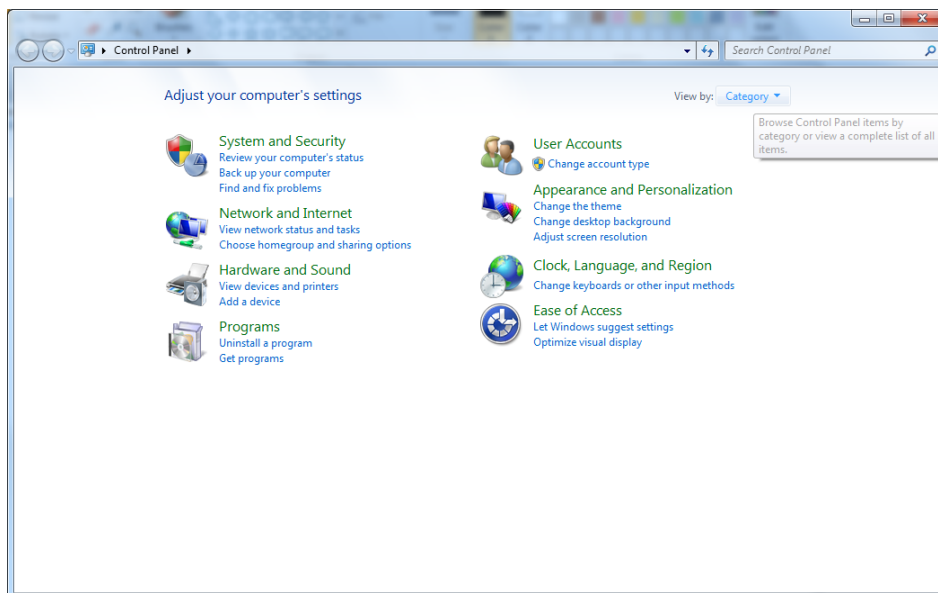
Windows 7

Windows will automatically find and install the driver for the USB Signal Generator. If Windows is unable to locate and install the drivers, proceed with the following steps in order to properly install the driver.

1. Click Start and select Control Panel.

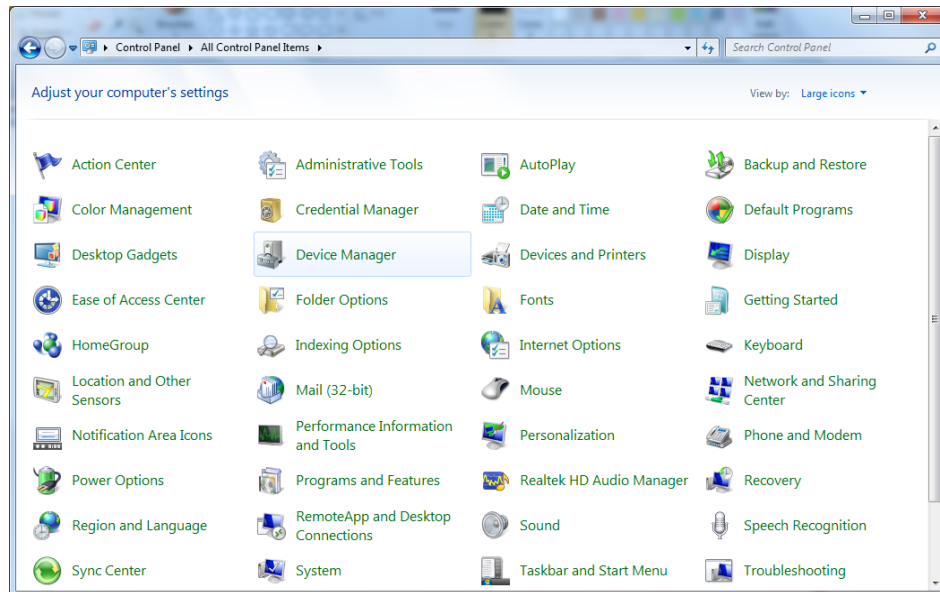


Windows will display the following dialog box.



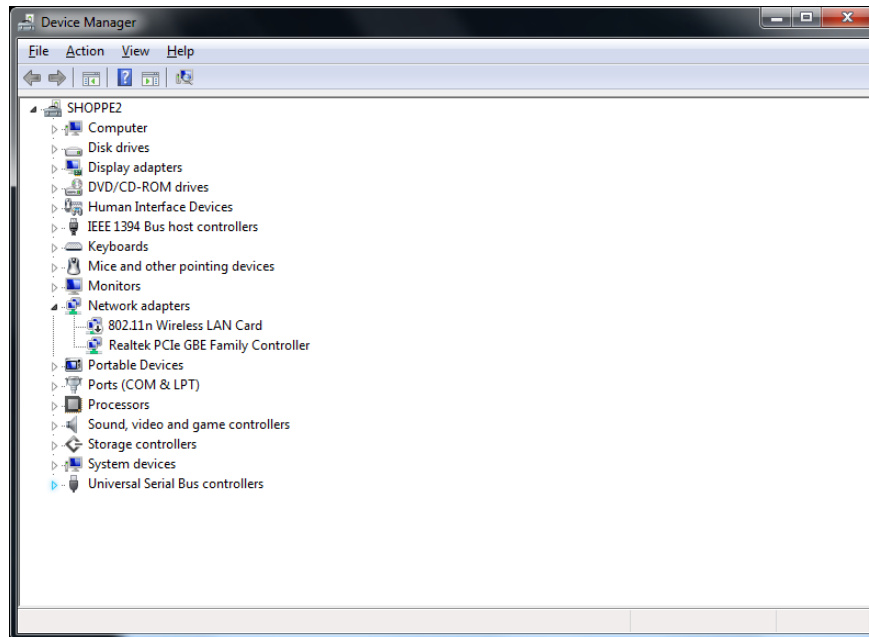
2. At the top right corner of the dialog box, click View By and select Large Icons.

Windows will display the following icons.



3. Choose Device Manager.

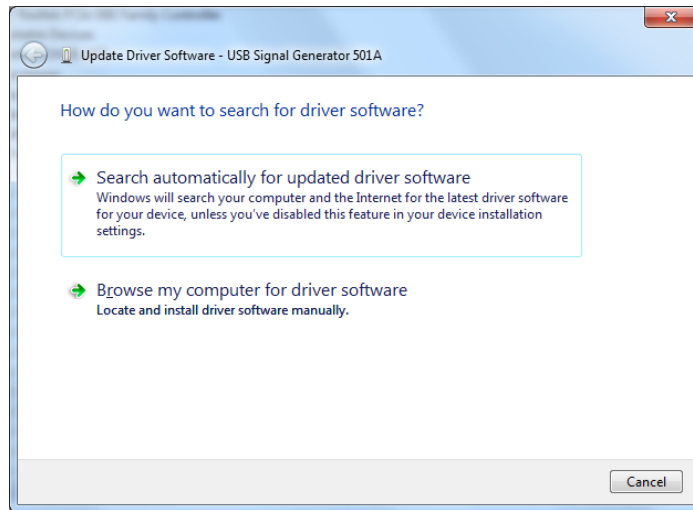
Windows will display the following dialog box.



4. Expand the 'Universal Serial Bus Controllers' listing.

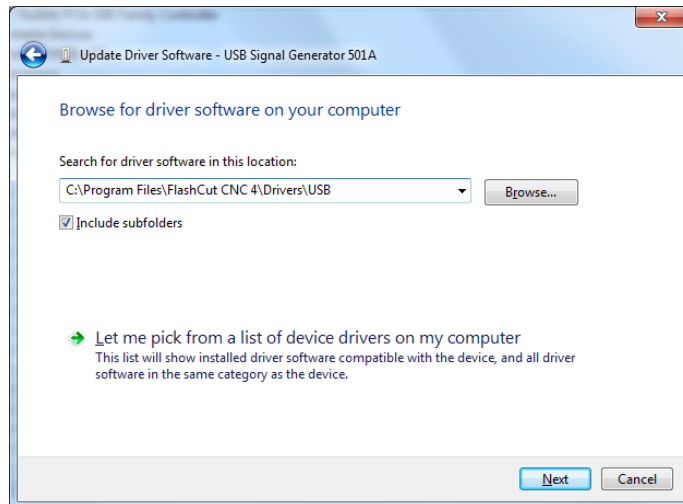
5. Right click on 'USB Signal Generator 501A' and select 'Update Driver Software'.

Windows will display the following dialog box.



6. Choose 'Browse my computer for driver software'.

Windows will display the following dialog box.



7. Click Browse and navigate to the following folder:

32-Bit Windows 7

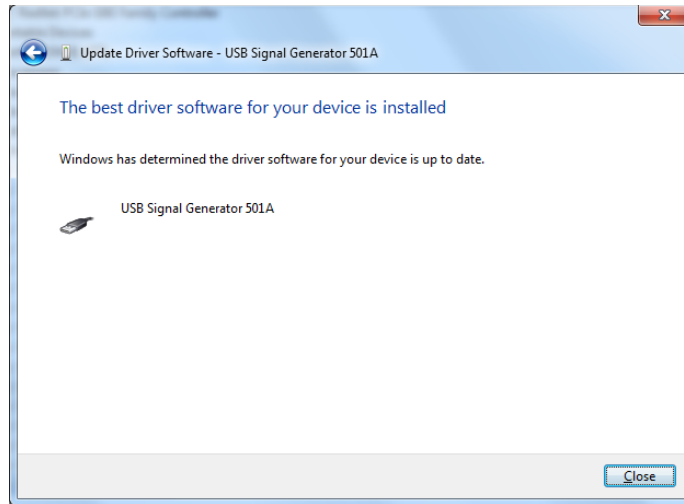
C:\Program Files\Techno CNC 4\Drivers\USB

64-Bit Windows 7

C:\Program Files (x86)\Techno CNC 4\Drivers\USB

8. Click Next.

Windows will take a few moments to install the driver, then display this dialog box to tell you that driver installation is complete.



Windows 8

Windows will automatically find and install the driver for the USB Signal Generator.

Command-Line Operation

Normally you'll start Techno by double-clicking the program icon on your desktop or selecting it from the Windows Start menu. However, some applications require launching Techno using the command-line.

Two common applications are:

- Integration with CAM software, which can send new G-Code files directly to Techno
- Integration with factory automation software, which can tell Techno to load and begin running a G-Code file

The syntax for starting Techno via the command-line is:

```
[Techno Executable File Spec] "[G-Code File Spec]" [/s]
```

Techno Executable File Specification: The complete path and file name for the Techno executable file. Typically this is:

```
c:\program files\Techno cnc 4\Techno4.exe
```

G-Code File Specification: The complete path and file name for the G-Code file to load.

/s: Optional flag that tells Techno to start running the G-Code file after it's loaded. This flag only works if Techno is already running and connected to the Signal Generator. The function is the same as if the Start button were clicked.

Example:

```
c:\program files\Techno cnc 3\Techno4.exe "c:\Techno  
data\somepart.fgc" /s
```

When Techno is launched via the command line with a G-Code file specified, it checks to see if another instance of Techno is already running. If so, Techno sends the command line parameters to the instance that's already running, then terminates immediately. This lets other software 'send' a G-Code file to Techno when Techno is already running.

Techno also provides a .NET-based remote control interface that allows other programs to send commands to and receive messages from Techno. Please see the Remote Control Interface section for more information.

Safety and Usage Guidelines

When running an automated machine tool, safety is of utmost importance. For proper and safe use of the Techno CNC program and your CNC machine, the following safety guidelines must be followed:

- 1. Never let the machine tool run unattended.**
- 2. Require any person in the same room as a running machine tool to wear safety goggles, and to stay a safe distance from the machine.**
- 3. Allow only trained operators to run the machine tool. Any operator must have:**
 - Knowledge of machine tool operation**
 - Knowledge of personal computer operation**
 - Knowledge of Microsoft Windows**
 - Good common sense**
- 4. Place safety guards around the machine to prevent injury from flying objects. It is highly recommended that you build a safety shield around the entire tool envelope.**
- 5. Never place any part of your body within the tool envelope while the machine has power, since unexpected machine movement can occur at any time.**
- 6. Always keep the tool envelope tidy and free of any loose objects.**
- 7. Be on alert for computer crashes at all times.**

Techno CNC and its affiliates are not responsible for the safe installation and use of this product. You and only you are responsible for the safety of yourself and others during the operation of your CNC machine tool. Techno CNC supplies this product but has no control over how it is installed or used. Always be careful!

Techno CNC is not responsible for damage to any equipment or workpiece resulting from use of this product.

If you do not understand and agree with all of the above, please do not use this product.

About this Manual

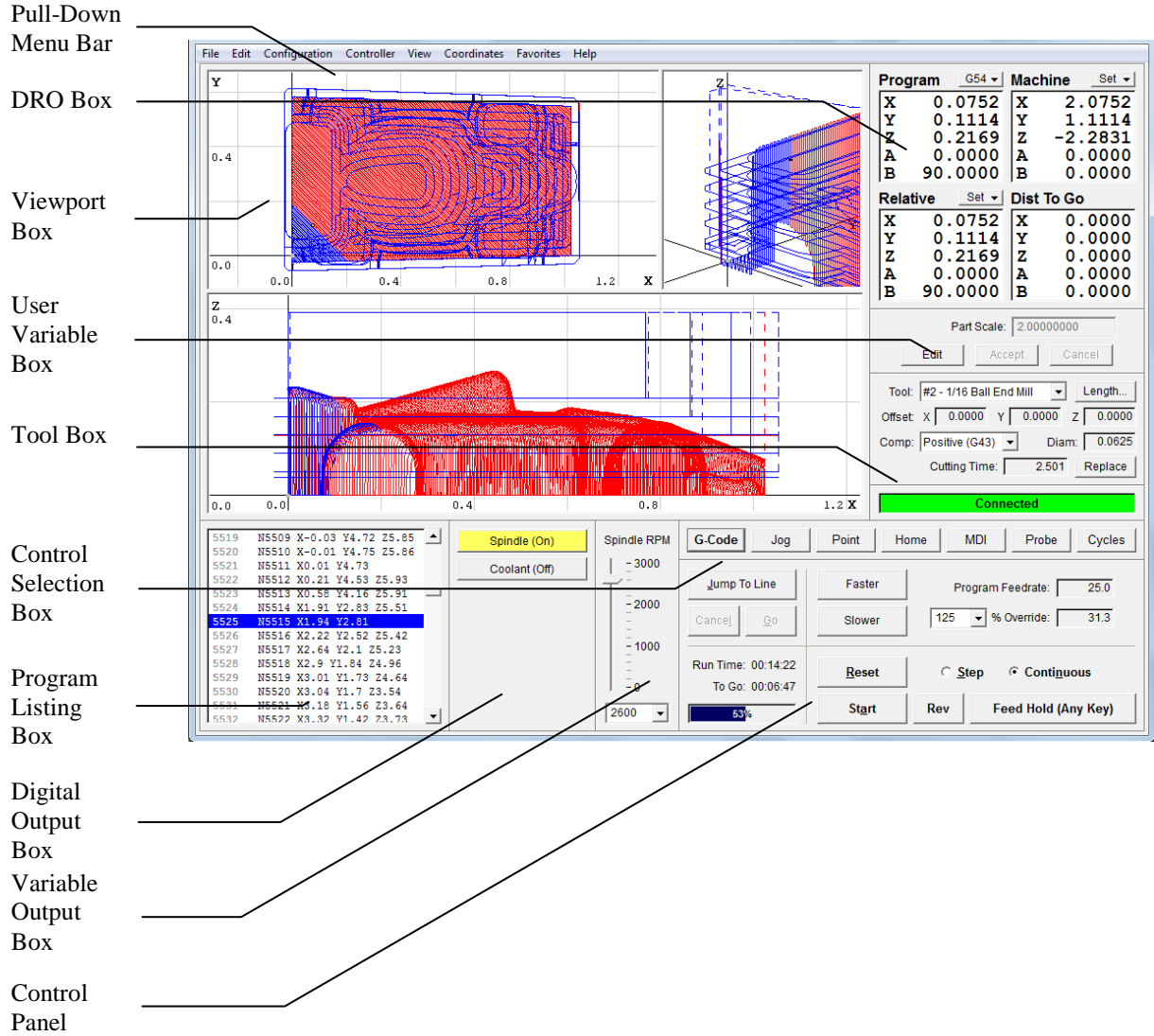
Techno CNC is a unique Windows application, so you'll need some instruction to get started. **Since automated machining is potentially dangerous, please take the time to completely read through this manual to understand operation of the software before running the system.**

It is assumed that you already have a working knowledge of the PC and Windows. If you are not familiar with either of these, please review your PC or Windows user's guides before you use Techno CNC.

2. Main Screen Features

The Main Screen

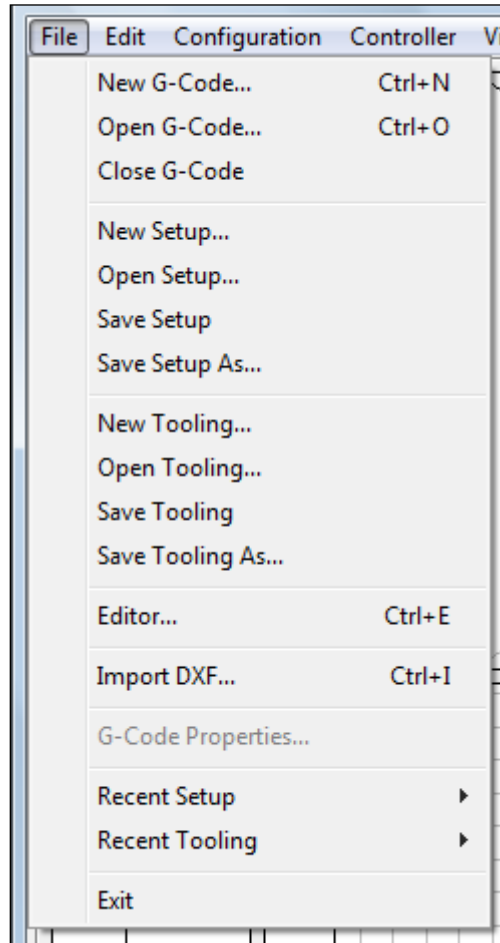
The main screen is shown below. An explanation of each area of the screen follows. Note that Techno has many configuration options that can affect the appearance of the screen.



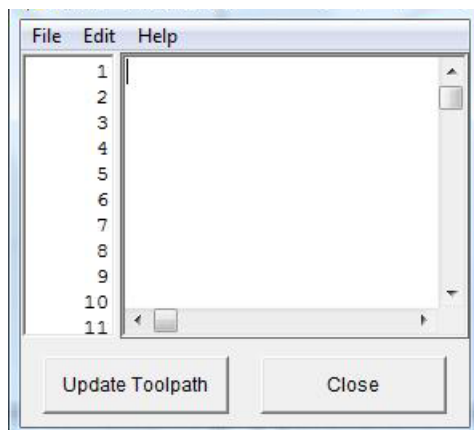
Pull-Down Menu Bar

This area contains the main menu headings for many system commands.

File Menu



New G-Code – Opens the editor dialog box with a new, empty file:



G-Code file names have an “FGC” extension by default. You can change the extension on the Files panel of the Configuration dialog box. Other common extensions used in the industry are “NC”, “TAP” and “CNC.”

Open G-Code – Opens an existing G-Code file, checks the file for errors, draws the toolpath in the Viewport Box, and displays the program in the Program Listing Box.

Close G-Code – Closes the open G-Code file.

Editor – Opens the editor dialog box and displays the current G-Code file. Using this feature you can edit any G-Code file without leaving Techno. Note that you can also double-click the Program Listing Box to open the editor.

New Setup – Creates a new Setup file by running the configuration wizard (see “Configuration Wizard” in the Initial Setup section for more information). The Setup file contains all of the configuration parameters for your machine tool. You can change the parameters using commands in the Configuration menu (described below). Setup file names have an “STP” extension by default.

Open Setup – Opens an existing Setup file.

Save Setup – Saves the current configuration parameters to the current Setup file. This menu command is disabled when the Setup file is “Read Only”.

Save Setup As – Saves the current configuration parameters to a new Setup file.

New Tooling – Creates a new Tooling file. The Tooling file contains all of the tool geometry information for your set of tools. You can change the parameters using the Tooling command in the Configuration menu (described below). Tooling file names have a “TLG” extension by default.

Open Tooling – Opens an existing Tooling file.

Save Tooling – Saves the current tool geometry information to the current Tooling file. This menu command is disabled when the Tooling file is “Read Only”.

Save Tooling As – Saves the current tool geometry information to a new Tooling file.

Import DXF – Converts a 2-dimensional DXF file into standard G-Code, and opens the G-Code file created.

Import DXF – Converts a 2-dimensional DXF file into standard G-Code, and opens the G-Code file created.

G-Code Properties – Displays toolpath travel information for the loaded G-Code file, in program and machine coordinates. If the toolpath goes outside the machine envelope the dialog displays the axis and direction in which the toolpath exceeds the Machine envelope. The dialog also shows the associated line of G-

Code, with a link to edit this line of the file. The dialog also displays the usage for each tool specified in the program.

Recent Setup – Displays recently opened Setup files.

Recent Tooling – Displays recently opened Tooling files.

Recent G-Code – Techno lists recently opened G-Code files at the bottom of the File menu. For information on setting the number of files listed, see “File Settings” in the Initial Setup section.

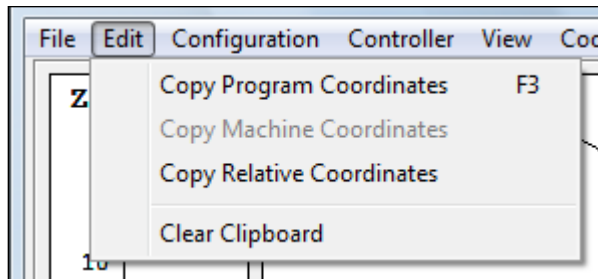
Exit – Disconnects from the Signal Generator, then exits the program. **This command does not automatically turn off the Signal Generator, motor driver or other electronics in your system. You must turn off all electronics separately.**

Techno saves the machine state to a file when exiting the program. The machine state includes the current:

- Machine tool position
- Program Zero location (including the current fixture offset)
- Relative Zero location
- Machine coordinates state (set or cleared)
- Tool loaded
- Tool offset in effect
- Last direction moved (per axis)

The next time you startup Techno, it will optionally load the saved machine state depending on your Save/Restore Machine State settings on the System:General panel of the Configuration dialog box. For more details, see “System General Settings” in the Initial Setup section.

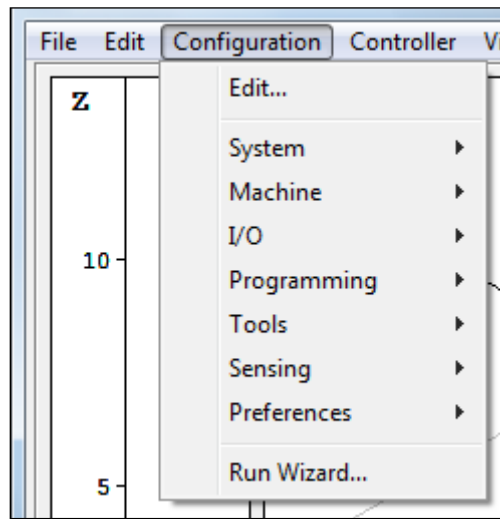
Edit Menu



Copy Program Coordinates, Copy Machine Coordinates, Copy Relative Coordinates – Copies the current coordinates in the DRO to the Windows clipboard. There are several configurable options for this feature. For details, see “General Preferences” in the Initial Setup section.

Clear Clipboard – Clears the Windows clipboard.

Configuration Menu



Edit – Opens the Configuration dialog box and jumps to the last-used panel.

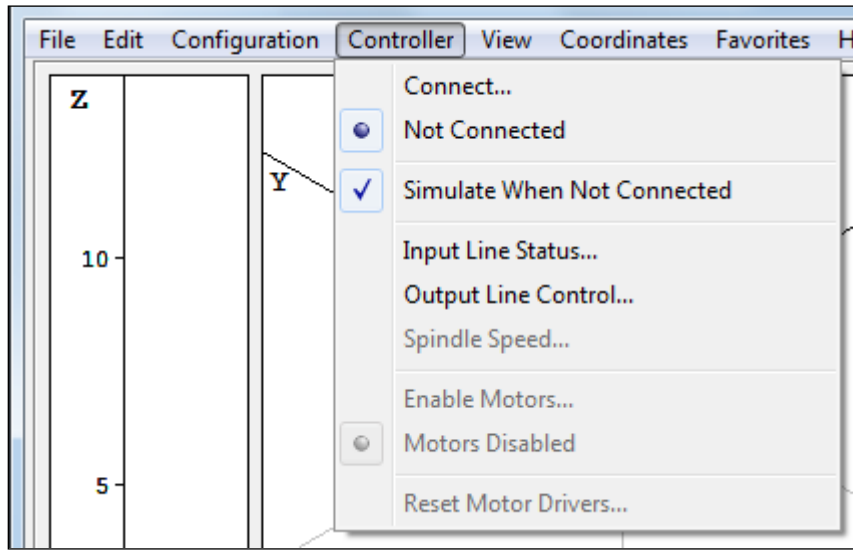
Edit Protected – This command only appears if you’re using a system password. See “Security Settings” in the Initial Setup section for more information.

Configuration Panel Shortcuts – Techno displays a menu command for any configuration panel that has its Display Shortcut checkbox selected (upper right corner).

System, Machine, I/O, Programming, Tools, Sensing, Preferences – These groups contain menu commands that lead to each panel in the Configuration dialog box (except panels that are password protected.)

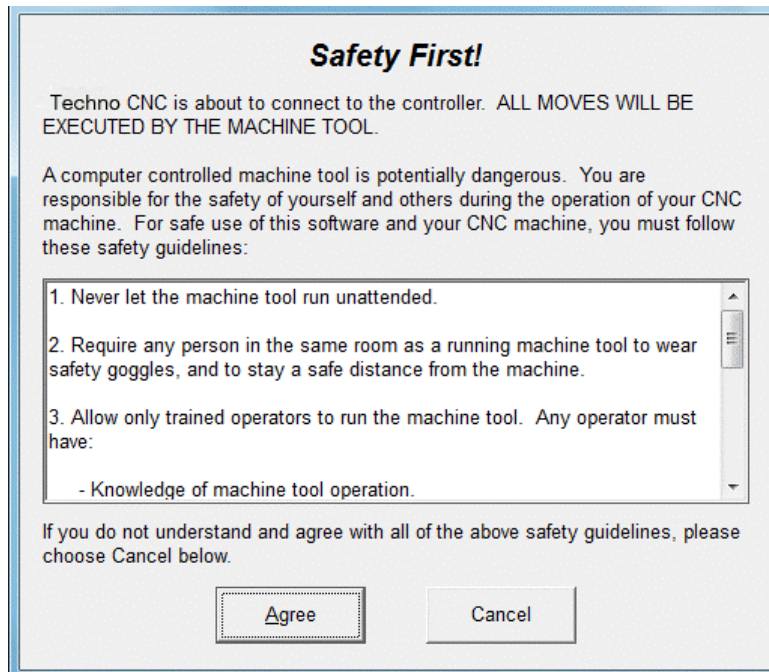
Run Wizard – Launches the configuration wizard. Please see “Configuration Wizard” in the Initial Setup section for more information.

Controller Menu



Connect – Establishes communications with the Signal Generator. When the Signal Generator is connected, all move commands will be executed by the machine tool, and the screen will update in real time. When the Signal Generator is not connected, Techno will not drive your machine tool.

Before the unit connects, a safety reminder screen appears. It is imperative that you and anyone else near the machine understand, agree with and adhere to all of the safety guidelines. If the safety guidelines are not accepted, the software will not connect to the Signal Generator.



Disconnect – Terminates communication with the Signal Generator. In this mode, the screen will update, but the machine tool will not move. When the

system is disconnected you can simulate running a G-Code file, which is useful for debugging before cutting a part.

Simulate When Not Connected – When checked, Techno displays machine motion at accurate speeds when the Signal Generator is not connected. When unchecked, Techno displays machine motion as quickly as your PC will allow.

Input Line Status – Shows the current status of the input lines. Techno displays the following dialog box:

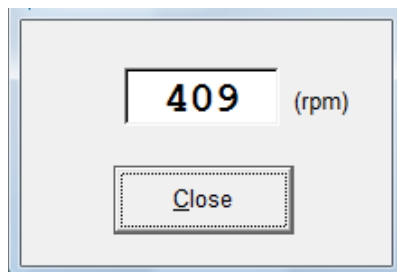
Line	Description	Status	Line	Description	Status
1		Closed	17		Open
2		Closed	18		Open
3		Closed	19		Open
4		Closed	20		Open
5		Closed	21		Open
6		Closed	22		Open
7		Closed	23		Open
8		Closed	24		Open
9		Open	25		Open
10		Open	26		Open
11		Open	27		Open
12		Open	28		Open
13		Open	29		Open
14		Open	30		Open
15		Open	31		Open
16		Open	32		Open

Close

A green status box indicates the switch is in its normal state, while a red status box indicates the switch is in its tripped state (regardless of whether the line is configured as normally open or normally closed.)

Output Line Control – Allows activation and deactivation of all output lines to test functionality.

Spindle Speed – Shows the current speed for a lathe spindle (using encoder feedback). Techno displays the following dialog box:



This feature is only supported when the I/O Expansion Board is installed.

Enable Motors – Enables all motors controlled by the system.

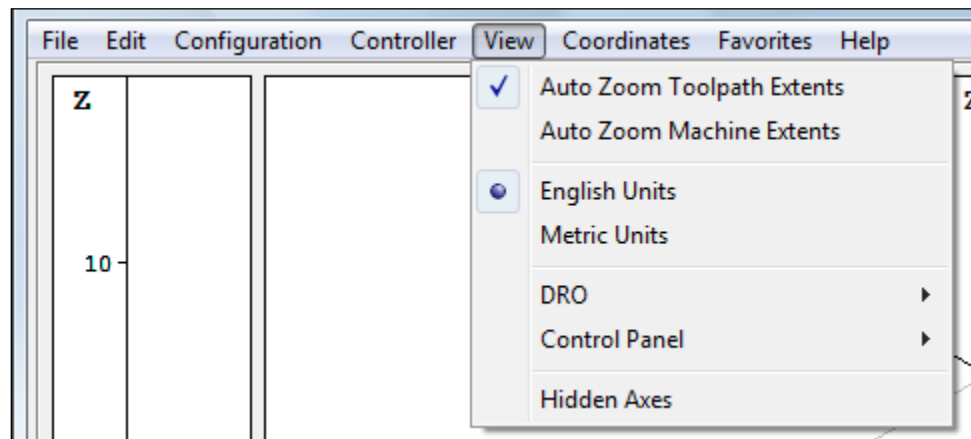
Disable Motors – Disables all motors controlled by the system.

Reset Motor Drivers – Toggles the enable line to reset all motor drivers controlled by the system.

View Menu

The commands at the top of the View menu vary depending on which viewports you're using: standard or advanced (see Viewports under Initial Setup for more information).

Standard Viewports



Auto Zoom Toolpath Extents – Causes the toolpath for the current G-Code File to expand as much as possible within all viewports.

Auto Zoom Machine Extents – Causes the machine envelope to expand as much as possible within all viewports.

English Units – Sets the units of the viewports and DRO's to english even if the system units are metric. (System units are defined on the System:General panel of the Configuration dialog box.)

Metric Units – Sets the units of the viewports and DRO's to metric even if the system units are english.

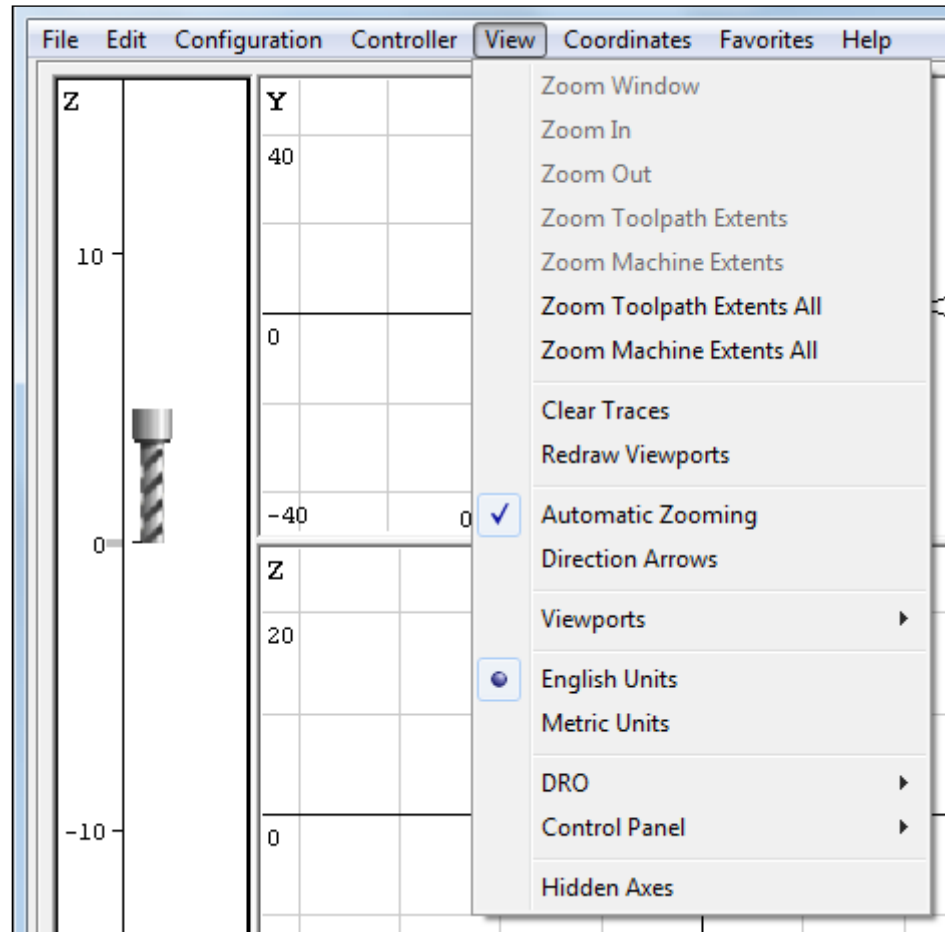
DRO – Selects the display mode for the DRO Box. Choosing Program Coordinates, Machine Coordinates, Relative Coordinates, or Distance To Go Coordinates will expand the chosen DRO into the entire DRO Box as shown below. Choosing All Coordinates will display all four coordinate systems simultaneously in the DRO Box. You can also change these view modes by double-clicking any of the DRO's.

Control Panel – Selects the control panel to display. These commands are functionally identical to the buttons in the Control Selection Box. Their main purpose is to provide keyboard equivalents and shortcuts for those buttons.

See “Control Selection Box” later in this section for a complete explanation of each command.

Hidden Axes – When checked, Techno displays controls on the main screen for axes that are classified as hidden. To learn more about hidden axes see “Advanced Definition” in the Initial Setup section.

Advanced Viewports



Zoom Window – Lets you define the corners of a box whose contents will expand to fill the entire viewport. Use this command to zoom in quickly on a section of the toolpath in the current viewport.

Zoom In – Zooms in one level in the current viewport.

Zoom Out – Zooms out one level in the current viewport.

Zoom Toolpath Extents – Zooms the current viewport to exactly fit the entire toolpath.

Zoom Machine Extents – Zooms the current viewport to exactly fit the entire machine envelope.

Zoom Toolpath Extents All – Zooms all viewports to exactly fit the entire toolpath.

Zoom Machine Extents All – Zooms all viewports to exactly fit the entire machine envelope.

Clear Traces – Removes the blue trace that shows where the machine tool has moved. This lets you see sections of the toolpath that have been obstructed by the blue trace.

Redraw Viewports – When you zoom a viewport, Techno does not automatically regenerate arcs at each zoom level, as it would slow down the zooming process. After zooming in, this command redraws all arcs as smooth curves.

Automatic Zooming – When checked, Techno will automatically zoom to show the entire toolpath in all viewports whenever the G-Code file is reset.

Direction Arrows – When checked, Techno displays arrows on every move to show the direction of travel.

Viewports – Displays a menu that lets you turn viewports on and off.

English Units – Sets the units of the viewports and DRO's to english even if the system units are metric. (System units are defined on the System:General panel of the Configuration dialog box.)

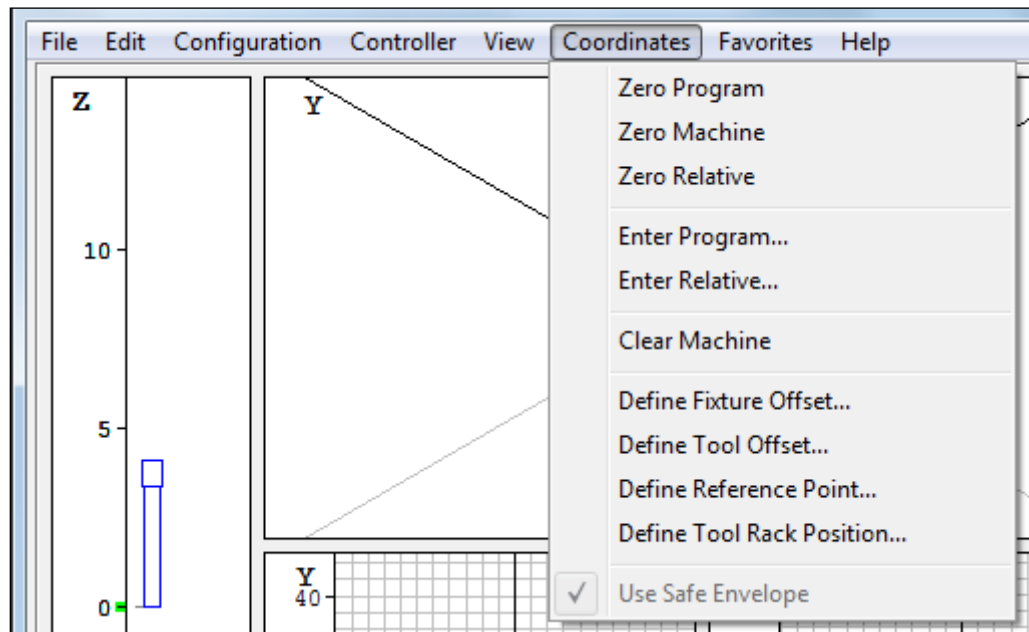
Metric Units – Sets the units of the viewports and DRO's to metric even if the system units are english.

DRO – Selects the display mode for the DRO Box. Choosing Program Coordinates, Machine Coordinates, Relative Coordinates, or Distance To Go Coordinates will expand the chosen DRO into the entire DRO Box as shown below. Choosing All Coordinates will display all four coordinate systems simultaneously in the DRO Box. You can also change these view modes by double-clicking any of the DRO's.

Control Panel – Selects the control panel to display. These commands are functionally identical to the buttons in the Control Selection Box. Their main purpose is to provide keyboard equivalents and shortcuts for those buttons. See “Control Selection Box” later in this section for a complete explanation of each command.

Hidden Axes – When checked, Techno displays controls on the main screen for axes that are classified as hidden. To learn more about hidden axes see “Advanced Definition” in the Initial Setup section.

Coordinates Menu



Most of the commands in the Coordinates menu are functionally identical to the pull-down menu commands available in the DRO Box. The main purpose of this menu is to provide keyboard equivalents and shortcuts for these commands. See “DRO Box” later in this section for a complete explanation of each command listed here.

Zero Program – Same as the Zero All command on the Set button next to the Program coordinates label.

Zero Machine – Same as the Zero All command on the Set button next to the machine coordinates label.

Zero Relative – Same as the Zero All command on the Set button next to the Relative coordinates label.

Enter Program – Same as the Enter command on the Set button next to the Program coordinates label.

Enter Relative – Same as the Enter command on the Set button next to the Relative coordinates label.

Clear Machine – Same as the Clear command on the Set button next to the machine coordinates label.

Define Fixture Offset – Same as the Define Fixture Offset command on the right-mouse-click popup menu available on some DRO's.

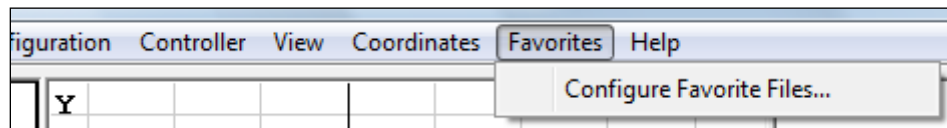
Define Tool Offset – Same as the Define Tool Offset command on the right-mouse-click popup menu available on some DRO's.

Define Reference Point – Same as the Define Reference Point command on the right-mouse-click popup menu available on some DRO's.

Define Tool Rack Position – Same as the Define Tool Rack position command on the right-mouse-click popup menu available on some DRO's.

Use Safe Envelope – At times, because of obstacles in the machine envelope, a more restricted work area must be defined. This more restricted area, called the Safe Envelope, is defined on the Advanced Definition panel of the Configuration dialog box. When Use Safe Envelope is checked, Techno prevents the machine from leaving the Safe Envelope, or from moving at all if it is already outside the Safe Envelope. Also, the G180 and G181 G-Code commands control activation of the Safe Envelope. For more information see “G180, G181 Safe Envelope Commands” in the System Programming section. Note that the Use Safe Envelope menu command overrides the G-Code commands.

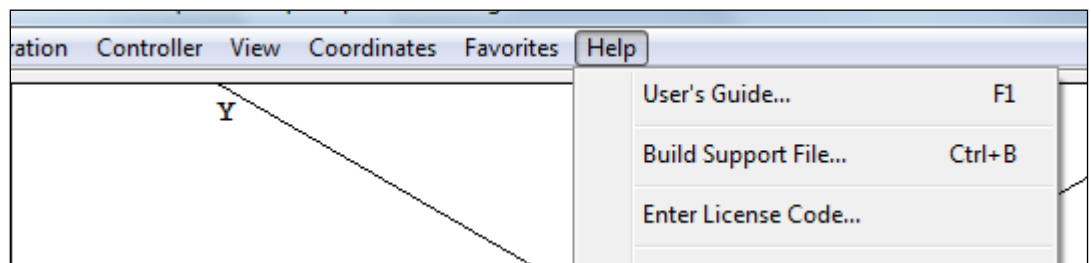
Favorites Menu



The favorites menu provides convenient access to your most often-used G-Code files. On the Files panel of the Configuration dialog box, you may define a folder on your PC as your ‘favorites’ folder. Once you do this, Techno renames the Favorites menu to match your chosen folder, and adds a menu command for every G-Code file contained in the folder. Techno includes subfolders as hierarchical menus.

Configure Favorite Files – Opens the Configuration dialog box to the Files menu.

Help Menu



User's Guide – Displays the Techno CNC User's Guide. You must have the Adobe Acrobat Reader installed (version 4 or higher) to view the User's Guide. Once the User's Guide is displayed, click the Bookmarks tab for easier navigation in Acrobat Reader.

Build Support File – Builds a compressed ZIP file that contains all the information about your system that's needed by Techno technical support to answer a support question. Techno displays the file Save As dialog so you can

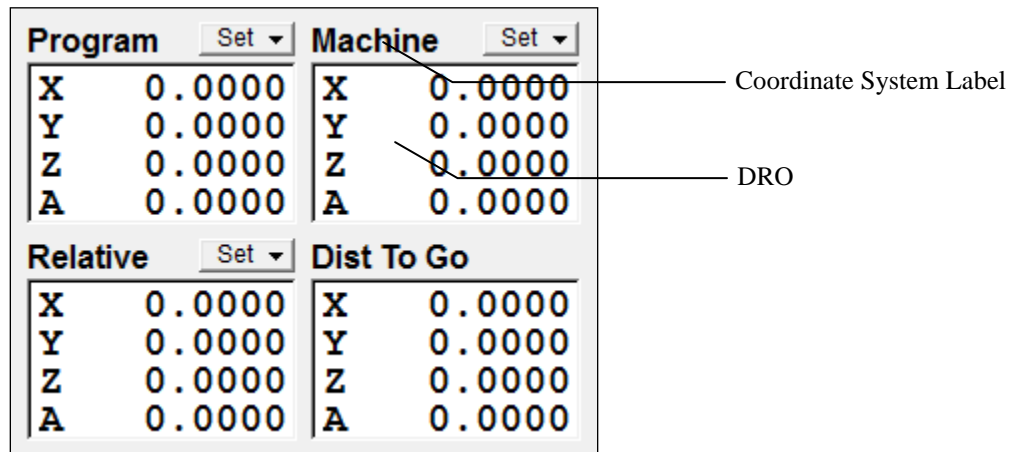
save the support file anywhere that's convenient for attachment to an email (such as the Windows desktop.) The default location is Techno Data\Support.

Enter License Code – Displays a dialog where you may enter the License Code necessary for upgrading your current software version.

About Techno CNC – Shows information about Techno CNC including the software version, Signal Generator properties, the complete path for all current configuration files, and other system information.

DRO Box

The DRO Box shows the current tool position in Program, Machine, Relative and Distance to Go coordinates. It also has a button that sets the coordinate values, and a right-mouse-click pull-down menu available for copying DRO values to the system configuration or the Windows clipboard.



Coordinate Systems

Program

Displays the coordinates of the current position of the tool relative to Program Zero.

Machine

Displays the coordinates of the current position of the tool relative to Machine Zero. This coordinate system is undefined if Machine Zero has not been set (displays "N/A").

Relative

Displays the current relative coordinates. The relative coordinate system is general purpose and may be used for anything you choose. For instance, to measure the distance from any point, zero the relative coordinates at the point from which you want to measure.

Distance To Go

Displays the distance to the ending position of the current move.

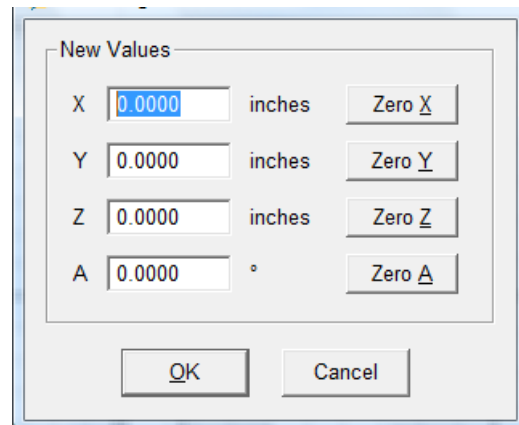
Note that you can double-click any DRO to expand it to fill the entire DRO Box.

Set Button

Each Set button has a pull-down menu containing all the commands that set coordinates to new values. Each coordinate system has its own Set button, and the choices on each Set button are different.

Program Coordinates and Relative Coordinates Set Button – The options are:

- Zero All – Sets all coordinates to zero. In some applications it's convenient to have one axis automatically retract from the workpiece after you set program zero. You can specify this behavior on the System:General panel of the Configuration dialog box. This behavior applies to all Set button commands that zero one or more program coordinates.
- Zero X, Y, Z, A, B, C – Sets one coordinate to zero.
- Enter – Allows setting all coordinates to any value. When chosen, the following dialog box appears:



You may enter new values into the text boxes, or use the Zero buttons to zero axes individually. After you click OK, these coordinates will become the current position of the tool.

- Sense Zero (program coordinates only) – Techno can sense the Program Zero location by moving slowly towards a workpiece until a sensor is tripped. Before using this feature you must first activate and configure it on the Program Zero Sensing panel of the Configuration dialog box. When the Use Program Zero Sensing checkbox is checked, menu commands appear on the Set button for all Activate for Axes checkboxes selected.

Make sure the sensor is wired correctly to the Signal Generator. Follow wiring instructions in the Signal Generator Hardware Guide and use the

Input Line Status dialog to check that Techno recognizes the sensor's tripped and untripped states. (When the sensor is tripped, the Status box should turn red.)

When you choose Sense [Axis] Zero, Techno moves the axis until it hits the sensor, retracts the axis away from the sensor, then sets the axis program coordinate to the sum of the Sensor Offset and the Retract Distance (both are settings on the Program Zero Sensing configuration panel.)

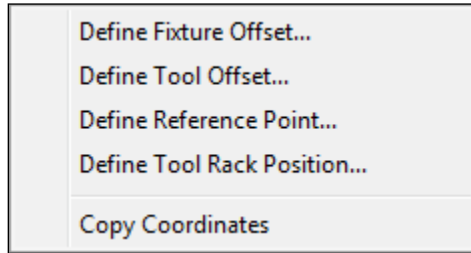
- Fixture Offsets – All fixture offsets that have been defined on the Fixture Offsets panel of the Configuration dialog box appear on this menu (by default there are none defined, so there are no fixture offset menu commands). Choosing a fixture offset activates it and sets the coordinates to new values that reflect the new program zero (or relative zero) location.
- Configure Fixture Offsets – Opens the Configuration dialog box to the Fixture Offsets panel.

Machine Coordinates Set Button – The options are:

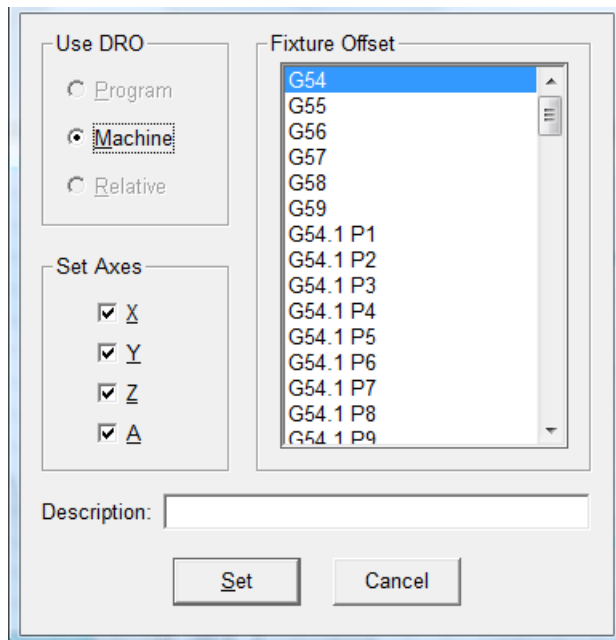
- Zero All – Sets all coordinates to zero.
- Clear – Clears the current Machine Zero setting, which deactivates machine coordinates. This command is useful when you've set Machine Zero and need to change the Machine Zero location.
- Set to Home – Sets all coordinates to the Home Position Machine Coordinate. This option appears only if the Home Position Machine Coordinate on the Advanced Homing configuration panel has been defined (at least one value is nonzero).

Popup Menus

A right mouse click on any DRO (except Distance to Go) displays a popup menu similar to the one shown below.

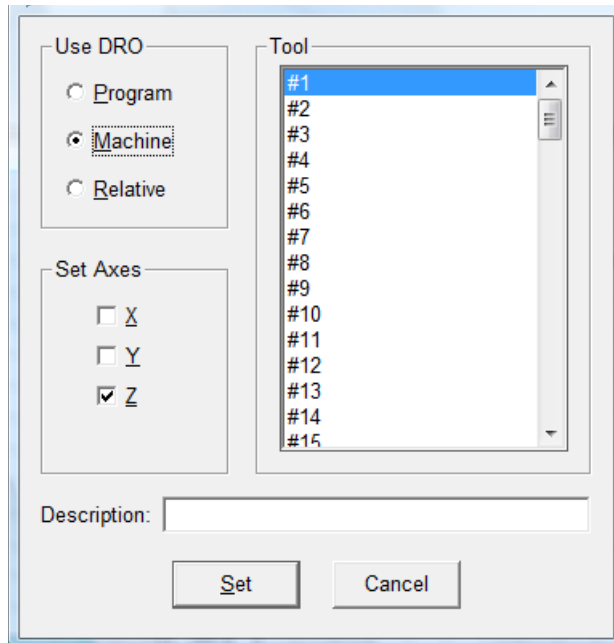


Define Fixture Offset (Machine coordinates only) – Opens the following dialog, which lets you copy the current coordinate values to any fixture offset.



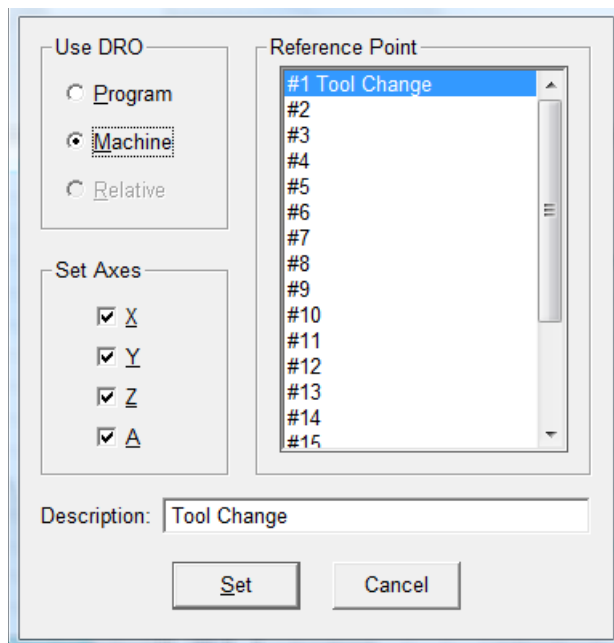
Choose the Set button to copy the current DRO values to the selected fixture offset. Techno copies the values for all axes checked, along with the Description you enter. Note that fixture offsets are listed on the Fixture Offset panel of the Configuration dialog box.

Define Tool Offset – Opens the following dialog, which lets you copy the current coordinate values to any tool offset.



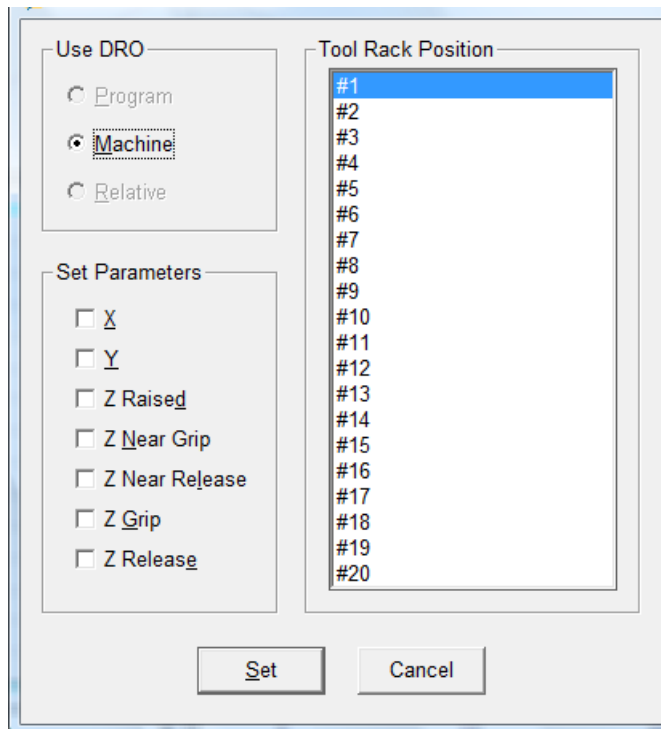
Choose the Set button to copy the current DRO values to the selected tool offset. Techno copies the values for all axes checked, along with the Description you enter. Note that tool offsets are listed on the Tool Library panel of the Configuration dialog box.

Define Reference Point (Machine and Program coordinates only) – Opens the following dialog, which lets you copy the current coordinate values to any reference point.



Choose the Set button to copy the current DRO values to the selected reference point. Techno copies the values for all axes checked, along with the Description you enter. Note that reference points are listed on the Reference Points panel of the Configuration dialog box.

Define Tool Rack Position (Machine coordinates only) – Opens the following dialog, which lets you copy the current coordinate values to any tool rack position.



Choose the Set button to copy the current DRO values to the selected tool rack position. Techno copies the values for all parameters checked. Note that tool rack positions are listed on the Tool Change panel of the Configuration dialog box.

Copy Coordinates – Copies the current coordinates in the DRO to the Windows clipboard. There are several configurable options for this feature. For details, see “General Preferences” in the Initial Setup section.

Viewport Box

The Viewport Box can display up to ten different views of the toolpath described by the current G-Code file. You can select the views to display on the Viewports panel of the Configuration dialog.

Techno provides standard and advanced viewports so you can match the viewport technology to the capabilities of your PC. For more information on choosing your viewport type, see “Viewports” in the Initial Setup section.

The advanced viewports include the following features:

Selection – Click anywhere inside a viewport to select it. When selected, a thin black border appears around the outside of the viewport.

Scrolling – Right mouse click anywhere in the viewport to change the cursor to a hand, then drag in any direction.

Zooming – Use the View menu zooming commands described earlier in this section (zooms the currently selected viewport). You may also use the mouse wheel to zoom. Spinning the wheel away from you will zoom in and spinning the wheel toward you will zoom out.

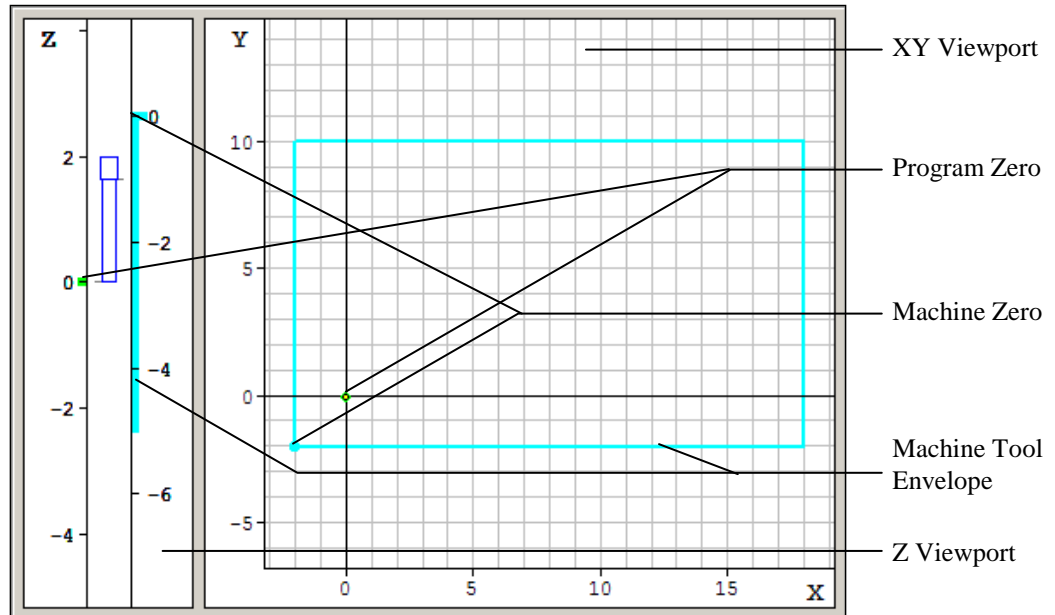
Rotating – To rotate the view in the XYZ and XYZA viewports, hold down the Control key (Ctrl), right mouse click anywhere in the viewport to change the cursor to a hand, then drag in any direction.

Dynamic On/Off – Use the Viewports submenu located on the View menu to turn viewports on and off.

Resizing – Drag the splitter bars between viewports to resize the viewports.

Direction Arrows – Techno displays arrows on every move to show the direction of travel.

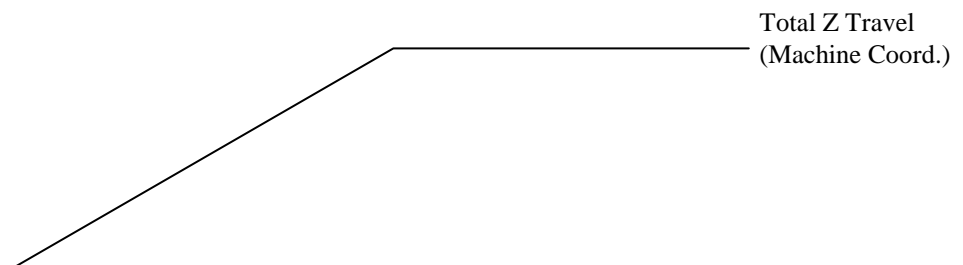
The figure below shows the Viewport Box in a typical configuration for milling (XY and Z viewports displayed). The XY viewport shows an aerial view of the tool envelope. The Z viewport shows the height of the tool during machining. Green and light blue dots represent the origins of the Program and Machine (if used) coordinate systems respectively. The machine tool envelope is shown as the light blue box on the XY viewport and by the light blue bar on the Z viewport.

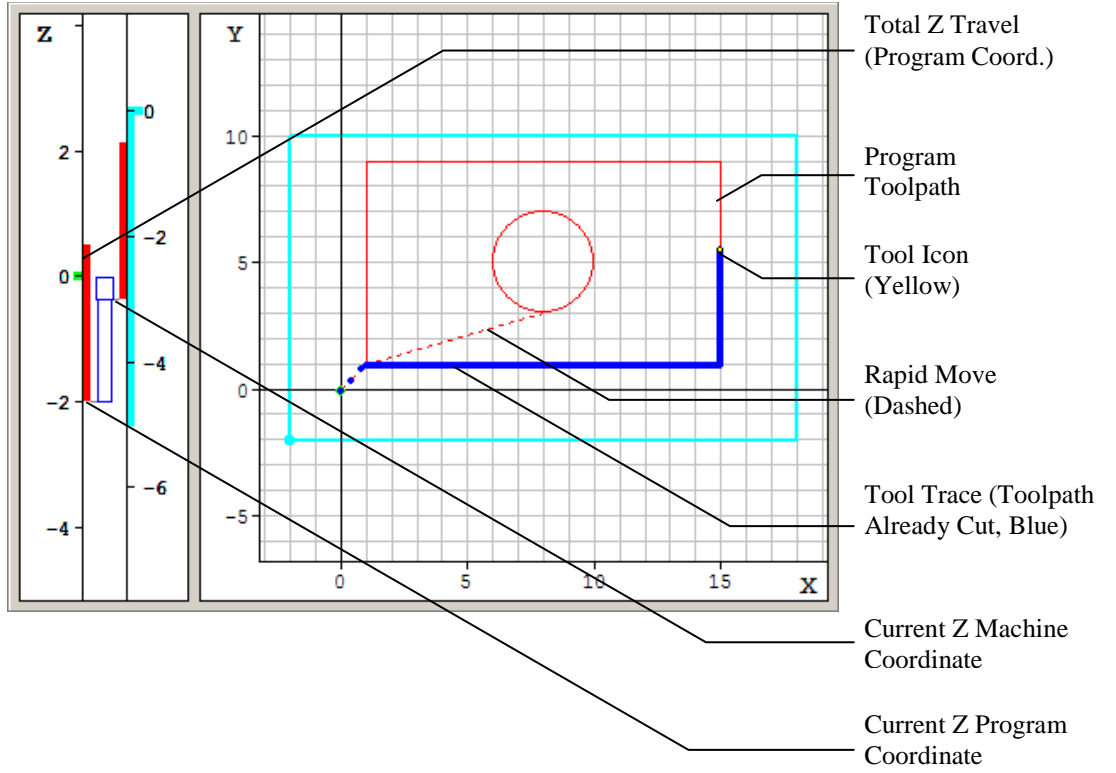


The figure below shows the Viewport Box during the execution of a G-Code file.

In the XY viewport (and all other multi-axis viewports), the yellow dot represents the current position of the tool. The red outline represents the toolpath described by the current G-code program. The blue trace represents the portion of the toolpath that has already been executed. Solid lines depict feedrate moves while dashed lines represent rapid moves.

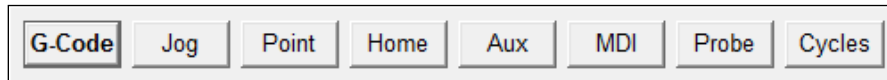
In the Z viewport, there are two scales when machine coordinates are in use: the scale on the left is in program coordinates, while the scale on the right is in machine coordinates. At the bottom of the tool icon, a short indicator line points to the current Z program coordinate. Near the top of the tool icon, an indicator points to the current Z machine coordinate. The left-side red bar represents the total Z travel relative to the program coordinate system. The right-side red bar represents the total Z travel relative to the machine coordinate system.





Control Selection Box

The Control Selection Box has buttons to select each of the seven Control Panels.



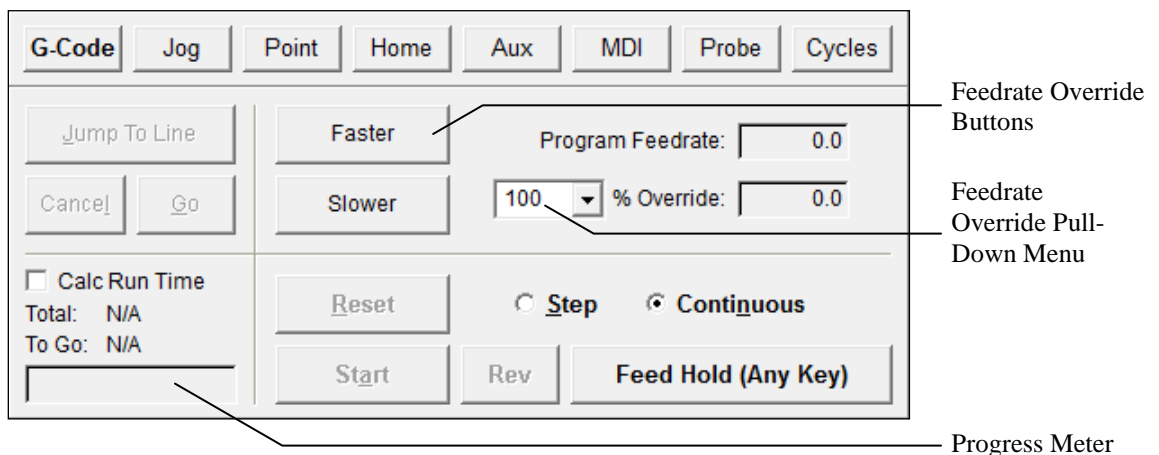
All of the controls you'll need to run your machine tool are located on these panels as follows:

- G-Code Controls to move the tool along the toolpath specified by a G-Code program.
- Jog Controls to manually move the tool one axis at a time.
- Point Controls to move the tool to any point specified.
- Home Controls to seek the home switches.
- Aux Controls for turning auxiliary devices on or off.
- MDI Controls for executing one or more lines of G-Code.
- Probe Controls for 3D digitizing.
- Cycles Controls for generating toolpaths for simple operations, such as drilling holes.

To switch between panels simply click on the appropriate button. Each control panel is described below.

G-Code Control Panel

The G-Code Control Panel provides controls to move the tool as directed by the current G-Code program.



Continuous / Step Radio Buttons – Sets the G-Code file processing mode. In continuous mode, the G-Code program runs without interruption. In Step mode, Techno executes one line of G-Code at a time.

Start Button – Begins execution of the current line of the G-Code file. When in Step mode, execution will stop automatically at the end of the current line, or when the Feed Hold button is hit. When in Continuous mode, execution continues until the end of the program, or until the Feed Hold button is hit. If the program has been stopped in the middle of a G-Code line, choosing the Start button will begin execution exactly where the program stopped.

Rev Button – Runs the G-Code file backwards, which is useful when a tool breaks or when cutting was incomplete.

Feed Hold Button – Pauses execution of the G-Code file. The machine tool will stop, ramping down if necessary. The slower the ramping rate, the longer it will take from the time the Feed Hold button is hit to the time the tool comes to a complete stop.

You can also pause the machine tool by hitting any key on the keyboard (except Shift or Ctrl).

This button pauses any motion including automatic tool changing, tool length sensing and so on.

Reset Button – Resets the current G-Code file to the first executable line and refreshes the Viewport Box.

Feedrate Override Buttons and Pull-Down Menu – Increases or decreases the feedrate on the machine as a percent of the programmed feedrate. You can also enter an exact feedrate override percentage by choosing a value from the Feedrate Override pull-down menu (or typing in any value). Both the programmed feedrate and the override feedrate are displayed. Rapid moves are not affected by feedrate override.

You can use the keyboard instead of the Faster/Slower buttons as follows:

Faster Ctrl + Up Arrow Key

Slower Ctrl + Down Arrow Key

The feedrate override buttons are available while the machine tool is moving, allowing feedrate adjustment on the fly. The buttons are responsive during feedrate moves, but not during rapid moves or dwells. (For dynamic feedrate adjustment during arc moves, the M202B chip is required in the model 401A Signal Generator only.)

Jump to Line Button – Lets you jump to any line in the G-Code file. When you click the button, Techno activates the Program Listing so you can select a G-Code line. The line you select must contain a command (it cannot be a comment or blank line). After you select an executable G-Code line, click the Go button to jump there, or the Cancel button to cancel the operation. If you click Go, Techno repositions the G-Code file, then displays a dialog giving you the option to move the machine tool to the correct location for restarting the file at the new G-Code

line. (If you're using automatic tool changing, Techno will first change to the correct tool.)

If you're using the advanced viewports, in addition to selecting a jump-to G-Code line, you may select a linear or arc move in any 2D or 3D viewport. When you select the move, Techno automatically highlights the matching G-Code line in the Program Listing. This method of finding a jump-to point in your program is usually easier than finding the correct G-Code line.

Note that Jump to Line is not available if the G-Code file contains a BeginGrid command (described later in the System Programming section).

Calc Run Time – Tells the system to calculate an estimate for the total run time for the G-Code file loaded, and to activate the Progress Meter. G-Code files will take slightly longer to load when this option is selected.

Total – Shows the total estimated time to run the G-Code file. This estimate does not take into account feedrate override adjustments after the file has started.

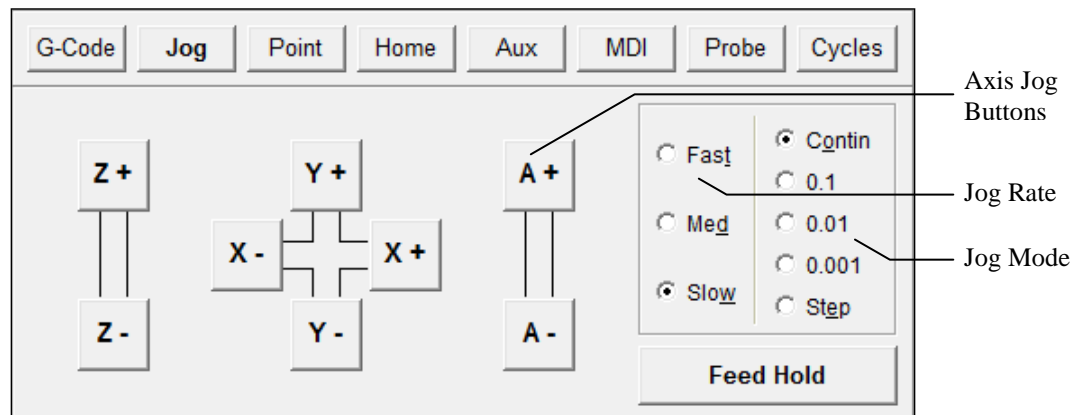
To Go – Shows the estimated time remaining to finish running the G-Code file.

Progress Meter – Shows the current progress running the file, based on the Run Time and To Go time.

Runtime Jog Buttons (X, Y, Z, A, B, C) – You may configure Techno to display runtime jog buttons for up to two axes on the G-Code Control Panel. The runtime jog buttons allow you to jog an axis while a G-Code file is running. The axis moves at the Slow Jog Rate. Note that the runtime jog axes may not be specified in any G-Code file you run. You'll activate the runtime jog buttons on the Jogging panel of the Configuration dialog box.

Jog Control Panel

The Jog Control Panel provides controls for manually positioning all axes.



Axis Jog Buttons

You can move a single axis of your machine tool by pressing and holding an Axis Jog Button. Ramping is used if the jog rate is faster than the Start/Stop Feedrate

for a given axis. Note that you can also jog the machine using the keyboard. The controls are mapped as follows:

X+	Ctrl + Right Arrow Key
X-	Ctrl + Left Arrow Key
Y+	Ctrl + Up Arrow Key
Y-	Ctrl + Down Arrow Key
Z+	Ctrl + Page Up Key
Z-	Ctrl + Page Down Key
A+	Ctrl + Plus (+) Key
A-	Ctrl + Minus (-) Key
B+	Ctrl + Close Bracket (]) Key
B-	Ctrl + Open Bracket ([) Key
C+	Ctrl + Period (.) Key
C-	Ctrl + Comma (,)Key

Jog Rate

The Slow, Medium and Fast buttons set the jog rate to the corresponding rate specified on the Jogging panel of the Configuration dialog box. Note that there are separate jog rates defined for each axis, and you may specify the default jog rate to use at program startup.

Jog Mode

- Continuous – Sets Techno to move continuously while you hold down an Axis Jog button.
- Discrete Distances – Sets Techno to move the indicated distance each time you click an Axis Jog button. You may customize these distances on the Jogging panel of the Configuration dialog box.
- Step – Sets Techno to move exactly one motor step each time you click an Axis Jog button.

Feed Hold Button

Stops motion during a discrete distance move, or any other operation including automatic tool changing, tool length sensing and so on.

Point Control Panel

The Point Control Panel provides controls for moving the tool to any position at any feedrate.

Techno CNC does not always move all axes simultaneously. The sequence of axis motions follows a general-purpose scheme based on three fields in the Advanced Definition and Basic Homing Panels of the Configuration dialog box: Home End, Home Order, and Point Move Linear Interpolate. These fields should be configured to allow for safe moves in Point mode, which generally means the

tool retracts from the workpiece before any other motion occurs. Techno uses the following rules to sequence the individual moves:

1. Do all non-interpolated moves that are towards the Home End, in the Home Order.
2. Do the interpolated move, if any.
3. Do all non-interpolated moves that are away from the Home End, in reverse Home Order.

By setting the three configuration fields properly, you can make the sequence safe for your machine tool configuration.

Example:

For a milling application that includes a rotary table, the configuration settings described above are typically set as follows:

Axis	Home End	Home Order	Pt. Move Linear Interp.
X	Negative	2	Yes
Y	Negative	3	Yes
Z	Positive	1	No
A	Negative	4	No

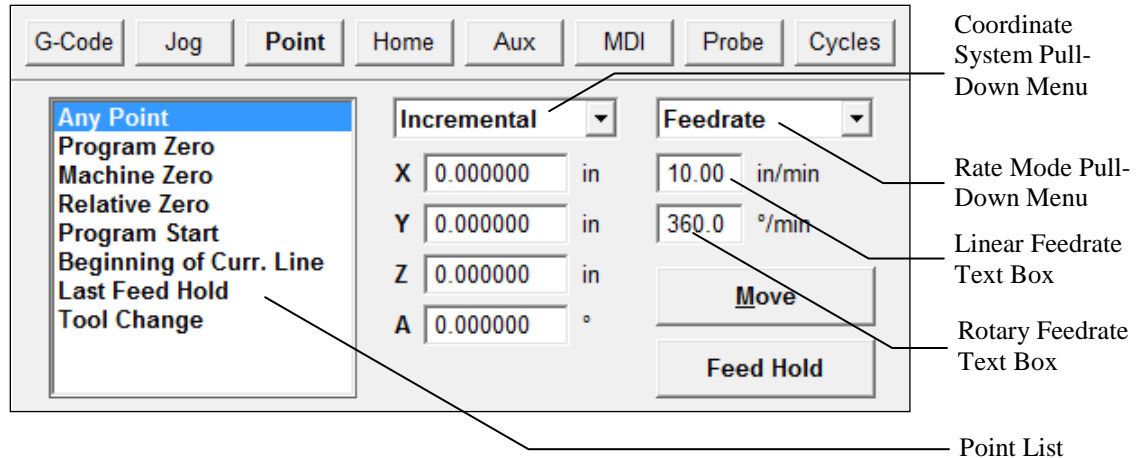
If the Z axis must be raised to reach the destination point, and the A axis must move in the positive direction, the individual moves will be:

1. Z axis moves up
2. X and Y axes move together
3. A axis moves

If the Z axis must be lowered to reach the destination point, and the A axis must move in the positive direction, the individual moves will be:

1. X and Y axes move together
2. A axis moves
3. Z axis moves down

The G28, G29 and G30 commands (described later in the System Programming section) use the same scheme.



Point List

Displays the list of points defined on the Point List panel of the Configuration dialog box. The default points are:

- Any Point – Moves to any XYZA point in the selected coordinate system.
- Program Zero – Moves to Program Zero.
- Machine Zero – Moves to Machine Zero (if defined).
- Relative Zero – Moves to Relative Zero.
- Program Start – Moves to where the tool was located when the current G-Code program was started.
- Beginning of Current Line – Moves to where the tool was located when the current G-Code line began execution. Also, after jumping to a new G-Code line, this value is the starting point for the G-Code line to which you jumped.
- Last Feed Hold – Moves to where the tool was located when G-Code execution was stopped by a feed hold.
- Tool Change – Moves to the G28 reference point specified on the Reference Point panel of the Configuration dialog box.

Note that you can select each item in the Point List using the keyboard as follows:

Any Point	Shift + 1
Program Zero	Shift + 2
Machine Zero	Shift + 3
Relative Zero	Shift + 4
Program Start Point	Shift + 5
Beginning of Current Line	Shift + 6
Last Feed Hold Point	Shift + 7
Tool Change Position	Shift + 8

If you customize the Point List, the 'Shift + n' keyboard commands select the first 8 points listed, in numerical order (e.g. 'Shift + 1' for the first point, 'Shift + 2' for the second point, and so on.)

Coordinate System Pull-Down Menu – The tool will move to the XYZA position in program coordinates, machine coordinates, relative coordinates, or incrementally from the current position of the tool, depending on the option you select.

Rate Mode Pull-Down Menu – You can set the travel rate by selecting one of the following:

- **Rapid** – The machine tool moves at the maximum feedrate allowed by your current maximum feedrate settings on the Feedrate/Ramping panel of the Configuration dialog box. For rapid rotary axis moves to a Program, Machine or Relative coordinate, Techno will make the shortest possible move in the direction indicated (always less than 360 degrees.)
- **Feedrate** – The machine tool moves at the feedrate you enter in one of the feedrate text boxes.

Linear Feedrate Text Box – If Feedrate is selected in the Rate Mode pull-down menu, any moves that involve linear axes only will use this feedrate. When you startup Techno, this value defaults to the Point Feedrate-Linear field on the Feedrate/Ramping panel of the Configuration dialog box.

Rotary Feedrate Text Box – If Feedrate is selected in the Rate Mode pull-down menu, any moves that involve the 'A' axis will use this feedrate. When you startup Techno, this value defaults to the Point Feedrate-Rotary field on the Feedrate/Ramping panel of the Configuration dialog box.

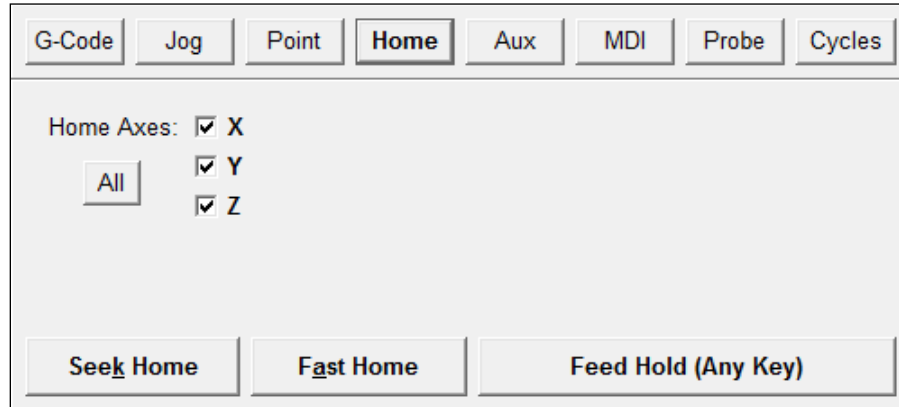
Move Button – Executes the move.

Feed Hold Button – Stops execution of the move. Hitting any key on the keyboard (except Shift or Ctrl) also stops the move.

This button pauses any motion including automatic tool changing, tool length sensing and so on.

Home Control Panel

The Home Control Panel provides controls to seek the home switches and establish the Machine Zero location.



X, Y, Z, A, B, C – Indicate which axes to home. If Machine Zero has not been set, you must home all axes.

All – Checks all axis checkboxes.

Seek Home – Finds the home switch on up to four axes. To save time, it is recommended that you first jog each axis near the home switch before homing. Once Machine Zero (home) is set, the machine tool envelope is redefined.

If Machine Zero was already set before homing, Techno displays a dialog showing the discrepancy between the previous Machine Zero and the new Machine Zero just found. This provides a convenient way to check that no steps were lost while cutting a part, within the accuracy limits of the home switch. The home switches supplied as a Techno accessory have a repeatability of ± 0.001 ".

When multiple motors are used to drive an axis, Techno simultaneously homes each motor independently. For example, if two motors are driving a single axis, Techno starts the homing process by moving both motors towards the homing switches. Once one motor reaches its switch, it stops and waits until the other motor reaches its switch. Once that occurs, all motors retract simultaneously from the switches. This process will correct any skewing that may have occurred between the two actuators.

The homing operation can be tailored several ways, to suit the particular machine tool configuration and your preferences (see "Homing Settings" in the Initial Setup section of this manual.)

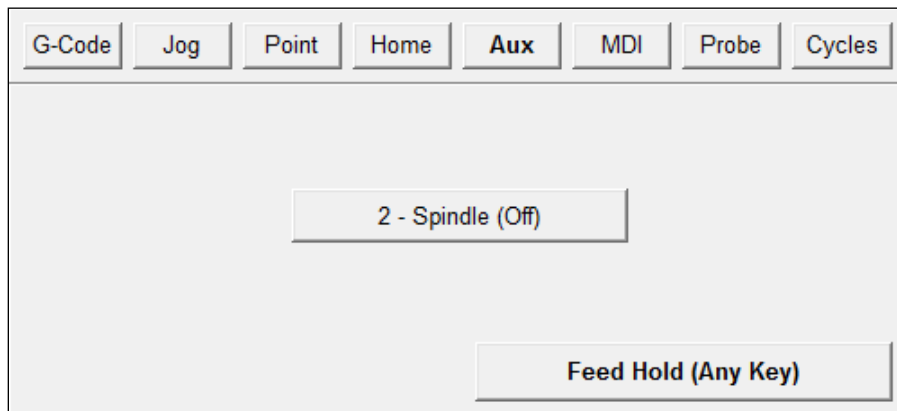
Fast Home – Executes a rapid move to a position near Machine Zero, then performs a normal homing operation. This button is enabled whenever Machine Zero has been set (so the first homing operation must be run via the Seek Home button.) The destination for the initial rapid move is set on the Basic Homing panel of the Configuration dialog box.

Feed Hold Button – Stops the homing operation. Hitting any key on the keyboard (except Shift or Ctrl) also stops the operation.

This button pauses any motion including automatic tool changing, tool length sensing and so on.

Auxiliary Control Panel

The Auxiliary Control Panel provides controls for turning on or off auxiliary devices, such as spindles or coolant pumps. By default Techno displays a toggle button for each output line in use. Details on displaying buttons are described in “Output Line Settings” in the Initial Setup section.



On / Off Buttons – Turns the indicated output line on or off. For safety, Techno displays a warning when you turn on a device (click OK to proceed). The warning may be turned off on the Messages panel of the Configuration dialog box.

If turning the output line on or off violates a safety interlock, Techno displays the appropriate message and does not set the output line. Safety interlocks are defined on the Output Lines panel of the Configuration dialog box.

Note that you can operate the On/Off buttons using the keyboard as follows:

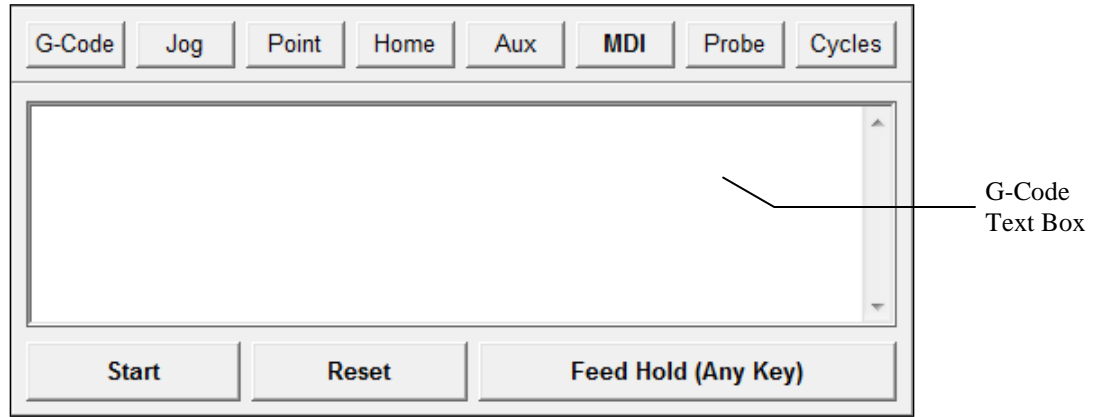
- Lines 1 – 9: Ctrl + output line number toggles the line on and off
- Lines 10 – 16: Ctrl + Shift + second digit of output line number toggles the line on and off (e.g. for line 12 use Ctrl + Shift + 2)

IMPORTANT NOTE: The keystroke combination ‘Alt + output line number’ does not turn the line off, as it did in all earlier Techno versions (1.00 – 2.0.8).

Feed Hold Button – Stops the current operation, including automatic tool changing, tool length sensing and so on. Hitting any key on the keyboard (except Shift or Ctrl) also stops the operation.

MDI Control Panel

The MDI Control Panel provides controls that let you run a few lines of G-Code without creating a G-Code file. The acronym “MDI” is an industry standard abbreviation for Manual Data Input.



G-Code Text Box – Enter one or more lines of G-Code. Techno will interpret these lines just as if they were contained in a short G-Code file.

Start Button – Begins execution of the G-Code.

Reset Button – Resets to the beginning of the G-Code.

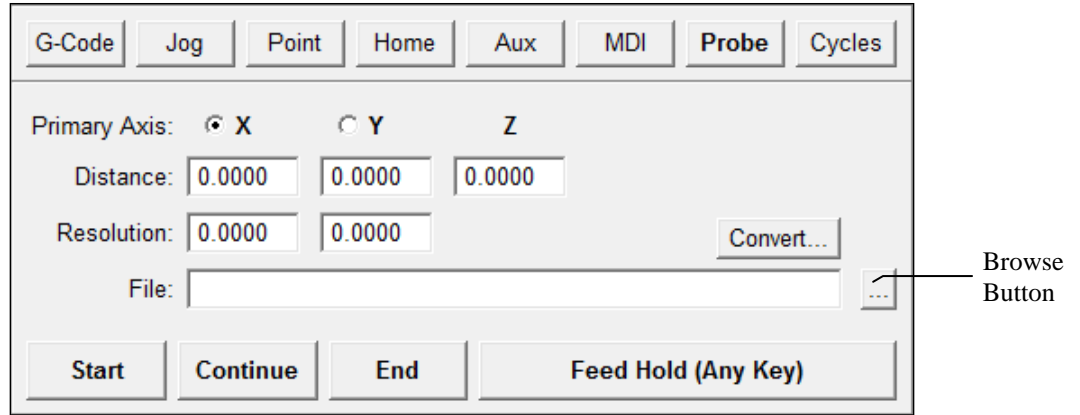
Feed Hold Button – Stops execution of the G-Code.

When using the MDI Control Panel there are a few things keep in mind:

- When you click Start, Techno replaces the currently-loaded G-Code file with the MDI G-Code, and displays the MDI G-Code in the Program Listing Box.
- After running the G-Code, you can open the editor to save it into a file.

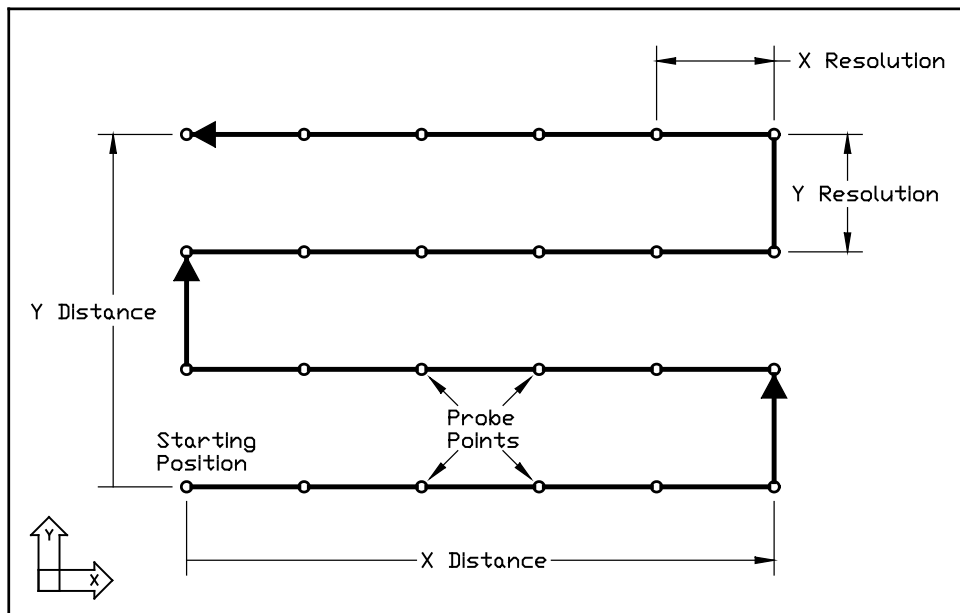
Probe Control Panel

The Probe Control Panel provides controls for digitizing any 3D object. Techno positions a touch probe at a series of points, and at each point moves the probe until it touches the object. When finished, Techno saves the probed positions in industry-standard file formats for import into CAD or CAM software.



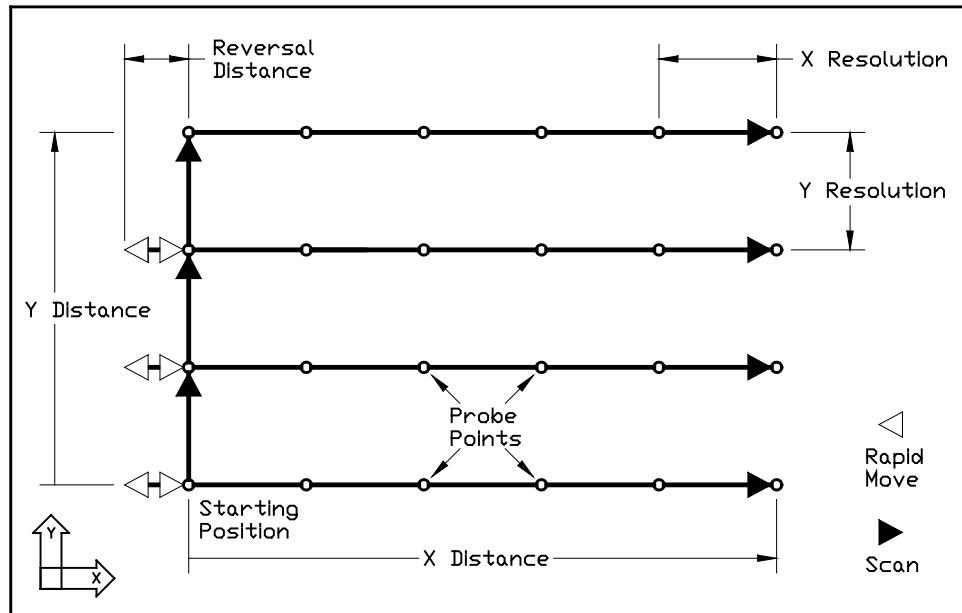
A typical application is to capture the shape of an object on a milling machine. In this application, Techno senses the Z position of the object's surface at each location on an XY grid of points. The examples below illustrate the XY path for each of the two available scan patterns: Zig Zag and Single Direction. In both examples X is the primary axis and Y is the secondary axis.

Zig Zag Scan Pattern



From the Starting Position, Techno first traverses the grid along the primary axis, then indexes one row along the secondary axis. The system repeats this pattern until it reaches the last probe point. The zig zag pattern is useful when backlash is not a concern.

Single Direction Scan Pattern



In this pattern, scanning is always performed in the same direction on each axis. Before scanning each row of probe points, Techno performs a rapid move to the reversal position, then back to the X starting position. This ensures that the drive mechanism is loaded in the same direction for all scanning, so backlash will not affect the results.

Note that the X and Y Distances have a length and a direction (+/-) from the Starting Position. In this example the direction is positive for both X and Y.

While most applications use an XY grid of points and sense the object by moving the Z axis, the grid can be in any plane: XY, XZ or YZ. The sensing axis is always perpendicular to the grid. Furthermore, for 2D applications, you can reduce the grid to a line of points by setting one of the grid distances to zero.

During the scan, Techno stores Program Coordinates as the data points. Therefore, before you start scanning it's important to set the program zero location so the points will be useful in your particular application. Often you can simply set program zero at the scan Starting Position.

After finishing a scan, Techno can save the data in several formats:

- SCN: Techno CNC native format, not used by other software
- STL: Industry standard format representing a 3D surface
- XYZ: Industry standard format representing a cloud of points

For the STL and XYZ formats you can have Techno offset the points to compensate for the radius of the touch probe tip. Select your desired output file formats on the Touch Probe panel of the configuration dialog box.

You'll use the following controls on the Probe Control Panel to set up and run your scan.

Primary – Specifies the primary axis.

Distance – Specifies the length and direction of the scanning envelope from the Starting Position to the diagonal corner, for each axis. The sign (+/-) of the value indicates the direction.

Res – Specifies the resolution of the scan (the distance between data points) for each axis.

File – Specifies the file in which to store the raw scan data (SCN format).

Browse – Lets you specify a file to store the raw scan data.

Start – Starts a new scan.

Continue – Continues a scan in progress.

End – Ends a scan before Techno finishes collecting the entire grid of data points.

Feed Hold – Pauses a scan.

Follow these steps to run a scan.

▶ To prepare for a scan:

1. Review the configuration settings on the Touch Probe panel of the Configuration dialog box.
2. Mount the probe in the spindle and make sure the spindle cannot be turned on accidentally.
3. Make sure the probe is wired to the Input connector on the Signal Generator. See the Signal Generator Hardware Guide for input line wiring information.
4. Test the probe wiring by choosing Input Line Status from the Controller menu, then pushing the probe tip to activate the switch. The Status box for the probe should turn red when you push the probe tip. If the switch is not wired correctly, the touch probe will be damaged when you start digitizing.
5. Secure the object to be scanned.
6. Click Probe on the Control Selection Box.

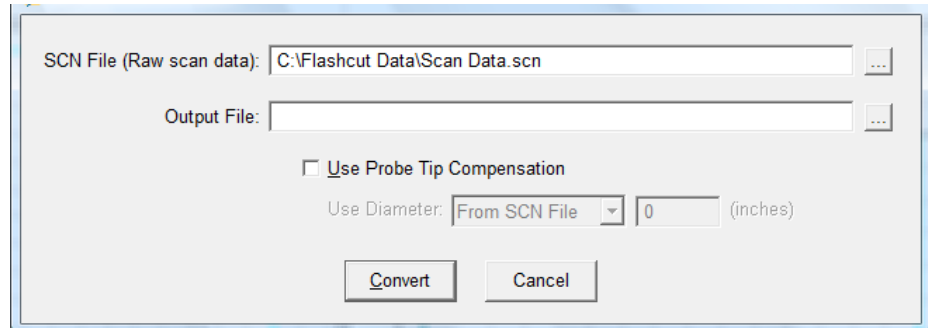
▶ To run a typical 3D scan on a mill:

1. Jog the machine to the Starting Position for the scan.
 - X and Y: The starting XY location defines one corner of the scanning envelope. Generally you should use the corner towards X- and Y-, but you can choose any of the four corners.

-
- Z: The Z position must be higher than the highest point on the object, and should allow extra room for Techno to retract the probe upward from the surface. (The Retract Distance is specified on the Touch Probe configuration panel.)
2. Zero Program Coordinates (or set Program Zero according to your needs).
 3. Select X as the primary axis for the scan (or set according to your needs).
 4. Enter the X and Y Distances to the diagonal corner of the scan. These will be positive numbers if the probe is currently at the X- and Y- corner of the scanning envelope.
 5. Enter the Z Distance (the depth of the scanning envelope). The sign must be negative. The Z Distance must be deep enough to allow scanning the lowest point on the object.
 6. Enter the scanning resolution for X and Y in the Res fields.
 7. Click the “...” button to select a file or define a new file to store the results (or skip this step and use the default file already displayed).
 8. Click Start.
- ▶ To run a 2D scan:
 1. Follow the same steps as for a 3D scan but set one of the Distances to zero.
 2. Click Start.
 - ▶ To scan a single point:
 1. Follow the same steps as for a 3D scan but set both of the Distances to zero.
 2. Click Start.

As described above, Techno can automatically generate output files of various formats at the end of the scan. If at the time of the scan the settings on the Touch Probe configuration panel didn't include all the files you need, you can create additional output files using the Convert button.

- To convert data for a completed scan from the SCN format to another available format:
1. Click the Convert button. The following dialog appears:



2. Enter values for all fields as follows:

SCN File – Enter the SCN file containing the raw scan data.

Output File – Specify the new file to create. Click the “...” browse button to display the Windows file-saving dialog. The file formats available in the dialog are STL and XYZ.

Use Probe Tip Compensation – Check this box if you want Techno to compensate for the radius of the touch probe tip.

Use Diameter – When creating a compensated output file, Techno must know what touch probe diameter value to use. The drop-down menu provides several choices:

- From SCN File: Techno uses the value stored in the SCN file at the time of the scan.
- From Cfg Dialog: Techno uses the current value from the Touch Probe configuration panel.
- Set Here: Techno uses the value you enter.

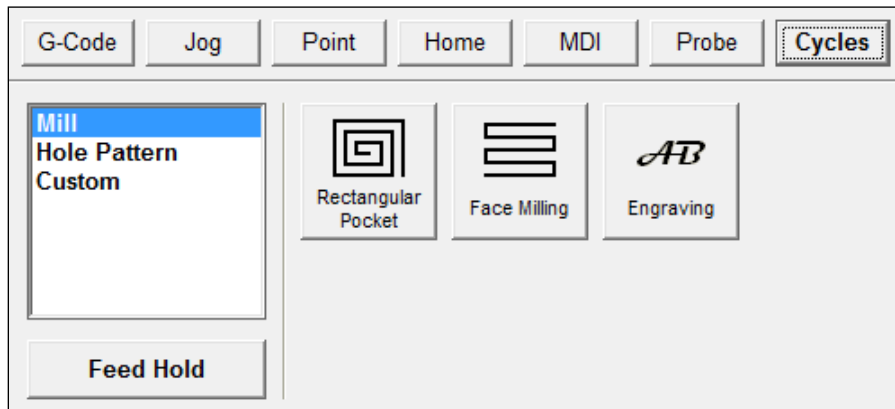
3. Click Convert.

Cycles Control Panel

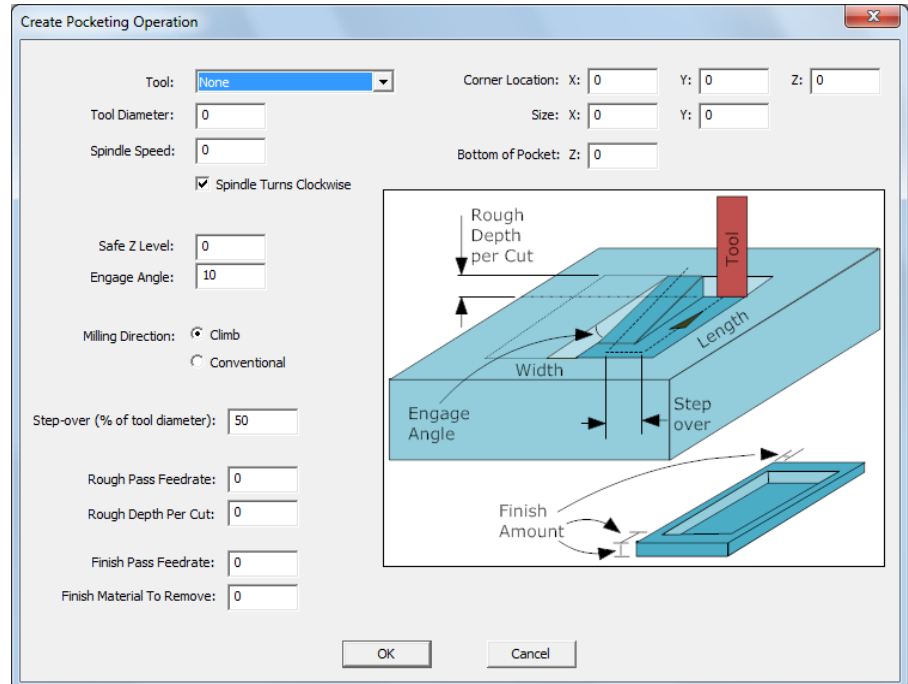
The Cycles Control Panel lets you generate G-Code for standard operations such as hole drilling, pocketing, and so on.

Mill

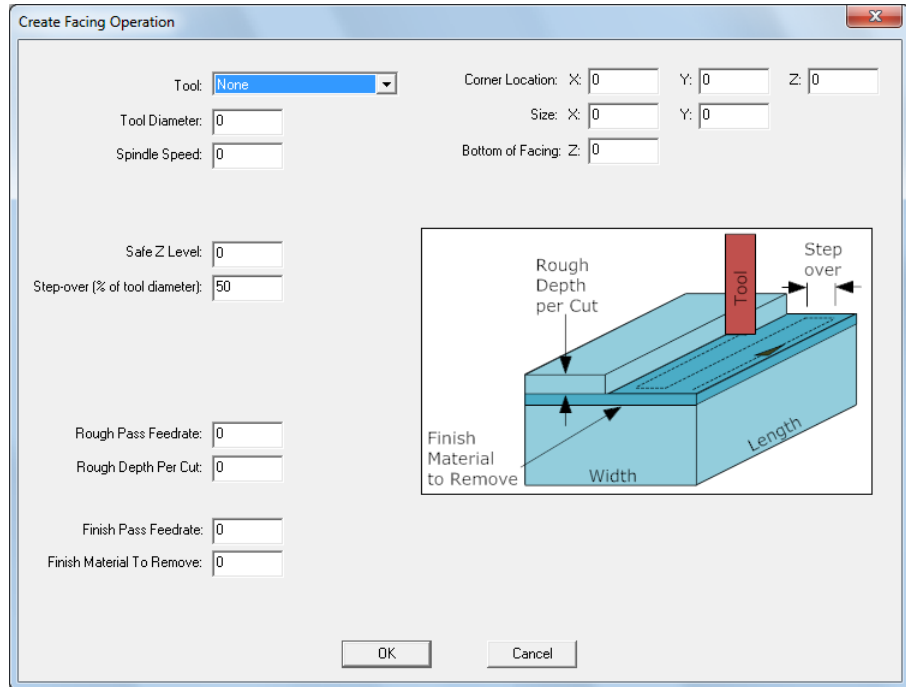
Generates G-Code for several milling processes including rectangular pocketing, facing and engraving.



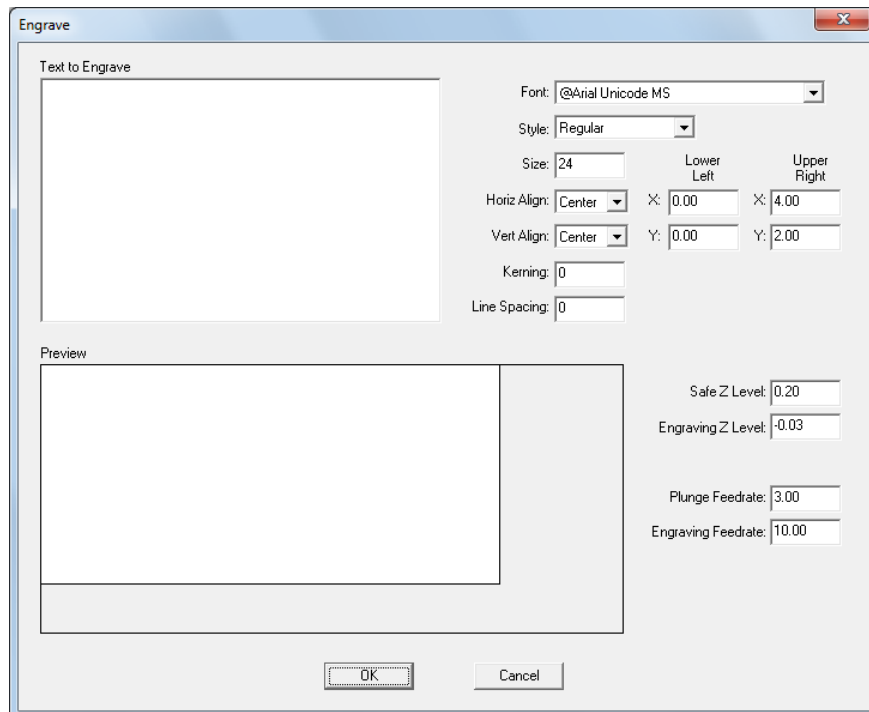
- **Rectangular Pocket** – Cuts a rectangular pocket starting from a specified Corner Location (lower left corner in XY view). The Tool and Spindle Speed fields are optional.



- **Face Milling** – Faces the material to the desired Z height, starting from a specified Corner Location (lower left corner in XY view). The Tool and Spindle Speed fields are optional.

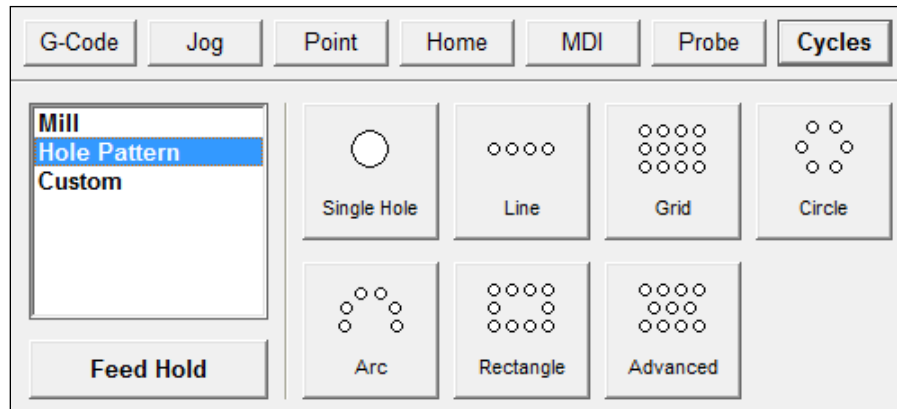


- **Engraving** – Engraves text using any font installed on your computer. Techno generates G-Code that traces the outside of each letter.



Hole Pattern

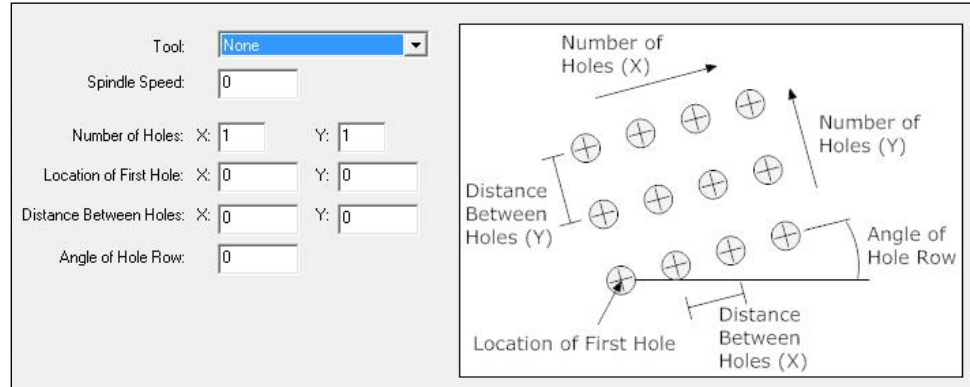
Generates G-Code for drilling holes in a variety of patterns.



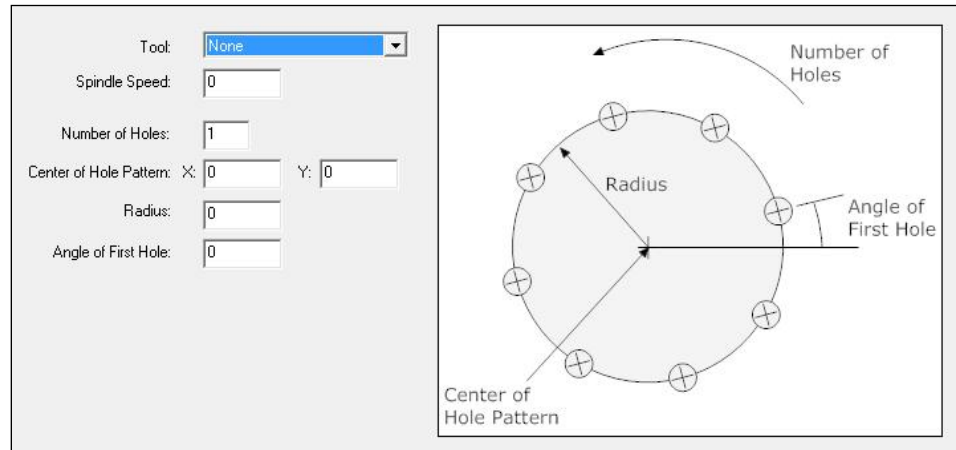
- **Single Hole** – Drills a single hole at the point you specify. The Tool and Spindle Speed fields are optional. You must select the type of drilling cycle and set the drilling cycle properties.

- **Line** – Drills a uniformly spaced line of holes starting at the specified Location of First Hole. The Tool and Spindle Speed fields are optional. You must select the type of drilling cycle and set the drilling cycle properties.

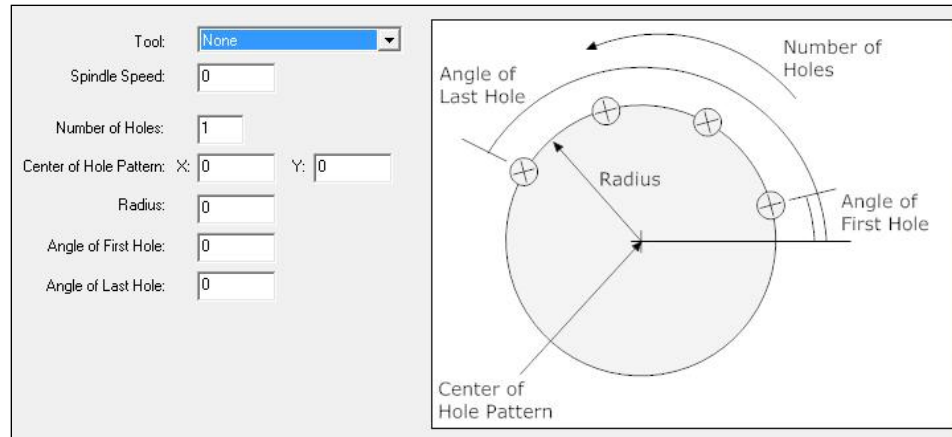
- **Grid** – Drills a uniformly spaced grid of holes starting at the specified Location of First Hole (lower left corner of grid in XY view). The Tool and Spindle Speed fields are optional. You must select the type of drilling cycle and set the drilling cycle properties.



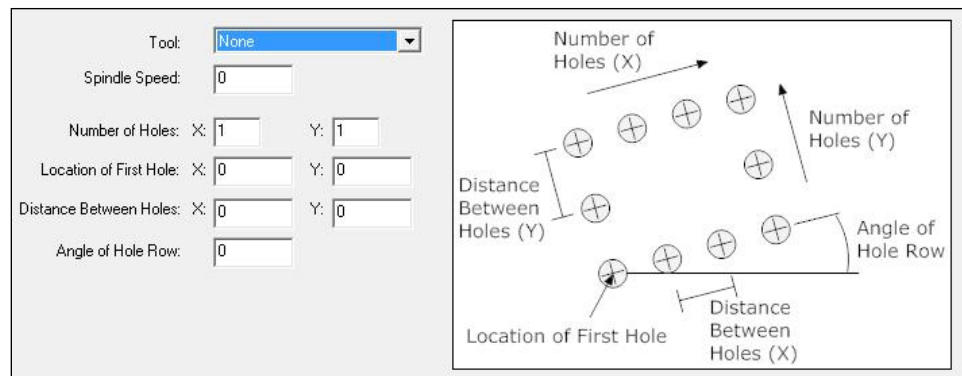
- **Circle** – Drills a uniformly spaced circle of holes. The Tool and Spindle Speed fields are optional. You must select the type of drilling cycle and set the drilling cycle properties.



- **Arc** – Drills a uniformly spaced arc of holes. The Tool and Spindle Speed fields are optional. You must select the type of drilling cycle and set the drilling cycle properties.



- **Rectangle** – Drills a uniformly spaced rectangle of holes starting at the specified Location of First Hole (lower left corner of rectangle in XY view). The Tool and Spindle Speed fields are optional. You must select the type of drilling cycle and set the drilling cycle properties.



- Advanced – Drills a custom pattern of holes. You'll build the pattern row by row using the Add button. The Tool and Spindle Speed fields are optional. You must select the type of drilling cycle and set the drilling cycle properties.

Tool:

Spindle Speed:

Rows:

Offset to First Row (inches): Y:

Pattern Length (inches): Y:

Custom

Custom cycles are third-party plug-in programs that generate G-Code for specific applications. Please contact technical support for information on creating custom cycles.

Feed Hold Button - Pauses any motion including automatic tool changing, tool length sensing and so on.

Tool Box

The Tool Box contains controls for selecting and displaying information about the current tool.

Tool:

Offset: X Y Z

Comp:

Cutting Time:

Current Tool Pull-Down Menu

Current Tool Pull-Down Menu – Lets you select the current tool loaded in the machine tool. When you select a tool, Techno applies tool length/geometry compensation according to the Compensation Mode Pull-Down Menu (described below). See the System Programming section for more information on using tool offsets.

If you are using an automatic tool changer, selecting a new tool will cause an automatic tool change. To stop execution of the automatic tool change, choose the Feed Hold button on the currently-displayed control panel.

If you're using an automatic tool changer, you can configure the system to display only tools that are currently loaded in the changer. See "Tools in Changer Settings" in the Initial Setup section of this manual for more details.

Current Tool Diameter – Displays the diameter for the current tool (from the Tool Library panel of the Configuration dialog box.)

Current Offset Applied – Displays the current XYZ offset applied. Note that in some cases this offset will *not* be the offset defined for the current tool. For example, a G-Code program may call for a tool change without specifying a subsequent tool offset command. In that case, the offset will still reflect the last offset command (or zero if no offset commands have been executed).

Compensation Mode Pull-Down Menu – Lets you select the current tool compensation mode. Techno uses this setting for tool length/geometry compensation (see the System Programming section for more information).

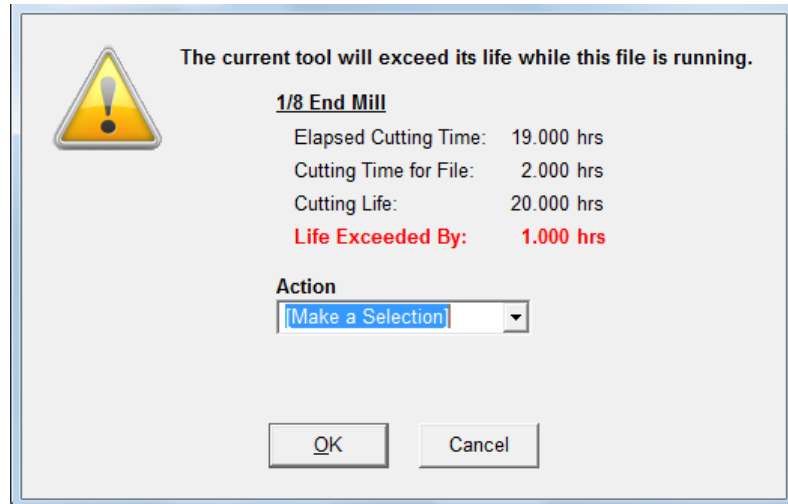
Length – Executes automatic tool length sensing (when activated), for milling-type machines only. The sequence of operations is described in "Tool Length Sensing" in the Initial Setup section.

To stop execution of tool length sensing, choose the Feed Hold button on the currently-displayed control panel.

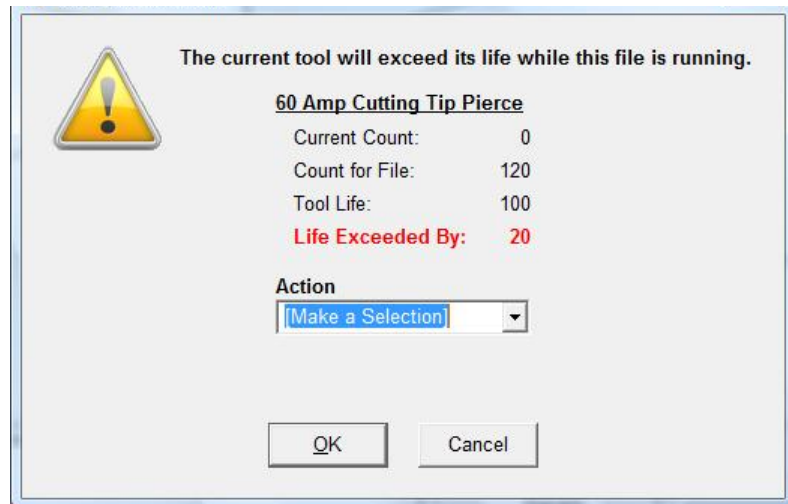
Cutting Time / Count – Displays either the cutting time or the count of the currently loaded tool as controlled through the tool life management system. See "Tool Life Management" in the Initial Setup section of this manual for information on configuring Tool Life Management. Note that the system only updates a tool's usage when the Signal Generator is connected.

When you start running a G-Code file and the tool life will be exceeded by the loaded program (or the life of the tool has already been exceeded), Techno will display one of the following alerts:

- Time-based Tool Life Management



- Counter-based Tool Life Management



For either dialog you'll need to select one of the following Actions:

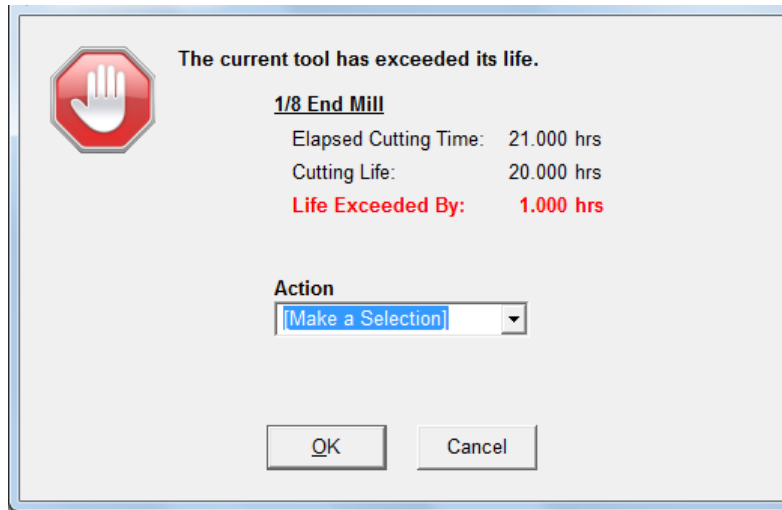
- Replace Tool

You'll change the tool and enter the starting usage for the tool loaded (0 if the tool is brand new).

- Continue Using Tool

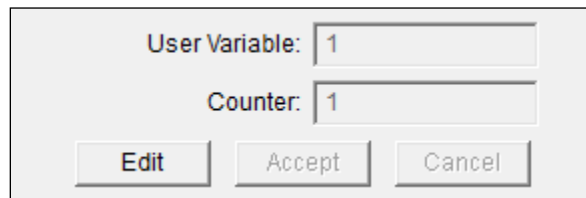
Techno lets you use the current tool for the G-Code file.

Replace – Click this button to inform Techno that you have replaced the tool. Enter the starting usage for the new tool (0 if the tool is brand new).



User Variables Box

The User Variables Box displays the current value of any User Variables or Counters configured for display on the main screen (see “User Variable Settings” and “Counter Settings” in the Initial Setup section for details).



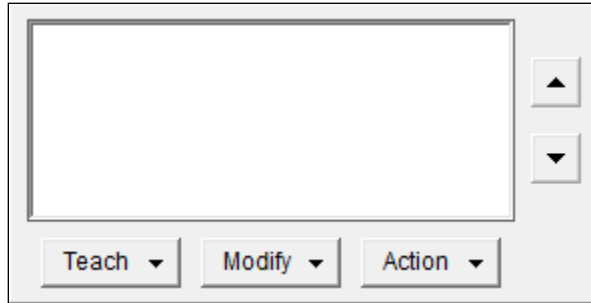
Edit – Allows you to change the value of any User Variable or Counter that is configured to be editable. If you change the value of a User Variable while a G-Code file is paused, Techno will reset the file.

Accept – Saves any changes made to the User Variables and Counters and locks the fields.

Cancel – Discards any changes made to the User Variables and Counters and locks the fields.

Teach Box

The Teach Box provides controls for teaching a series of linear move commands and generating a G-Code file from the commands.



Teach

- G00 Rapid Move – Adds a command for a rapid move to the current program coordinates. The number of decimal places is specified on the Preferences:General panel of the Configuration dialog box.
- G01 Feedrate Move – Adds a command for a feedrate move to the current program coordinates.

Modify

- Edit – Opens a dialog that allows you to edit the command.
- Delete – Deletes the command.
- Clear – Deletes all commands.

Action

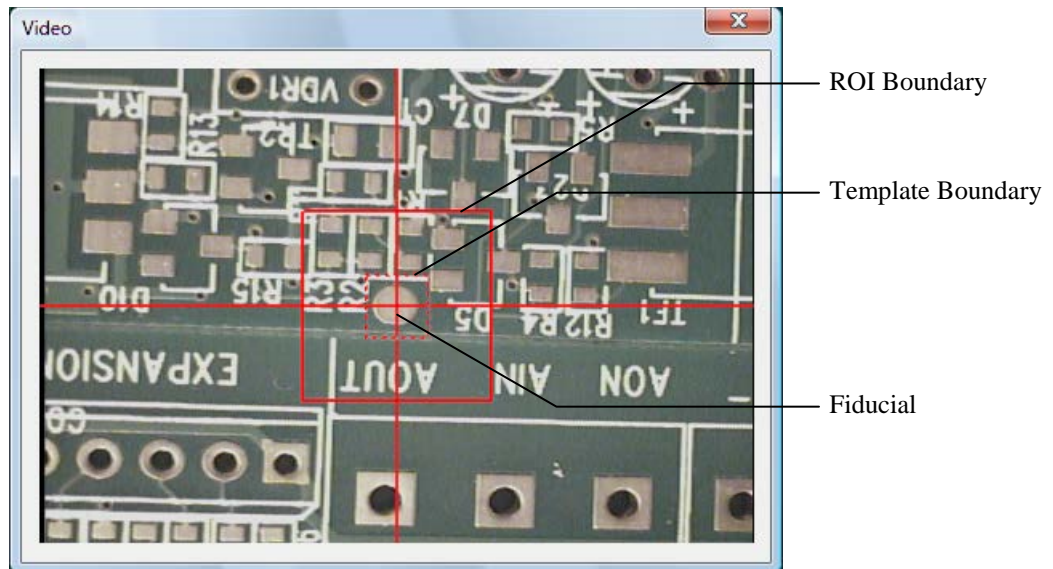
- Create G-Code – Generates a G-Code file from the command list, and loads the file into Techno. The feedrate is specified on the Preferences:General panel of the Configuration dialog box.
- Edit G-Code – Generates G-Code from the command list, and loads it into the editor (unsaved).
- Move – Moves the machine to the program coordinate for the command.

Up/Down Arrows – Moves the selected command up/down in the list.

Fiducial Box

The Fiducial Box provides controls for using Techno's Fiducial Correction System. The Fiducial Correction System uses a camera to locate fiducials (reference marks) on a workpiece and adjust the toolpath in the XY plane to align the toolpath with the fiducials. A fiducial can be any mark on the workpiece, but for accurate rotational correction a filled (solid) circle works the best.

When fiducial correction is in use a separate video window appears on your screen.



The template is the area stored as a fiducial image. It should be just large enough to completely contain the fiducial. The region of interest (ROI) is the area in which Techno will search for the fiducial image while running a G-Code file. The ROI must be large enough to allow for variation between workpieces, but not so large as to unnecessarily slow down image processing or allow the wrong fiducial to be found.

To use the Fiducial Correction System you must generate a fiducial file that contains a list of the fiducials for a given G-Code file. The fiducial file must have the same name as the G-Code file, but use the “.fid” extension. It must be located in the same folder as the G-Code file. The fiducial file consists of one line per fiducial formatted as follows:

$(x,y),r,t,s$

where the values are

x: X coordinate of the fiducial (program coordinates)

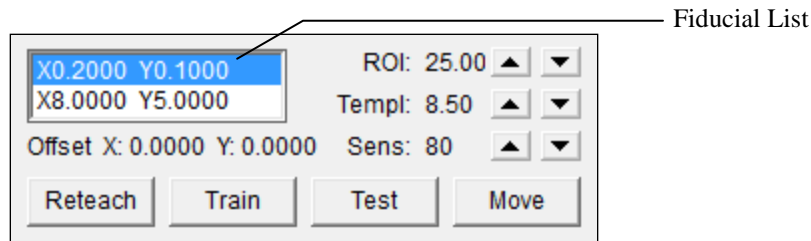
y: Y coordinate of the fiducial (program coordinates)

r: Size of the region of interest

t: Size of the template image

s: Sensitivity (1-100) that will be used for finding a fiducial

When fiducial correction is in use the Fiducial Box appears on the right side of the main screen.



The controls in the Fiducial Box let you manage the fiducials as follows:

Fiducial List – Shows the list of fiducials from the currently-loaded fiducial file.

ROI Size – Length of each side of the region of interest for the selected fiducial. You may change the size using the up/down arrows. Changes are stored immediately to the fiducial file.

Template Size – Length of each side of the template for the selected fiducial. You may change the size using the up/down arrows. Changes are stored immediately to the fiducial file.

Sensitivity – The threshold at which the selected fiducial will be found (1-100 scale). You may change the sensitivity using the up/down arrows. Changes are stored immediately to the fiducial file.

Offset – The offset applied to the toolpath to align it with the fiducials. Techno resets it to zero whenever you load a new G-Code file.

Reteach – Resets the XY coordinate of a fiducial to the current program coordinates. The new XY location is saved immediately to the fiducial file.

Train – Captures the image contained within the template and associates it with the selected fiducial.

Test – Attempts to find the image stored for the selected fiducial within the template boundary in the video window.

Move – Moves the machine tool to the XY location of the selected fiducial.

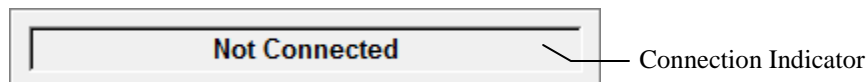
- ▶ To train the Fiducial Correction System for a new G-Code file
 1. Attach a USB digital camera to your PC. The camera must support the Windows DirectShow interface.
 2. Configure Techno for your camera using the Camera panel of the configuration dialog. See “Camera Settings” in the Initial Setup Section for more information.
 3. Select the Fiducial Box checkbox on the Main Screen panel of the Configuration dialog.

4. Generate a fiducial file for the G-Code file you will run.
5. Load the G-Code file. Techno automatically loads the associated fiducial file and displays the list of fiducials. The G-Code file should include the G170 command near the top (described below).
6. Use the Move To button to move the machine to the selected fiducial.
7. Use the Jog controls to align the camera window crosshairs with the center of the fiducial.
8. Use the Reteach button if you need to adjust the XY location of the selected fiducial to the current program coordinates
9. Click Train to capture and store an image for the fiducial.
10. Click Test to test finding the fiducial. Adjust the sensitivity as necessary.
11. Repeat steps 6-10 for each fiducial.

Once the fiducials have been trained, whenever you run the G-Code file, the G170 command will cause the system to move to each fiducial location and attempt to find the fiducial within its ROI. If Techno fails to find a fiducial, it pauses and displays an error message. If a fiducial is found, the corresponding XY offset and rotation is applied for the remainder of the file run. For more information see “G170 Find Fiducials” in the System Programming section.

Connection Status Box

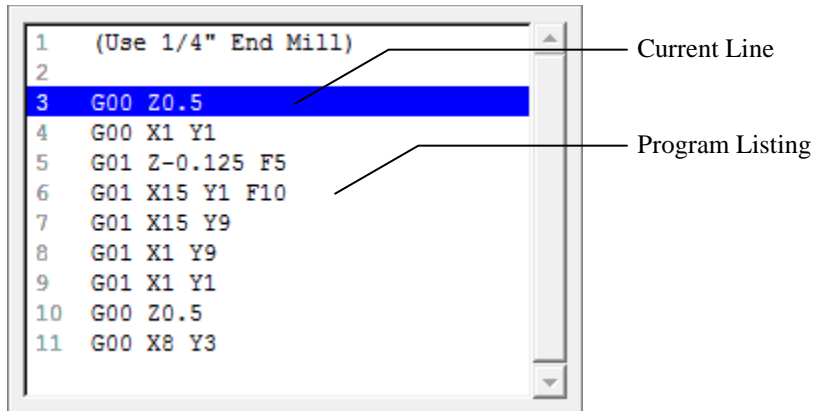
The Connection Status Box indicates whether or not the software is connected to the Signal Generator.



Connection Indicator – The connection indicator is green when the software is connected to the Signal Generator.

Program Listing Box

The Program Listing Box displays the current G-Code program and highlights the current line. You can use the scroll bar to view the entire program.



Program Listing – A listing of the current G-Code program.

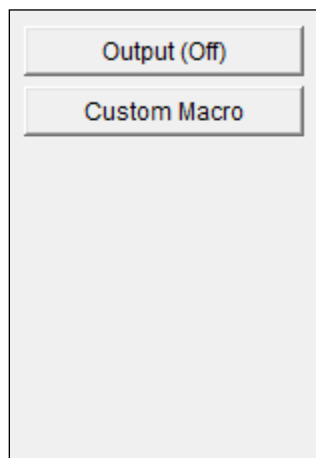
Current Line – The line currently being executed, or about to be executed by Techno.

There are several Program Listing Box settings that affect system performance. See “Performance” in the Initial Setup section of this manual for more information.

Note that you can open the Editor dialog box by double-clicking the Program Listing Box.

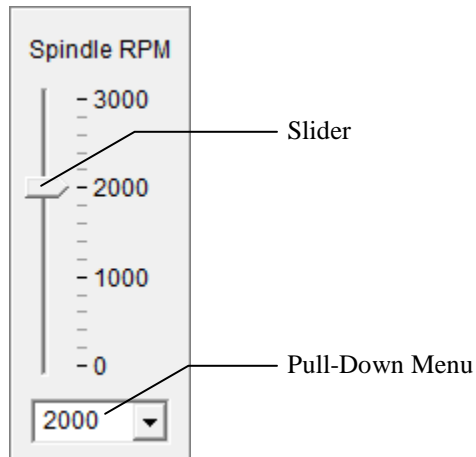
Digital Output Box

The Digital Output Box displays buttons for any output line or macro configured to be visible on the main screen. See “Output Line Settings” and “M-Code Definitions” in the Initial Setup section for more details.



Variable Output Box

The Variable Output Box provides controls for variable outputs including analog voltage and PWM/PFM signals. You may use this box to control an external device, such as a spindle speed controller. For information on configuring this box, see “Variable Output Settings” in the Initial Setup section.



Slider – Allows setting the variable output to any value in the range defined on the Variable Output panel of the Configuration dialog box.

Pull-Down Menu – Shows all the preset choices defined on the Variable Output panel of the Configuration dialog box.

3. Initial Setup

This section describes how to set up Techno for use with your machine tool. It's very important that the software and hardware are set up properly. Incorrect setup may cause the machine tool to behave in a potentially dangerous manner. Please read through this section carefully to gain a thorough understanding of how Techno controls your machine.

Windows Setup

Since Techno is a real time control program, it must have full control of the operating system while running. It is very important that you do the following before running Techno:

Disable all screen savers and power management programs.

Make sure there are no background programs running such as back-up software and calendar reminders.

Make sure no other programs are open.

Software Setup

The Configuration Files

All software settings are stored in three files:

- Setup file (extension "STP"): All machine-related settings
- Tooling file (extension "TLG"): All tooling settings
- User file (extension "USR"): All user preferences (not directly related to the machine or tooling)

Configuration Wizard

The first time you run the software Techno launches the configuration wizard. The configuration wizard asks a series of questions about your machine and the control system you purchased, then uses that information to set many configuration settings to appropriate default values. It's very important that you go through the wizard and answer each question correctly.

If you purchased a Techno machine tool the configuration wizard will completely configure your system so you can get started with your machine right away. However, we highly recommend reading this section to understand all the configuration options available to you in the system.

If you purchased a control system or retrofit kit, the wizard creates a default Setup file that serves as a starting point for the remainder of the configuration process. You will still need to adjust some of the system's configuration settings to run

your machine properly. The rest of this section describes the settings in detail. To get started, pay close attention to these configuration panels:

- Motor Signals
- Basic Definition
- Drive Parameters
- Basic Homing
- Feedrate / Ramping
- Input Lines

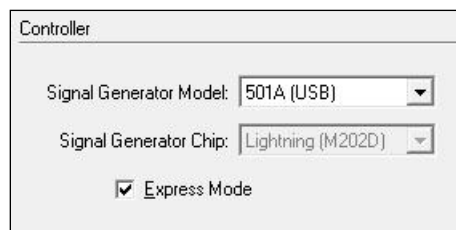
For machines that operate in more than one mode, such as combination mill-lathe machines, you'll create a separate Setup file for each mode of operation. If you use Techno's tool offset and compensation features extensively, you may find it convenient to create several Tooling files.

Each time you start the program, Techno will automatically load the last Setup and Tooling files used. There is only one User file, which Techno loads automatically at startup.

Each configuration panel has a Display Shortcut checkbox in the upper right corner. Techno creates a command in the Configuration menu for each selected panel, allowing quicker access to configuration panels that you use frequently.

Controller Settings

The Controller command on the Configuration menu displays the following panel:



The screenshot shows a window titled "Controller" with the following settings:

- Signal Generator Model: 501A (USB)
- Signal Generator Chip: Lightning (M202D)
- Express Mode

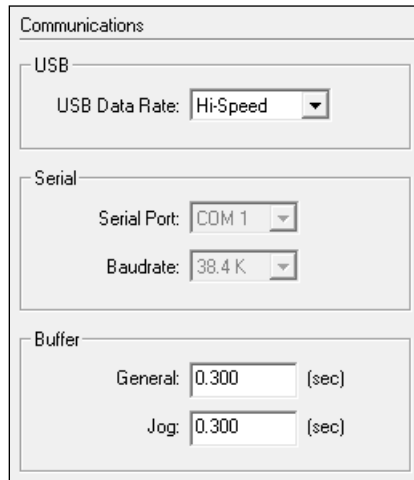
Signal Generator Model – Specifies the type of Signal Generator you're using. The two models are 401A (serial port connection) and 501A (USB connection).

Signal Generator Chip – (Model 401A Signal Generator only) Indicates which chip version is installed in the Signal Generator. If you're not sure of your chip version, you can look on the label of the 40 pin microprocessor inside of your Signal Generator.

Express Mode – Allows the controller to simultaneously control 4 or 5 axes at a time. Express Mode also allows step rates of up to 100,000 steps per second.

Communications Settings

The Communications command on the Configuration menu displays the following panel:



The screenshot shows a software interface for configuring communications. It is titled "Communications" and contains three main sections: "USB", "Serial", and "Buffer".

- USB:** Contains a "USB Data Rate" dropdown menu currently set to "Hi-Speed".
- Serial:** Contains a "Serial Port" dropdown menu set to "COM 1" and a "Baudrate" dropdown menu set to "38.4 K".
- Buffer:** Contains two input fields: "General" and "Jog", both set to "0.300" with "(sec)" next to them.

USB (Model 501A Signal Generator only)

USB Data Rate – Allows you to specify the speed of USB communications with the signal generator. If your machine is exposed to excessive electrical noise, setting the USB Data Rate to Full Speed helps to prevent communication problems.

Serial (Model 401A Signal Generator only)

Serial Port – The serial port your computer will use to communicate with the Signal Generator. Typically, this will be COM 1, COM 2, or COM 3.

Baud Rate – The speed at which Techno communicates across the serial port with the Signal Generator. It is measured in bits per second. The standard setting is 38,400, which is fast enough for most applications. If you're running the Signal Generator at high step rates use 57,600 or 115,200. If you're getting communications errors, try 9600 to help troubleshoot the problem (and move the machine only at slow speeds).

Buffer

The Signal Generator buffer prevents system events (such as screen updates) from affecting motor movement on the machine tool. The larger the buffer, the less effect system events have on motor movement. The smaller the buffer, the more responsive the machine tool is to mouse clicks and keyboard commands. In most cases, the lag-time between the PC and the motor movement is imperceptible. The value can range from 0.01 to 1.0 seconds. Slower computers may require a higher value.

General – Sets the buffer size on the Signal Generator for all operations except jogging in continuous mode.

Jog – Sets the buffer size on the Signal Generator for jogging in continuous mode. Typically this value is lower than the General value, to make the jog buttons more responsive.

Motor Signal Settings

Techno provides three signals for motor drivers: step, direction, and enable. Different driver manufacturers have varying requirements for the polarity and timing of these signals. Techno provides the flexibility to tailor the motor signals to run any driver.

The Motor Signals command on the Configuration menu displays the following panel:

Motor Lines	Axis Letter	Subaxis	Motor Direction
1	X	N/A	Positive
2	Y	N/A	Positive
3	Z	N/A	Positive
4	A	N/A	Positive
5	B	N/A	Positive

Signals

Driver Model: FlashCut 8A Pro Series Micro Stepper (6501-x-080-M)

Driver Type: Stepper

Step Pulse Polarity: High

Step Pulse Activation: Normal

Step Pulse Width: 3 (µsec)

Min Time Between Steps: 3 (µsec)

Direction-Step Setup: 3 (µsec)

Min Step-Direction Lag: 3 (µsec)

Enable Signal Polarity: High

Dynamic Enable Line

Enable Signal State

On Power-Up: Motors Disabled

After Disconnect from Sig. Gen.: Motors Enabled

After Limit/Safety Error: Motors Enabled

Motor Lines

Motor lines are the physical lines on the Signal Generator used for driving motors.

Axis Letter – Assigns a logical axis letter (X, Y, Z, A, B, C) to a physical Signal Generator motor line (1, 2, 3, 4, 5). If you have more than one motor driving an axis on your machine, you may assign the same axis letter to more than one motor line. For example, if there were two motors driving a milling machine Y axis, wired to physical motor lines 2 and 4 on the Signal Generator, the settings would be:

Motor Lines			
	Axis Letter	Subaxis	Motor Direction
1	X	N/A	Positive
2	Y	1	Positive
3	Z	N/A	Positive
4	Y	2	Positive
5	None	N/A	Positive

Subaxis – Index number that allows the rest of the Techno system to distinguish between multiple motors used on the same axis. This is important for homing, when Techno handles each motor independently. See “Input Line Settings” later in this section for more information.

Motor Direction – Sets the direction of rotation for the motor connected to each Signal Generator axis. Depending on how a motor is wired, the same signal from the motor driver can turn it clockwise or counter clockwise. Use the jog buttons to make sure that a positive move in each axis on the screen corresponds to a positive move in each axis on the machine tool. Note that the direction of movement is defined as the direction of the tool relative to the workpiece. For example, on a typical milling machine, a positive X move (tool movement to the right) requires table movement to the left. If any direction is incorrect, change the Motor Direction from Positive to Negative (or vice-versa) to reverse the correspondence between the software and machine tool.

Signals

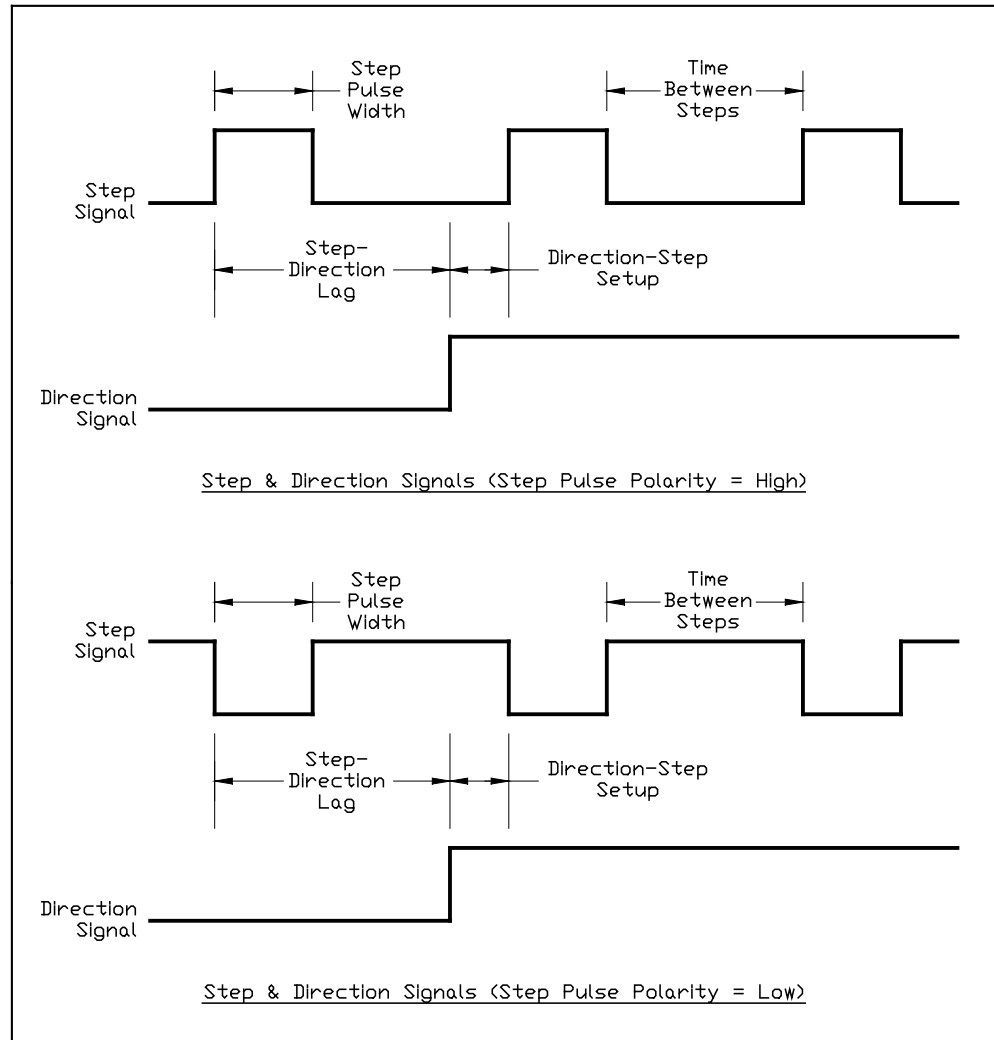
Driver Model – Sets the model of your motor driver or combination controller/driver box.

If you’re using a Techno driver or combination controller/driver box, the model number is printed on a sticker located on the back or bottom of the box. When you select your driver model, Techno enters the correct values for all other Signal settings (Step Pulse Polarity through Dynamic Enable Line). The fields are disabled and cannot be changed. If you need to change the values, choose Other (generally not necessary).

If you’re using non-Techno drivers, choose Other. This enables all fields, allowing you to enter the proper settings based on your motor driver requirements. Note that some Signal fields do not apply to all Signal Generator models, and may be disabled based on your Controller panel settings (Signal Generator Model and Chip).

Driver Type – Indicates your motor driver type: stepper or servo.

Step Pulse Polarity – Sets the polarity of the step signal to High or Low. See the diagram below.



Step Pulse Activation – (Model 401A Signal Generator only) Indicates how the Signal Generator creates step pulses as follows:

- Normal: The Signal Generator times the duration of each step pulse
- Inverted: The Signal Generator times the delay between step pulses, while guaranteeing the step pulse duration is at least the Step Pulse Width entered. When the motor driver requires a long step pulse, this mode yields higher step rates.

Step Pulse Width – Sets the duration of the step pulse (in microseconds). See the diagram above.

Min Time Between Steps – Sets the minimum time between step pulses (in microseconds). See the diagram above.

Direction-Step Setup – Sets the time between a direction change and the leading edge of the subsequent step pulse (in microseconds). See the diagram above.

Min Step-Direction Lag – Sets the minimum time from the leading edge of a step pulse to a subsequent direction change (in microseconds). See the diagram above.

Enable Signal Polarity – Sets the polarity of the enable signal. Choose High if the motor driver is enabled by a high signal, or Low if the motor driver is enabled by a low signal.

Dynamic Enable Line – Sets the behavior of the enable line. When checked, the enable line turns on when any axis is in motion, and turns off when all axes are idle. Generally this box should be unchecked.

Enable Signal State

On Power-Up – (Model 501A Signal Generator only) Sets the state to which the Signal Generator drives the motor enable line when turned on, to initially enable or disable the motors.

After Disconnect from Signal Generator – Sets the state to which the Signal Generator drives the motor enable line when you disconnect from the Signal Generator.

After Limit/Safety Error – Sets the state to which the Signal Generator drives the motor enable line after a limit or safety error. See “Input Line Settings” later in this section for more information on these errors.

Performance Settings

The Performance command on the Configuration menu displays the following panel:

Program Listing Box

Display – Affects how the software displays the Program Listing Box when running a G-Code file in continuous mode. When running G-code files with many short moves, the Techno software runs faster if you select “Current Line,” and runs fastest if you select “Nothing”.

Update Period – The period at which Techno updates the Program Listing Box. To improve system performance, the Program Listing Box periodically highlights the line of G-Code being executed, rather than highlighting every line executed.

Show Line Numbers – Determines whether or not Techno displays line numbers in the Program Listing Box.

File Loading

Preprocess – If checked, Techno checks the G-Code file for errors whenever a file is opened or reset. If unchecked, Techno skips this step. When running very large files, you may want to uncheck this option to save time opening and resetting G-Code files.

Preview in Viewports – If checked, Techno displays a preview of the toolpath.

Calculate Run Time – If checked, Techno will estimate the run time for the program.

Reset Automatically – Determines whether Techno redraws and rechecks a G-Code file each time you take action that affects processing of the file (such as resetting program coordinates). If you uncheck this checkbox, you will need to reset the file manually just before running the file (using the Reset button on the G-Code Control Panel).

Buffer Sizes

Loading, Buffering, Reverse, Timing – These internal buffer sizes are advanced settings that you won't need to adjust except under guidance from technical support.

Display Update Period

DRO's – Determines how often the coordinates are updated in the DRO Box while the machine tool is moving. Higher values may increase performance on slower computers.

Viewports – Determines how often the toolpath trace is updated in the Viewport Box while the machine tool is moving. Higher values may increase performance on slower computers.

Advanced

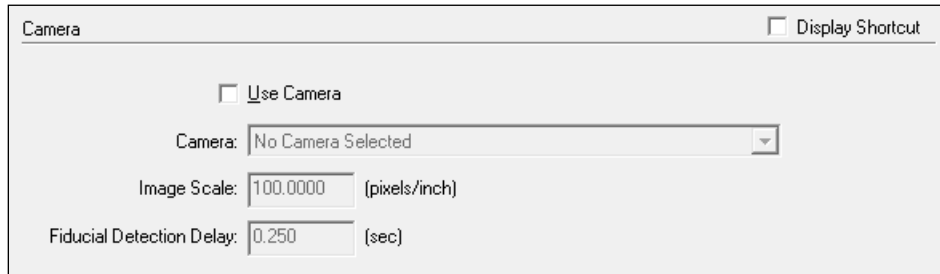
Timing Factor – (Model 401A Signal Generator only) Sets the precision for timing on interpolated moves. The higher the number, the greater the precision and the lower the system's top speed. For digital servo drivers, set this value to 2. For stepper drivers, set this value between 3 and 10.

Start Delay – Sets the duration of the delay that occurs when you click the Start button on the G-Code Control Panel (in Continuous mode only). This delay is required on slower PC's to allow background processing to get well ahead of machine tool motion.

Use Advanced Diagnostics – Directs Techno to save additional diagnostic information to the support file. Leave this unchecked unless otherwise directed by a support engineer.

Camera Settings

The Camera command on the Configuration menu displays the following panel:



The screenshot shows a window titled "Camera" with a "Display Shortcut" checkbox in the top right corner. Inside the window, there is a "Use Camera" checkbox, which is currently unchecked. Below this is a "Camera:" label followed by a dropdown menu showing "No Camera Selected". Underneath the dropdown is an "Image Scale:" label with a text input field containing "100.0000" and the unit "(pixels/inch)". At the bottom, there is a "Fiducial Detection Delay:" label with a text input field containing "0.250" and the unit "(sec)".

Use Camera – Indicates whether or not Techno displays the video window. You may use the video window for a variety of purposes including setting program zero and using the Fiducial Correction System. See “Fiducial Box” in the Main Screen Features section for more information on the Fiducial Correction System.

Camera – The camera that provides video to the Video window. Choices include all USB cameras connected to your PC.

Image Scale – The scaling factor for converting pixels in the video window to length units.

Fiducial Detection Delay – The duration of the delay between moving to a fiducial and beginning the fiducial detection process. This delay allows the machine tool to settle before capturing an image with the camera, which improves fiducial correction results.

General System Settings

The System:General command on the Configuration menu displays the following panel:

System Units – Indicates the units for all software settings, and the default units when interpreting G-Code files. If a G-Code file uses different units, the G20 or G21 command must be used (see “G20, G21 Inch Units and Metric Units” in the System Programming section). When you change the System Units, Techno gives you the option to convert all of your settings to the new units.

Save/Restore Machine State

Save – Specifies when Techno will save the current machine state. The machine state includes

- Machine tool position
- Program Zero location (including current fixture offset)
- Relative Zero location
- Machine coordinates state (set or cleared)
- Tool loaded
- Tool offset in effect
- Last direction moved (per axis)

The two choices for when to save the machine state are:

- Only at Program Exit

When you exit Techno it saves the machine state to a file. This is the standard setting.

- Frequently

Techno saves the machine state whenever it can, including whenever the machine comes to a rest, or whenever you do something to change the state (e.g. zero program coordinates or change tools). This guarantees the system will have the most recent machine state possible in case of a power outage that terminates the software unexpectedly.

Restore – Specifies what action Techno will take at program startup to restore the saved machine state:

- Never

Techno never restores the saved machine state.

- Ask at Startup

Techno displays a dialog asking if you want to restore the machine state.

- Always at Startup

Techno restores the machine state without displaying any message.

Move After Zeroing Program Coordinates

After you zero a single program coordinate or all program coordinates, Techno can automatically retract an axis from the workpiece. This makes subsequent moves safer, since the tool will not be touching the workpiece when the next move occurs.

Axis – Specifies the axis that will move after setting its program coordinate to zero.

Distance – The distance and direction the axis will move. The value may be positive or negative.

Move After Feed Hold

After a feed hold occurs, Techno can move the tool away from the workpiece to avoid producing witness marks. Generally this feature is used to raise the Z axis on a mill or router.

Axis – Specifies the axis that will move after a feed hold occurs.

Coordinate System – Specifies the coordinate system for the endpoint of the move as follows:

- Program
Techno will move to the specified Program Coordinate
- Machine
Techno will move to the specified Machine Coordinate
- Incremental
Techno will move the specified distance incrementally

Coordinate – The endpoint for the move (or distance for Incremental mode)

Return to Toolpath after Start

When you pause a G-Code program and move the machine tool away from the toolpath, then continue running the G-Code program, Techno provides an option to move back to the toolpath and continue running (when you click Start). The settings below let you indicate when and how much to slow down the machine as it nears the end of the return move.

Linear Move Offset – Distance from the end of the return move to slow down, for all linear axes.

Rotary Move Offset – Distance from the end of the return move to slow down, for all rotary axes.

Linear Feedrate – Feedrate for the end of the return move, for all linear axes.

Rotary Feedrate – Feedrate for the end of the return move, for all rotary axes.

Basic Definition Settings

The Basic Definition command on the Configuration menu displays the following panel:

	Axes In Use	Limited Travel	Length (inches)	Display Order
X	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	40.0000	1
Y	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	40.0000	2
Z	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	15.0000	3
A	<input checked="" type="checkbox"/>	<input type="checkbox"/>	360.0000 *	4
B	<input type="checkbox"/>	<input type="checkbox"/>	360.0000 *	5
C	<input type="checkbox"/>	<input type="checkbox"/>	360.0000 *	6

General

Machine Type – Specifies what type of machine you are using. The choices are:

- Mill
- Router
- Plasma
- Waterjet
- Laser
- Lathe
- Hot Wire
- Wire EDM
- Punch Press
- General

Lathe Tool – Indicates which side of the lathe the tool is primarily located. Your choice sets the orientation of the X+ direction, on the Jog Control Panel and in the Viewport Box. This lets you match the direction on the screen to the actual direction of movement of the tool relative to you (for lathes only).

Axis Settings

Axes In Use – Indicates whether or not the axis is used on your machine tool.

Limited Travel – Indicates whether or not the axis travel is limited. Many rotary axes do not have limited travel.

Length – Sets the length of travel of each axis of your machine tool. You may want to define the axis length slightly smaller than the values published by the machine tool manufacturer. This will leave some room for error.

Display Order – Allows you to specify the order in which the axes are displayed on the main screen.

Advanced Definition Settings

The Advanced Definition command on the Configuration menu displays the following panel.

The screenshot shows the 'Advanced Definition' dialog box with the 'Axis Settings' section expanded. The 'Activate' checkbox is checked. The settings for each axis are as follows:

Axis	Pt. Move Linear Interp.	Parallel To	Right Handed	Axis Group	Hidden	Safe Envelope Min	Safe Envelope Max
X	Yes	X	Yes	1	<input type="checkbox"/>	0.0000	0.0000
Y	Yes	Y	Yes	1	<input type="checkbox"/>	0.0000	0.0000
Z	No	Z	Yes	1	<input type="checkbox"/>	0.0000	0.0000
A	No	X	Yes	1	<input type="checkbox"/>	0.0000 *	0.0000 *
B	No	Y	Yes	1	<input type="checkbox"/>	0.0000 *	0.0000 *
C	No	Z	Yes	1	<input type="checkbox"/>	0.0000 *	0.0000 *

Axis Settings

Activate – Activates the Advanced Definition settings.

Point Move Linear Interpolation – Determines if the motion for the axis will be included in the interpolated move for a Point move or G28/G29/G30 command (see “Point Control Panel” in the Main Screen Features section of this manual).

Parallel To – Indicates the orientation of the axis. For rotary axes this setting indicates the axis of rotation.

Right Handed – Indicates whether or not the rotary axis obeys the right hand rule. If you curl your hand closed in the direction the rotary axis revolves and your thumb points in the positive direction of the axis of rotation (Parallel Axis setting), the axis is right-handed.

Axis Group – Some machine tools use two separate groups of axes to cut a part. For instance, a hotwire machine uses a pair of perpendicular axes to move each end of the wire. Techno considers each pair of axes to be an axis group. When executing a move, Techno sets the motor speeds such that the faster-moving end of the wire (axis group) moves at the specified feedrate.

Hidden – Allows an axis to be hidden from the main screen. A typical use is an axis used in a tool changer. Techno controls the axis the same as any other, but the axis is not visible to the operator. The Hidden Axes command on the main screen's View menu allows you to temporarily display the axis (for setup, maintenance and so on).

Safe Envelope Min – The machine coordinate of the safe envelope lower bound. For more information, see “Coordinates Menu” in the Main Screen Features section.

Safe Envelope Max – The machine coordinate of the safe envelope upper bound.

Rotary Axes Settings

The Rotary Axes command on the Configuration menu displays the following panel:

Rotary Axes
 Display Shortcut

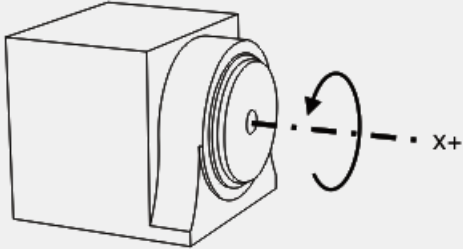
Rotary Axis Configuration: Rotary Table (Four Axes)

Rotary Table

Rotary Axis: A

Parallel To: X

Right Handed: Yes



General

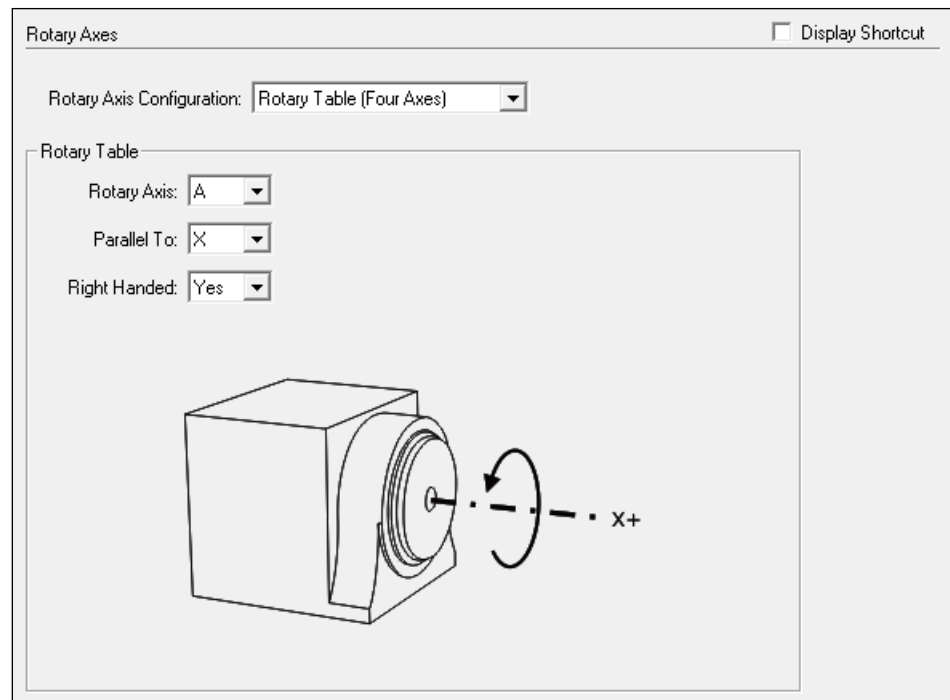
Fixture Offset for Rotary Zero Position: None

F Command Interpretation for Rotary Moves: Linear Feedrate (Distance/Minute)

Rotary Axis Configuration – Techno supports the following options:

- Rotary Table (Four Axes)
Select Rotary Table if you have only one rotary axis.
- Dual Rotary Table (Five Axes)
Select Dual Rotary Table if you are using two rotary axes built into a single table.
- Rotary Knife
Select Rotary Knife if you are running a CNC knife cutting machine.
- None
Select None if you are not using a rotary axis.

Rotary Table

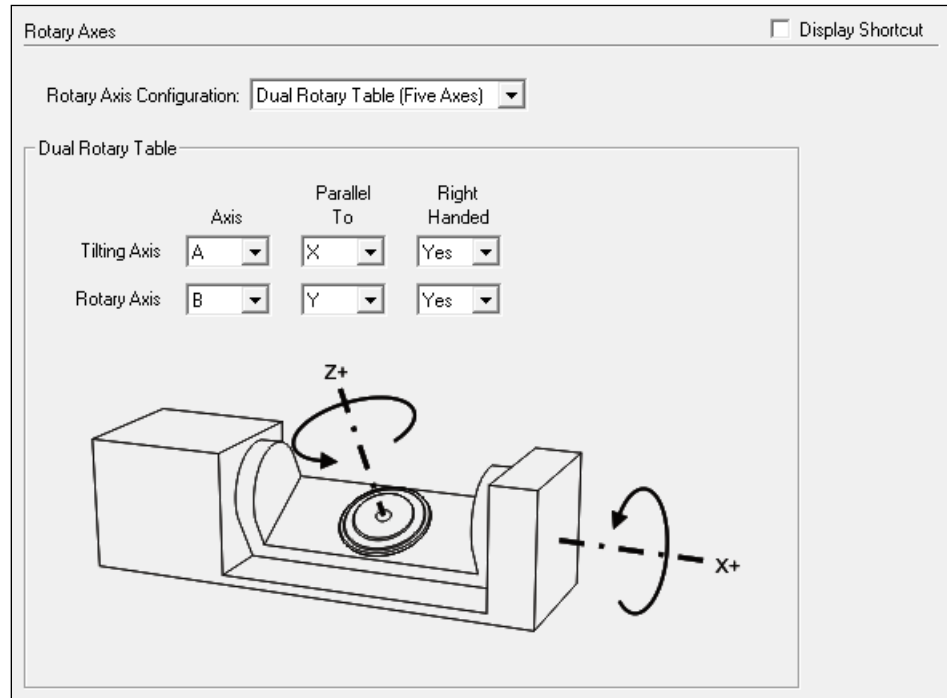


Rotary Axis – The rotary axis letter (A, B or C)

Parallel To – See “Parallel To” under Advanced Definition in this section

Right Handed – See “Right Handed” under Advanced Definition in this section.

Dual Rotary Table



Tilting Axis – The axis directly attached to the mill/router bed.

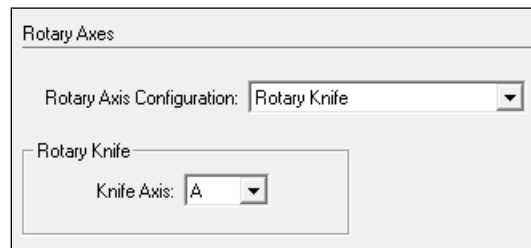
Rotary Axis – The axis positioned by the tilting axis. Typically it is parallel to the Z axis when the tilting axis is at program zero.

Axis – The axis letter for the tilting and rotary axes (A, B or C).

Parallel To – See “Parallel To” under Advanced Definition in this section

Right Handed – See “Right Handed” under Advanced Definition in this section.

Rotary Knife



Knife Axis – The rotary axis letter (A, B or C)

General

Fixture Offset for Rotary Zero Position – Indicates the fixture offset for the rotary zero position.

- 4-Axis Machining

Rotary zero is anywhere along the axis of rotation of the rotary axis. The parallel axis in the fixture offset is ignored.

- 5-Axis Machining

Rotary zero is at the intersection of the axes of rotation of both rotary axes.

If you select None, you must set program zero at rotary zero before machining a part. Otherwise, you may set program zero anywhere you choose.

For more information see “Fixture Offset Settings” later in this section.

F Command Interpretation for Rotary Moves – For moves that include rotary motion, this setting tells Techno whether to interpret the feedrate command (F) as a linear or rotary feedrate.

- Linear Feedrate

Techno uses the distance from the axis of rotation to calculate the proper rotary axis speed from the commanded linear feedrate.

- Rotary Feedrate

Techno interprets the feedrate command as a rotary feedrate in degrees per minute.

For more details, see the description for the G01 command in the “System Programming” section.

Drive Parameters Settings

The Drive Parameters command on the Configuration menu displays the following panel:

Drive Parameters		<input type="checkbox"/> Display Shortcut
X	Stepper Control Step Mode: 4 (micro steps/full step) Motor Res: 200 (full steps/motor rev)	Lead Screw Gear Ratio: 1.00000000 (motor rev/screw rev) Thread Count: 20.00000000 (screw rev/inch) Correction Factor: 1.00000000
Tool Positioning Resolution: 0.000062500000 (inches)		
Y	Stepper Control Step Mode: 4 (micro steps/full step) Motor Res: 200 (full steps/motor rev)	Lead Screw Gear Ratio: 1.00000000 (motor rev/screw rev) Thread Count: 20.00000000 (screw rev/inch) Correction Factor: 1.00000000
Tool Positioning Resolution: 0.000062500000 (inches)		
Z	Stepper Control Step Mode: 4 (micro steps/full step) Motor Res: 200 (full steps/motor rev)	Lead Screw Gear Ratio: 1.00000000 (motor rev/screw rev) Thread Count: 20.00000000 (screw rev/inch) Correction Factor: 1.00000000
Tool Positioning Resolution: 0.000062500000 (inches)		
A	Stepper Control Step Mode: 4 (micro steps/full step) Motor Res: 200 (full steps/motor rev)	Rotary Gear Ratio: 72.00000000 (motor rev/axis rev) Correction Factor: 1.00000000
Tool Positioning Resolution: 0.006250000000 (degrees)		

Drive Parameter Display – Shows all drive parameters for each axis. For each one:

- The Tool Positioning Resolution is the length of axis movement per Signal Generator pulse, automatically calculated from the control and drive mechanism settings.
- A double-click in the control parameters area opens the Control Parameters dialog box. A double-click in the drive mechanism area opens the Drive Mechanism dialog box.

Axis Pull-Down Menu

Drive Parameters	
X	Stepper Control
Edit Control Parameters... Edit Drive Mechanism... Set Same As Y Set Same As Z	

Set Same As – Sets all control/drive mechanism settings the same as another axis.

Edit Control Parameters – Opens the Control Parameters dialog box. Select Stepper or Servo from the Driver Type pull-down menu.

- Stepper

Driver Type:	Stepper	
Step Mode:	4	(micro steps/full step)
Motor Resolution:	200	(full steps/motor rev)
Tool Positioning Resolution: 0.000062500000 (inches)		

Step Mode – The number of micro steps between each full motor step. Enter “1” for full-stepping, “2” for half-stepping, “4” for quarter-stepping, and so on.

Motor Resolution – The number of full motor steps for one revolution of the motor. For example, a 1.8° stepper motor has 200 full steps per revolution; a 0.9° stepper motor has 400 full steps per revolution.

- Servo

Driver Type:	Servo	
Encoder Divisor:	5	(encoder ticks/SG pulse)
Encoder Resolution:	1000	(lines/rev)
Tool Positioning Resolution: 0.000125000000 (inches)		

Encoder Divisor – The number of encoder pulses that correspond to one pulse from the Signal Generator.

Encoder Resolution – The number of encoder pulses per revolution, not including any adjustment for running in quadrature mode. When calculating the Tool Positioning Resolution, Techno assumes the encoder is running in quadrature mode.

Edit Drive Mechanism – Opens the Drive Mechanism dialog box. Select a drive mechanism from the Drive Mechanism pull-down menu. Note that Rotary is an option only for axes A, B and C.

- Lead Screw

Drive Mechanism:

Gear Reduction:
 (motor rev/screw rev)

Lead Screw Spec: in units
 (screw rev/inch)

Correction Factor:

Tool Positioning Resolution: 0.000062500000 (inches)

Gear Reduction – Enter the parameters for any gearing between the motor and lead screw. A decimal gear ratio of 1.0 indicates no gearing.

Lead Screw Specification – Enter the parameters for the lead screw.

Correction Factor – Use this value to fine tune the axis, so the distance it moves exactly matches the commanded distance.

- Rack and Pinion

Drive Mechanism:

Gear Reduction:
 (motor rev/pinion rev)

Rack Spec: in units
 (inch/tooth)

Pinion Tooth Count: (teeth)

Correction Factor:

Tool Positioning Resolution: 0.007500000000 (inches)

Gear Reduction – Enter the parameters for any gearing between the motor and pinion. A decimal gear ratio of 1.0 indicates no gearing.

Rack Specification – Enter the parameters for the rack.

Pinion Tooth Count – The number of teeth on the pinion.

Correction Factor – Use this value to fine tune the axis, so the distance it moves exactly matches the commanded distance.

- Belt

Drive Mechanism:

Gear Reduction:
 (motor rev/pulley rev)

Belt Spec: in units
 (inch/tooth)

Pulley Tooth Count: (teeth)

Correction Factor:

Tool Positioning Resolution: 0.007500000000 (inches)

Gear Reduction – Enter the parameters for any gearing between the motor and drive pulley. A decimal gear ratio of 1.0 indicates no gearing.

Belt Specification – Enter the parameters for the belt.

Pulley Tooth Count – The number of teeth on each pulley.

Correction Factor – Use this value to fine tune the axis, so the distance it moves exactly matches the commanded distance.

- General Linear

Drive Mechanism: <input type="text" value="General Linear"/>
Drive Ratio: <input type="text" value="Inches per Motor Rev"/>
<input type="text" value="0.05000000"/> (inch/motor rev)
Tool Positioning Resolution: 0.000062500000 (inches)

Drive Ratio – Enter the distance moved per motor revolution or the number of motor revolutions per unit distance.

- Rotary

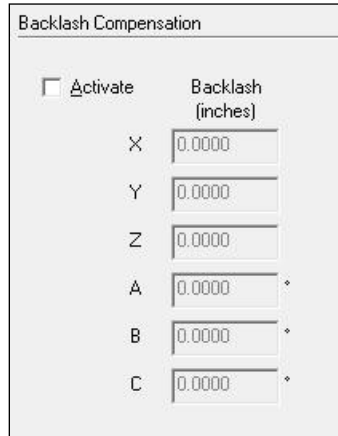
Drive Mechanism: <input type="text" value="Rotary"/>
Gear Reduction: <input type="text" value="Enter Decimal Gear Ratio"/>
<input type="text" value="72.00000000"/> (motor rev/axis rev)
Correction Factor: <input type="text" value="1.00000000"/>
Tool Positioning Resolution: 0.006250000000 (degrees)

Gear Reduction – Enter the parameters for any gearing between the motor and the rotational axis. A decimal gear ratio of 1.0 indicates no gearing.

Correction Factor – Use this value to fine tune the axis, so the distance it moves exactly matches the commanded distance.

Backlash Compensation Settings

The Backlash Compensation command on the Configuration menu displays the following panel:



Axis	Backlash (inches)
X	0.0000
Y	0.0000
Z	0.0000
A	0.0000 *
B	0.0000 *
C	0.0000 *

CNC control works best if there is little or no backlash on the machine tool. However, if your machine doesn't have a zero-backlash drive mechanism (such as ball screws or anti-backlash nuts), Techno can compensate for the backlash. Immediately after an axis changes direction, Techno turns the motor a small amount to absorb the backlash before performing the next commanded move.

Activate – When checked, Techno will use backlash compensation for all direction changes.

Backlash – The amount of Backlash for each axis.

► To Set Backlash

1. Make sure the Activate checkbox is unchecked, and then choose OK.
2. Choose the Jog button on the Control Selection Box.
3. Drive the X axis in either direction at least 0.25" (to take out any backlash in that direction).
4. Zero the relative coordinates by choosing Zero All from the Set button next to the Relative label.
5. Choose the Step radio button. Jog the axis step by step in the opposite direction until you detect table movement (using a dial indicator).
6. The Relative coordinate X axis value is the amount of X axis backlash on your machine tool.
7. Write down this number and repeat the above process at different places along the X axis.
8. Choose Backlash Compensation from the Configuration menu. The Backlash Compensation panel of the Configuration dialog box will appear.

9. Record the average of all backlash values in the X axis Backlash text box. If you have no backlash on an axis, or if you don't want backlash compensation on an axis, enter zero.
10. Repeat the above steps for each axis. When you're finished, select the Activate checkbox and make sure there is a check in it.

Basic Homing Settings

The Homing command on the Configuration menu displays the following panel:

Basic Homing Display Shortcut

Axis Settings

	Home End	Home Order	Home Axis	Home Switch Offset (inches)	Homing Rate (in/min)	Homing Tolerance (inches)
X	<input type="text" value="Neg"/>	<input type="text" value="2"/>	<input type="text" value="Yes"/>	<input type="text" value="0.1000"/>	<input type="text" value="5.00"/>	<input type="text" value="0.0000"/>
Y	<input type="text" value="Neg"/>	<input type="text" value="3"/>	<input type="text" value="Yes"/>	<input type="text" value="0.1000"/>	<input type="text" value="5.00"/>	<input type="text" value="0.0000"/>
Z	<input type="text" value="Pos"/>	<input type="text" value="1"/>	<input type="text" value="Yes"/>	<input type="text" value="0.1000"/>	<input type="text" value="5.00"/>	<input type="text" value="0.0000"/>
A	<input type="text" value="Neg"/>	<input type="text" value="4"/>	<input type="text" value="No"/>	<input type="text" value="2.0000"/>	<input type="text" value="360.00"/> */min	<input type="text" value="0.0000"/>
B	<input type="text" value="Neg"/>	<input type="text" value="5"/>	<input type="text" value="No"/>	<input type="text" value="2.0000"/>	<input type="text" value="360.00"/> */min	<input type="text" value="0.0000"/>
C	<input type="text" value="Neg"/>	<input type="text" value="6"/>	<input type="text" value="No"/>	<input type="text" value="2.0000"/>	<input type="text" value="360.00"/> */min	<input type="text" value="0.0000"/>

Seek Home Button

Hide Results if in Tolerance

Reset Machine Zero

Always Display Results

General

Maximum Homing Attempts:

Fast Home Offset Factor:

Require Homing before Loading G-Code

G27 Command

Hide Results if in Tolerance (Resets Machine Zero)

Always Display Results

Axis Settings

Home End – The end of the axis at Machine Zero (Home).

Home Order – The axis order for the homing operation. For safety, set the order so the initial homing move retracts the cutting tool away from the workpiece.

Home Axis – Determines whether or not an axis is included in the homing operation.

Home Switch Offset – The distance each axis backs away from the home switch after the switch is tripped during homing.

Homing Rate – The feedrate at which the machine tool will move when locating home switches.

Homing Tolerance – The distance allowed from the original Machine Zero location to the new Machine Zero location found during homing (used with the Hide Results if in Tolerance radio buttons).

Seek Home Button

These settings apply to the homing process initiated by the Seek Home Button on the Home Control Panel.

Hide Results if in Tolerance – Tells Techno not to display the results of re-homing, if the discrepancy from the previous Machine Zero location is within the Homing Tolerance for all axes.

Reset Machine Zero – Tells Techno to automatically reset Machine Zero to the new location found, if the discrepancy from the previous Machine Zero location is within the Homing Tolerance for all axes.

Always Display Results – Tells Techno to always display the results of re-homing, even if the discrepancy from the previous Machine Zero location is within the Homing Tolerance for all axes.

G27 Command

These settings apply to the homing process initiated by the G27 Command. They work the same as the Seek Home Button controls described above, except that if you choose Hide Results if in Tolerance, Techno always resets Machine Zero to the new location found (no checkbox option is provided).

General

Maximum Homing Attempts – The maximum number of times the machine tool will re-home, while trying to get the discrepancy from the previous Machine Zero location within the Homing Tolerance for all axes.

Fast Homing Offset Factor – During fast homing, the destination for the initial rapid move is set as follows:

Distance from Machine Zero =

Home Switch Offset x Fast Home Offset Factor

The larger the number, the greater the safety margin for the initial rapid move.

Require Homing before Loading G-Code – When checked, a Seek Home operation must be performed before the system can run a G-Code program.

Advanced Homing Settings

The Advanced Homing command on the Configuration menu displays the following panel:

Advanced Homing			
Axis Settings			
	Execute M-Codes		Home Switch Machine Coordinate (inches)
	Before	After	
X	<input type="checkbox"/>	<input type="checkbox"/>	0.0000
Y	<input type="checkbox"/>	<input type="checkbox"/>	0.0000
Z	<input type="checkbox"/>	<input type="checkbox"/>	0.0000
A	<input type="checkbox"/>	<input type="checkbox"/>	0.0000 *
B	<input type="checkbox"/>	<input type="checkbox"/>	0.0000 *
C	<input type="checkbox"/>	<input type="checkbox"/>	0.0000 *

Execute M-Codes – You may specify M-Codes to execute before and after each axis homes. For more information on defining M-Codes see “M-Code Definitions” later in this section.

Home Switch Machine Coordinate – If the home switch is positioned at the travel limit, the machine coordinate of the home switch is zero. If the home switch is positioned somewhere within the machine envelope, set this value to the machine coordinate of the home switch location (nonzero value).

Feedrate and/Ramping Settings

The Feedrate/Ramping command on the Configuration menu displays the following panel:

Feedrate/Ramping <input type="checkbox"/> Display Shortcut					
Axis Settings					
	Start/Stop Feedrate (in/min)	Maximum Feedrate (in/min)	General Ramping Rate (in/sec ²)	Feedrate Move Ramping Rate (in/sec ²)	Continuous Contouring Feedrate Tol. (in/min)
X	10.00	200.00	2.00	2.00	10.00
Y	10.00	200.00	2.00	2.00	10.00
Z	10.00	200.00	2.00	2.00	10.00
A	360.00 */min	1440.00 */min	125.00	125.00	360.00 */min
B	360.00 */min	1440.00 */min	125.00	125.00	360.00 */min
C	360.00 */min	1440.00 */min	125.00	125.00	360.00 */min

General		Advanced	
Max Arc Feedrate:	20.00 (in/min)	<input type="checkbox"/> Use Advanced Transitions	
Point Feedrate - Linear:	10.00 (in/min)	Centripetal Acceleration Factor:	1.00
Point Feedrate - Rotary:	360.0 (*/min)	Line Sequence Smoothing Factor:	0

For your system to run accurately, it's important to tune the values on this panel correctly for your specific combination of drivers, motors and machine tool. All parameters must be set conservatively to prevent the system from losing position (losing steps). An axis can lose a step when the motor gets the signal to move a step, but isn't physically able to perform the move. The usual cause is insufficient torque at a given motor RPM. Since most stepper motors are open loop systems, there is no way of telling when a step is lost without physically measuring the movement of the axis and comparing that to the amount it should have moved. However, when stepper motors are not "over torqued" they are very reliable and accurate. For that reason, we highly recommend finding the values for the following settings at which steps are not lost, then limiting the settings to about 70% of those values. Due to variations in the drive mechanism for each axis, make sure you follow the instructions below at several positions along each axis, in both directions.

Axis Settings

Start/Stop Feedrate – Sets the initial and ending feedrate for ramped moves.

► To Set the Start/Stop Feedrates

1. Enter 499 for the X axis Start/Stop Feedrate, and 500 for the Maximum Feedrate, and then choose OK. (If your software settings do not allow you to set the Maximum Feedrate to 500, set it to the largest value the software

allows, and set the Start/Stop feedrate to 1 less than that.) Note that all settings in these steps assume inch units; if your machine is set for metric units, you should multiply these settings by 25.

2. Choose the Point button in the Control Selection Box. Select Any Point in the Point List and Incremental from the Coordinate System pull-down menu. Enter 1.0 (or 25 if in metric mode) in the X text box. Choose Feedrate from the Rate Mode pull-down menu and enter a relatively slow feedrate (such as 5) in the Linear Feedrate text box. Make sure you have room to move the X axis 1 inch, and then choose Move.
3. If the motor slips, repeat this process with a slower feedrate. If the motor doesn't slip, try a faster feedrate. Note that slight slippage can be detected using a dial indicator, homing to limit switches, or reading the values on a hand-wheel if your machine has them.
4. Repeat this process until you find the highest feedrate that doesn't cause motor slippage.
5. Now run the entire length of the X axis in both directions to make sure there is no slippage at any point on the entire axis.
6. Choose Feedrate/Ramping from the Configuration menu.
7. Enter **70%** of the value you found in the Start/Stop Feedrate text box for the X axis, and then choose OK.
8. Repeat this process for all axes.

Maximum Feedrate – Sets the maximum feedrate for the system. No matter how fast the system is commanded to move (from the G-Code, Point, Jog or MDI control panels), the axis will never move faster than this value. The system will use ramping (acceleration and deceleration) to achieve this feedrate.

► To Set the Maximum Feedrates

1. Make sure you've set the Start/Stop Feedrates correctly as described above.
2. Enter 1 in/sec² in the General Ramping Rate text box for the X axis. (This is a conservative ramping rate.)
3. Set the Maximum Feedrate to 500 in/min (or the highest value your settings allow), and then choose OK.
4. Choose the Point button on the Control Selection Box. Select Any Point in the Point List and Incremental from the Coordinate System pull-down menu. Enter 1.0 (or 25 if in metric mode) in the X text box. Choose Feedrate from the Rate Mode pull-down menu and enter a feedrate that is double the Start/Stop Feedrate for the X axis. Make sure you have room to move the X axis 1 inch, and then choose Move.

5. If the motor slips, repeat this process with a slower feedrate. If the motor doesn't slip, try a faster feedrate. Note that slight slippage can be detected using a dial indicator, homing to limit switches, or reading the values on a hand-wheel if your machine has them.
6. Repeat this process until you find the highest feedrate that doesn't cause motor slippage.
7. Now run the entire length of the X axis in both directions to make sure there is no slippage at any point on the entire axis.
8. Choose Feedrate/Ramping from the Configuration menu.
9. Enter **70%** of the highest no-slip feedrate you found in the X axis Maximum Feedrate text box.
10. Repeat this process for all axes.

General Move Ramping Rate – Sets the rate of acceleration and deceleration for rapid moves and jogging.

Feedrate Move Ramping Rate – Sets the rate of acceleration and deceleration for feedrate moves.

Ramping Rates typically range from 1 to 10 in/sec². Slower ramping rates require more time to ramp up to the maximum feedrate and to ramp down to a stop. This may become a potentially dangerous situation when using the Feed Hold button or jogging since the machine will take longer to come to a complete stop. The goal is to choose a fast ramping rate that will start and stop the tool responsively without losing steps. Fast ramping rates can also allow acceleration past resonant speeds of a stepper motor.

► To Set the Ramping Rates

1. Make sure you've set the Start/Stop and Maximum Feedrates correctly as described above.
2. Enter 1 in/sec² in the General Ramping Rate text box for the X axis. (This is a conservative ramping rate.)
3. Choose OK.
4. Choose the Point button on the Control Selection Box. Select Any Point in the Point List and Incremental from the Coordinate System pull-down menu. Enter 1.0 (or 25 if in metric mode) in the X text box. Choose Rapid from the Rate Mode pull-down menu. Make sure you have room to move the X axis 1 inch, and then choose Move. If the table doesn't reach a continuous speed before ramping down to a stop, there isn't enough room to fully ramp up and down, and you should increase the length of the move.

5. If the motor slips, repeat this process with a lower ramping rate. If the motor doesn't slip, try a higher ramping rate. Note that slight slippage can be detected using a dial indicator, homing to limit switches, or reading the values on a hand-wheel if your machine has them.
6. Repeat the above steps until you determine an optimal General Ramping Rate for the X axis.
7. Choose Feedrate/Ramping from the Configuration menu.
8. Enter **70%** of the highest no-slip ramping rate you found in the X axis General Ramping Rate text box.
9. Copy the General Ramping Rate to the Feedrate Move Ramping Rate text box. If you determine that your system would benefit from a lower ramping rate for feedrate (cutting) moves, enter a lower Feedrate Move Ramping Rate.
10. Repeat this process for all axes.
11. Note that once the optimal ramping rate is determined for each axis, you may want to re-test the Maximum Feedrates to see if they can be set any higher.

Continuous Contouring Feedrate Tolerance – Continuous contouring (also called “Look-ahead”) is a high performance feature that lets Techno run G-code files smoother and faster. The advantages of using continuous contouring are:

- Smooth transitions from one G-code line to the next if the two moves are in line with one another. There is no ramp down period and subsequent ramp up period. This minimizes witness marks, burning, melting and other quality problems sometimes encountered in machining. It also provides smooth, high speed motion for any other control application that requires it. A continuous smooth motion may span hundreds or thousands of G-code lines.
- High speed machining of point-to-point type G-Code files (composed of many short moves) used in complex 2D profiling and 3D surfacing. These G-Codes files can be executed much faster with continuous contouring.

To determine if two moves are lined up enough to allow a smooth transition without ramping, Techno calculates the change in speed of each motor through the transition, and compares that to the setting for Continuous Contouring Feedrate Tolerance. If the change in speed for each motor is less than the tolerance, Techno executes the two moves as part of the same smooth motion. Otherwise, Techno ramps down to end one motion and ramps up to begin another.

Any command that breaks the continuity of motion causes the machine tool to ramp down to a stop (such as any M code to control an auxiliary device, a G04 command for dwell, and so on).

A typical default setting for the Continuous Contouring Feedrate Tolerance is the Start/Stop Feedrate value. The higher you set the tolerance, the smoother a G-Code file will be executed (i.e., more moves will be linked together into smooth motions). If the values are set too high, loss of steps (stepper motor drivers) or servo errors (digital servo drivers) can occur. The best way to find an optimal setting is through trial and error. Incrementally raise the values and run demanding test files, checking for lost steps or servo errors.

Continuous contouring is always turned on. If you prefer that all moves ramp up and down, set the Continuous Contouring Feedrate Tolerance to zero for all axes.

General

Point Feedrate – Linear, Point Feedrate - Rotary – Sets the default feedrate values in the Linear Feedrate and Rotary Feedrate text boxes on the Point Control Panel, whenever you start the Techno program. These values are also used for positioning the machine after you've jumped to a new G-Code line (see "G-Code Control Panel" in the Main Screen Features section of this manual). Enter the feedrate you most commonly use for Point moves.

Max Arc Feedrate – (Model 401A Signal Generator only) Sets the maximum feedrate for arc moves (G02 and G03 commands). Due to the computations involved during circular interpolation, an arc cannot be executed as fast as a line. If your system is losing position during arc moves only, you can use this setting to limit the speed of all arcs.

- ▶ To Set the Maximum Arc Feedrate
 1. Write a G-Code program that moves in a circle at the Maximum Feedrate. For example, if the Maximum Feedrate were 100:

```
G00 X2 Y2  
  
G02 X2 Y2 I1 J1 F100
```
 2. Run the program and notice if either of the motors loses steps. If so, decrease the feedrate in the program.
 3. Repeat the above process until neither motor loses any steps. Then enter **70%** of the final feedrate from the program into the Maximum Arc Feedrate text box. If there was no loss of steps when the Maximum Feedrate was used, enter the Maximum Feedrate into the text box.

Advanced

Use Advanced Transitions – Smooths transitions from one G-code line to the next if the two moves are tangent with one another but their feedrates are different. This can occur when a fast linear move leads into an arc with a small radius. Since Techno slows down for arcs with small radii, the feedrate changes at the junction of the two moves. The Use Advanced Transitions checkbox causes the system to ramp smoothly between the two moves, rather than ramping down to the Start/Stop Feedrate.

Centripetal Acceleration Factor – Techno slows down for arcs with small radii to avoid exceeding the Feedrate Move Ramping Rate on either axis while moving through the arc. This factor adjusts the amount Techno slows down. The lower the value, the slower the machine will move through arcs. The default value is 1.0.

Line Sequence Smoothing Factor – Techno has an advanced smoothing algorithm available to improve cut quality when the system moves through a series of short lines. The default value of zero deactivates the algorithm. The higher the number, the smoother the motion. Any positive integer is allowed, but you should start at one and increase the value slowly while testing your results.

Motion Delay Settings

The Motion Delays command in the Configuration menu displays the following panel:

Direction Change Delays

Direction Change Delay – The direction change delay is a brief pause that occurs when a motor changes direction. It gives the stepper motor time to settle and come to a complete rest before moving in the opposite direction. A short delay helps prevent loss of position on some stepper systems, depending on the driver, motor and mechanics. A delay can also improve part quality in some applications. Note that the direction change delay is not used when a motor changes direction during circular interpolation or during a continuous contouring motion.

► To Set the Direction Change Delay

1. Write a G-Code file that goes back and forth on a given axis at the Start/Stop Feedrate. For example, if the Start/Stop Feedrate were 8:

```
G01 X2 F8
X0
X2
X0
```

2. Set the Direction Change Delay to 0.01 seconds for the given axis.

3. Run the program and notice if the motor loses steps. If so, increase the Direction Change Delay. Otherwise decrease the value.
4. Repeat the above process until you reach a reasonable delay time that eliminates any motor slippage. Note that this number is typically between 0.0 and 0.1 seconds. If you do not see any slippage at a delay of 0 seconds, it is recommended you enter at least 0.01 seconds.
5. Repeat the above process for all axes.

Corner Delays

Select one of the three available options.

- None

There are no corner delays.

- Use Single Delay Value

The delay you enter will be used for every corner regardless of the feedrate.

- Calculate Delay Based on Feedrate

Techno will use a simple linear function to calculate the delay based on the current feedrate. To create the linear function you'll enter the delay and associated feedrate for each endpoint. Note that the corner delay is constant before and after the endpoints of the linear function.

Minimum Angle for Delay – The smallest angle at which a delay will occur. If the angle at a corner is smaller than this value no delay will occur.

Check Angle In Planes – Use the Add and Remove buttons to specify which planes will generate corner delays.

Reference Point Settings

Reference points provide a way to move the machine tool to a predefined location. They are accessible using the G28 and G30 commands (described in the “System Programming” section of this manual). Also, you may add reference points to the Point List displayed on the Point Control Panel.

The Reference Point command on the Configuration menu displays the following panel:

Reference Points <input type="checkbox"/> Display Shortcut								
G-Code Cmd	Description	Coord System	X Coord. (inches)	Y Coord. (inches)	Z Coord. (inches)	A Coord. (degrees)	B Coord. (degrees)	C Coord. (degrees)
G28	Tool Change	Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P2		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P3		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P4		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P5		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P6		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P7		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P8		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P9		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P10		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P11		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P12		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P13		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P14		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P15		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P16		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P17		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P18		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P19		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
G30 P20		Mach	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

G-Code Cmd – This field identifies the command in a G-Code file that moves the machine to the point (not editable).

Description – Enter a description for the point. For G28, the standard description is “Tool Change”.

Coordinate System – Choose between machine and program coordinates.

X, Y, Z, A, B, C Coordinates – Enter the coordinate for each axis.

Fixture Offset Settings

Fixture offsets provide a way to set Program Zero to a predefined location. They are activated using the G54-G59 and G54.1 commands (described in the System Programming section of this manual). The XYZABC coordinates are defined in machine coordinates. These coordinates represent the offset from Machine Zero to the new Program Zero location.

The Fixture Offsets command on the Configuration menu displays the following panel:

Fixture Offsets								<input type="checkbox"/> Display Shortcut
G-Code	Description	X Offset (inches)	Y Offset (inches)	Z Offset (inches)	A Offset (degrees)	B Offset (degrees)	C Offset (degrees)	
54		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
55		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
56		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
57		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
58		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
59		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P1		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P2		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P3		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P4		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P5		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P6		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P7		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P8		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P9		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P10		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P11		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P12		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P13		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P14		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P15		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P16		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P17		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P18		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P19		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P20		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P21		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P22		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
54.1 P23		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	

G-Code – This field identifies the ‘G’ command in a G-Code file that sets Program Zero to the offset (not editable).

Description – Enter a description for the offset.

X, Y, Z, A, B, C Offset – For each axis, enter the offset from machine zero to program zero (i.e., the machine coordinate when the machine tool is at the desired program zero location).

Threading Settings

Techno supports thread cutting on lathes equipped with a spindle encoder.

The Threading command on the Configuration menu displays the following panel:

Threading

Hardware

Spindle Encoder Resolution: 1000 (pulses per rev)

Spindle Encoder has Index Channel

Toolpath

Minimum Depth of Cut: 0.0020 (inches)

Final Pass Depth of Cut: 0.0200 (inches)

Spring Passes: 1

Hardware

Spindle Encoder Resolution – Sets the number of encoder pulses per revolution.

Spindle Encoder has Index Channel – Indicates whether or not the encoder has an index channel (one pulse per revolution).

Toolpath

Minimum Depth of Cut, Final Pass Depth of Cut, Spring Passes – See “G76 Thread Cutting Canned Cycle Command” in the System Programming section for explanations of these settings.

Input Line Settings

Techno monitors the Signal Generator’s digital input lines for a variety of purposes:

- Limit sensing
- Home sensing
- General safety (e.g. a servo error line)
- General control (e.g. a sensor that detects completed motion of an air cylinder, or a tool length sensor).
- Feed Hold (e.g. a door switch on a safety enclosure causes a feed hold to occur)
- Loading a G-Code file
- Starting a G-Code file

Please see the Signal Generator Hardware Guide for wiring details, including the pin layout of the input line connector and sample circuit diagrams.

The Input Lines command on the Configuration menu displays the following panel:

Input Lines Display Shortcut

Line #	Function	Location	Description	Wiring	Sensing Debounce (inches)
1	Unused	N/A		N.C.	0.0040
2	Unused	N/A		N.C.	0.0040
3	Unused	N/A		N.C.	0.0040
4	Unused	N/A		N.C.	0.0040
5	Unused	N/A		N.C.	0.0040
6	Unused	N/A		N.C.	0.0040
7	Unused	N/A		N.C.	0.0040
8	Unused	N/A		N.C.	0.0040
9	Unused	N/A		N.C.	0.0040
10	Unused	N/A		N.C.	0.0040
11	Unused	N/A		N.C.	0.0040
12	Unused	N/A		N.C.	0.0040
13	Unused	N/A		N.C.	0.0040
14	Unused	N/A		N.C.	0.0040
15	Unused	N/A		N.C.	0.0040
16	Unused	N/A		N.C.	0.0040

Debounce Factor:

Debounce During Motion: (msec)

Debounce When Idle: (sec)

Set All Wiring:

Line # – Line number on the Signal Generator’s Input connector.

Function – The function of the switch as follows:

- Home/Limit – The switch will be used for homing and limiting travel.
- Home – The switch will be used for homing only.
- Limit – The switch will be used for limiting travel only.
- Safety – The switch will stop motion immediately (e.g. a servo error line.) The switch behaves the same as a Limit switch, but does not have a Location associated with it.
- Control – The switch will be used for general control purposes (e.g. for tool length sensing or program zero sensing).
- Feed Hold – The switch will cause a feed hold to occur, just as if the Feed Hold button were clicked. (Note that on the 401A Signal Generator, this feature requires the Lightning chip version M202A and higher.)
- Start – Techno will start running the current G-Code file, just as if the Start button were clicked.
- Load G-Code – Techno will load the specified G-Code file.

- Load & Start – Techno will load the specified G-Code file and start running it.
- Unused – The input line is not used and will be ignored by the system.

Note for Limit and Safety Switches:

- When Techno detects a tripped limit or safety switch, it immediately stops all tool movement without ramping. The accuracy of the machine tool position will most likely be lost at that point.
- Techno will not allow any machine movement until you clear the switch. If a limit switch has been tripped, Techno lets you jog away from the switch that was tripped, but disables all other jog options.

Location – Identifies the location (axis and direction) for all Limit and Home switches. For example, choose X- if the switch is located at the negative end of the X axis.

Choose the '+/-' option if the switches at both ends of an axis are wired in series to a single input line.

If your machine uses two motors to drive an axis (defined on the Motor Signals panel), Techno will include the two subaxes in the list of choices. For example, if the X axis is driven by two motors, the choices for the X axis would be:

- X- (1) Negative end of subaxis 1
- X- (2) Negative end of subaxis 2
- X- (All) Negative end of both subaxes
- X+ (1) Positive end of subaxis 1
- X+ (2) Positive end of subaxis 2
- X+ (All) Positive end of both subaxes
- X+/- (1) Positive and negative end of subaxis 1
- X+/- (2) Positive and negative end of subaxis 2
- X+/- (All) Positive and negative end of all subaxes

Description – A brief explanation of the input line, used by Techno in various system messages.

Wiring – Indicates whether the switch is wired normally open or normally closed. Use the Normally Open or Normally Closed button to set the wiring type for all input lines at once.

Debounce Distance – (Model 501A Signal Generator only) For any operation that detects position (homing, tool length sensing, program zero sensing) this is the

distance the machine must travel with the switch continuously tripped for detection to occur.

Debounce Factor – (Model 401A Signal Generator only) The higher the number, the longer the debounce time for all input lines. A typical setting is 8. If Techno is falsely reporting your switches as tripped (due to vibration or other causes), try raising the value.

Debounce During Motion – (Model 501A Signal Generator only) During any motion, Techno constantly monitors all Limit and Safety switches. The Debounce Time is the time a switch must be continuously tripped for Techno to halt motion and report a safety or limit error. If Techno is falsely reporting your switches as tripped (due to vibration or other causes), try raising the value.

Debounce When Idle – When Techno is idle and monitoring input lines, this is the time a switch must be continuously tripped for Techno to recognize it as tripped. Applications for this include:

- ‘Start’ and ‘Load G-Code’ input lines
- M100/M101 command
- ‘Execute when Idle’ features on the M-Code Execution panel

Output Line Settings

Techno controls output lines to activate external devices such as a spindle or coolant pump. You can control the output lines with the Auxiliary Control Panel, the Digital Output Box, the Output Line Control command on the Controller menu, or with user-defined M-Codes.

Please see the Signal Generator Hardware Guide for wiring details, including the pin layout of the output line connector and sample circuit diagrams.

The Output Lines command on the Configuration menu displays the following panel:

Output Lines Display Shortcut

Line #	Description	Button Text (replaces Description)	Display Warning	Display in Aux Panel	Display in Output Box
1	Spindle		Before On	Yes	No
2	Mist Coolant		Before On	Yes	No
3	Flood Coolant		Before On	Yes	No
4			Before On	Yes	No
5			Before On	Yes	No
6			Before On	Yes	No
7			Before On	Yes	No
8			Before On	Yes	No
9			Before On	Yes	No
10			Before On	Yes	No
11			Before On	Yes	No
12			Before On	Yes	No
13			Before On	Yes	No
14			Before On	Yes	No
15			Before On	Yes	No
16			Before On	Yes	No
17			Before On	Yes	No
18			Before On	Yes	No
19			Before On	Yes	No
20			Before On	Yes	No

Safety Interlocks (Invalid Output Line Combinations)

Output Line #	Set To	Output Line #	Set To
None	On	None	On
None	On	None	On
None	On	None	On
None	On	None	On

Output Lines Table

Line # – Line number on the Signal Generator's Output connector.

Description – A brief explanation of the output line. Techno will display an on/off toggle button on the Auxiliary Control Panel for each output line with a Description entered.

Button Text – (Optional) Text that will be used for any output line buttons if the Description is not suitable.

Display Warning – Indicates whether or not Techno will display a warning when the specified output line is turned on.

Display in Aux Panel – Indicates whether or not a button to control the output line will be displayed in the Auxiliary Control Panel on the main screen.

Display in Output Box – Indicates whether or not a button to control the output line will be displayed in the Digital Output Box on the main screen.

Safety Interlocks

The Safety Interlocks table lets you define incompatible combinations of output line on/off states. Techno will prevent the output lines from reaching a state that violates any of the safety interlocks you define.

To create an interlock, set the four pull-down menus across a row of the table to define the incompatible states of two output lines.

Example: The machine tool has an automatic tool changer that uses a spindle with a pneumatic tool chuck, and the chuck should never be open when the spindle is turning. Assuming the spindle is on when output line 1 is on, and the chuck is open when output line 2 is on, the four pull-down menus on the first line of the table would be set as follows:

Output Line #: 1

Set To: On

Output Line #: 2

Set To: On

Variable Output Settings

Techno provides analog, PWM (Pulse Width Modulation) and PFM (Pulse Frequency Modulation) output lines to control external devices such as spindles and lasers. Up to two ports are available, depending on your hardware configuration. Please see the Signal Generator Hardware Guide for details.

The Variable Output command on the Configuration menu displays the following panel:

The screenshot shows the 'Variable Output' configuration window. It has a title bar with 'Variable Output' and a 'Display Shortcut' checkbox. The window is divided into two main sections: Port 1 and Port 2.

Port 1:

- Default Port for S Command
- Type: Analog (0-10V)
- Output Voltage Range: Minimum 0.00 to Maximum 10.00 (volts)
- Spindle RPM Range: 0 to 3000
- Main Screen: Show Controls on Main Screen
- Title: Spindle RPM (above slider)
- Pull-Down Menu Choices: 3000, 2800, 2600, 2400, 2200, 2000, 1800, ...
- Buttons: Add, Remove, Set to Default

Port 2:

- Default Port for S Command
- Type: Analog (0-10V)
- Output Voltage Range: Minimum 0.00 to Maximum 10.00 (volts)
- Range: 0 to 3000
- Main Screen: Show Controls on Main Screen
- Title: (empty) (above slider)
- Pull-Down Menu Choices: 3000, 2800, 2600, 2400, 2200, 2000, 1800, ...
- Buttons: Add, Remove, Set to Default

Port 1

Port 1 is always an analog output using an industry-standard 0-10V signal.

Default Port for S Command – Indicates that any G-Code ‘S’ command will be applied to the port.

Output Voltage Range – The minimum and maximum voltages for the analog output line. The typical setting is 0-10V.

Spindle RPM Range – The spindle speeds (or laser power levels, etc.) that correspond to the minimum and maximum output voltages.

Techno uses these settings to calculate an output voltage that’s proportional to the desired spindle speed (or laser power, etc.)

Example:

Output Voltage Range is configured as 0 – 10V

Spindle RPM Range is configured as 0 – 30000 rpm

To set the spindle speed to 3000 rpm, Techno sets the analog output line to 1.0V using the following math:

$$((3000 \text{ rpm} - 0 \text{ rpm}) / (30000 \text{ rpm} - 0 \text{ rpm})) \times (10\text{V} - 0\text{V}) + 0\text{V} = 1.0\text{V}$$

Example:

Output Voltage Range is configured as 5 – 10V

Spindle RPM Range is configured as 1000 – 10000 rpm

To set the spindle speed to 5000 rpm, Techno sets the analog output line to 7.2V using the following math:

$$((5000 \text{ rpm} - 1000 \text{ rpm}) / (10000 \text{ rpm} - 1000 \text{ rpm})) \times (10\text{V} - 5\text{V}) + 5\text{V} = 7.2\text{V}$$

Main Screen

Show Controls on Main Screen – Tells Techno whether or not to display a slider and pull-down menu for controlling port 1 on the main screen.

Title – The title displayed above the slider on the main screen.

Pull-Down Menu Choices – The list of options included in the pull-down menu on the main screen.

Add – Adds an item to the list of choices.

Remove – Removes an item from the list of choices.

Set List to Default – Resets the list to default choices based on the current Spindle RPM Range.

Port 2

The settings for port 2 are the same as port 1 with the following exceptions.

Type – Indicates the type of signal for port 2: Analog, PWM or PFM.

- Analog

All remaining settings are the same as described for port 1 above.

- PWM

The following settings appear:

Duty Cycle Range – The minimum and maximum duty cycles (0 - 100%).

Range – The spindle speeds (or laser power levels, etc.) that correspond to the minimum and maximum duty cycles.

Frequency – Frequency of the PWM signal (hz).

- PFM

The following settings appear:

Frequency Range – The minimum and maximum frequencies (hz).

Range – The spindle speeds (or laser power levels, etc.) that correspond to the minimum and maximum frequencies.

Duty Cycle – Duty cycle of the PFM signal (0 - 100%).

M-Code Definitions Settings

Techno lets you define custom M-Codes to control output lines or to execute macros.

The M-Code Definitions command on the Configuration menu displays the following panel:

M-Code Definitions
 Display Shortcut

M-Codes to Control Output Lines (For Line Columns: '1' = Turn On '0' = Turn Off '-' = No Action)

General			Action									
M Code	Description	Program Line First Action	M-Code First Action	Delay (sec)	Ln. 1	Ln. 2	Ln. 3	Ln. 4	Ln. 5	Ln. 6	Ln. 7	Ln. 8
3	Spindle On	M-Cd	Set Ln	1.0	-	-	-	-	-	-	-	-
5	Spindle Off	Move	Set Ln	0.0	-	-	-	-	-	-	-	-
		Move	Set Ln	0.0	-	-	-	-	-	-	-	-
		Move	Set Ln	0.0	-	-	-	-	-	-	-	-
		Move	Set Ln	0.0	-	-	-	-	-	-	-	-
		Move	Set Ln	0.0	-	-	-	-	-	-	-	-
		Move	Set Ln	0.0	-	-	-	-	-	-	-	-
		Move	Set Ln	0.0	-	-	-	-	-	-	-	-
		Move	Set Ln	0.0	-	-	-	-	-	-	-	-
		Move	Set Ln	0.0	-	-	-	-	-	-	-	-
		Move	Set Ln	0.0	-	-	-	-	-	-	-	-

M-Codes to Execute Macros

M Code	Description	Macro	Program Line First Action	Display in Output Box
			Move	No
			Move	No
			Move	No
			Move	No
			Move	No
			Move	No
			Move	No
			Move	No
			Move	No
			Move	No

M-Codes to Control Output Lines

M-Code – The number for the M-code that you want to define.

Description – A brief description of the action taken when this M-code is executed.

Program Line First Action – Indicates the sequence of actions when there is a machine tool move command on the same program line as the M-Code:

- M-Cd
 - 1) Execute the M-Code
 - 2) Perform the move
- Move
 - 1) Perform the move
 - 2) Execute the M-Code

M-Code First Action – Indicates the sequence of actions triggered by the M-Code as follows:

- Set Ln
 - 1) Set Output Lines
 - 2) Perform the delay (if any)
- Delay
 - 1) Perform the delay (if any)
 - 2) Set Output Lines

Delay –The duration of the delay that will occur when this M-Code is executed. For example, if the spindle motor takes about 3 seconds to reach full speed, this value should be at least 3.

Ln. # – Choose the action for each line upon execution of the M-code. The choices are 1 for On, 0 for Off, or '-' to indicate the state of the line remains unchanged.

M-Codes to Execute Macros

M-Code – The number for the M-code that you want to define.

Description – A brief description of the action taken when this M-code is executed.

Macro – Click in this field to display the down-arrow button. Click the down-arrow button to display the Edit M-Code Macro dialog. In this dialog, enter any number of G-Code lines. Whenever Techno encounters the M-Code in a G-Code program, these commands will be executed.

Program Line first Action – Indicates the sequence of actions when there is a machine tool move command on the same program line as the M-Code:

- M-Cd
 - 1) Execute the M-Code
 - 2) Perform the move
- Move
 - 1) Perform the move
 - 2) Execute the M-Code

Display in Output Box – Indicates whether or not a button to run the macro will be displayed in the Digital Output Box on the main screen.

M-Code Execution Settings

The M-Code Execution command on the Configuration menu displays the following panel:

M-Code Execution
 Display Shortcut

Automatic Execution Mode

None

Basic

Start (After Hold): M

Feed Hold: M

Full

Feedrate Move: M

All Other Times: M

Execute When Idle

On Feed Hold/Safety Tripped: M

On Change of Input Line State

Input Line #	State	Execute M-Code
None	Tripped	<input type="text"/>
None	Tripped	<input type="text"/>
None	Tripped	<input type="text"/>
None	Tripped	<input type="text"/>

Execute After Motion Errors

After Limit/Safety/Spindle Error: M

Execute at Start/End of Motion

Start of Motion: M

End of Motion: M

Execute During Ramping

Beginning of Ramp: M

End of Ramp: M

Execute While Connecting/Disconnecting

Connecting: M

Disconnecting: M

Any M-Code entered on this panel must be defined on the M-Code Definitions panel.

Automatic Execution Mode

Enter an M-Code in the field(s) associated with the selected radio button.

None – Techno doesn't execute any M-Codes automatically.

Basic – This mode is intended for applications where it's convenient to automatically turn off spindle, vacuum and other devices whenever the machine is stopped.

Techno automatically executes an M-Code whenever Feed Hold is clicked, or an error occurs. Techno executes another M-Code whenever Start is clicked after a Feed Hold has occurred (but not when initially starting the file).

Note that you don't need to fill in both M-Code fields. For example, you can specify a Feed Hold M-Code, and leave the Start (After Hold) field blank.

Full – This mode is intended for applications where the controller must turn output lines on during feedrate moves and off during rapid moves. Applications include plasma cutting, laser cutting, waterjet cutting and liquid dispensing.

Techno executes the Feedrate Move M-Code before any feedrate move (G01, G02, G03), and before restarting a feedrate move after a Feed Hold.

Techno executes the All Other Times M-Code:

- before any rapid move (G00)
- whenever Feed Hold is hit during a feedrate move
- at the end of the program if the program ends with a feedrate move
- when an error occurs

Execute When Idle

On Feed Hold/Safety Tripped – Techno executes this M-Code whenever a Feed Hold or Safety input line is tripped. In addition, Techno prevents you from using the Aux panel to set output lines to a state that would conflict with this M-Code.

A typical use for this feature is to turn off the spindle when the operator opens an enclosure door.

On Change of Input Line State – Techno executes the M-Code when the specified input line reaches the specified state.

Execute After Motion Errors

After Limit/Safety/Spindle Error – Techno executes the M-Code after an error occurs. This is useful in applications where it's safest to automatically turn off the spindle, vacuum and other devices whenever the Signal Generator detects an error. Errors include limit switch tripped, safety switch tripped, or spindle stopped during thread cutting (lathe only).

When running a lathe, and especially when cutting threads, this mode is highly recommended. If an error occurs (e.g. a limit switch gets tripped), Techno immediately stops moving the tool. The spindle must then be

stopped, or else there will be interference between the stationary tool and the spinning workpiece. The automatic M-Code should be configured to stop the spindle.

Execute at Start/End of Motion

A typical use for this feature is to turn on a warning light whenever the machine tool is moving.

Start of Motion – Techno executes this M-Code immediately before any motion begins (including starting a G-Code program, jogging, and so on).

End of Motion – Techno executes this M-Code immediately after motion stops.

Note that ‘motion’ includes actual machine moves as well as dwells.

Execute During Ramping

A typical use for this feature is to turn on/off an external device when the machine tool is accelerating or decelerating. Techno turns the device on/off only while running a G-Code file.

Beginning of Ramp – Techno executes this M-Code when ramping begins.

End of Ramp – Techno executes this M-Code when ramping ends.

Execute While Connecting/Disconnecting

A typical use of this feature is to turn on and off a warning light indicating the machine tool is active.

Connecting – Techno executes this M-Code whenever the software is connecting to the Signal Generator.

Disconnecting – Techno executes this M-Code whenever the software is disconnecting from the Signal Generator.

G-Code Settings

Techno lets you customize handling of some G-Code and M-Code commands.

The G-Code command on the Configuration menu displays the following panel:

General

Message on M00/M01 Program Pause – Directs Techno to display a message dialog whenever it encounters an M00 or M01 command in a G-Code program. If you uncheck this option, Techno will still pause processing of the G-Code program on M00, but will not display a message to the operator.

Warn if Offset/Diameter Doesn't Match Current Tool – Tells Techno whether or not to display an error message when a G41, G42, G43 or G44 command specifies an offset or diameter number that does not match the current tool.

Use Fanuc-Compatible G74/G84 Tapping Cycles – Tells Techno to use the Fanuc-style G74 and G84 commands for tapping holes. See “G74, G84 Tapping Canned Cycle Commands” in the System Programming section for details.

M100/M101 Timeout – Sets the timeout for the M100 and M101 commands. See “M100, M101 Wait for Input Line” in the System Programming section for details.

G04 'P' Parameter Units – Sets the units for the G04 command. See “G04 Dwell” in the System Programming section for details.

G73/G83 Retract Distance – Sets the ‘D’ distance for canned drilling cycles. See “G73, G80, G81, G82, G83, G85, G98, G99 Canned Cycles” in the System Programming section for details.

Helical Interpolation Accuracy – Sets the maximum chord error for the short linear moves that compose a helical move. A lower number results in more accurate helixes but slower G-Code file loading.

Arc Radius Tolerance – Sets the maximum allowable difference between the starting and ending radius for an arc move (G02 or G03). While loading files, Techno checks every arc to make sure the arc radius difference is within the tolerance specified here.

G28/G30 Applies to Axes – Sets the axes that are moved by the reference point G-Code commands. Only the selected axes will move when a G28/G30 is executed.

Engraving

Join Tolerance – In the font definition, if two drawing entities are touching end to end, Techno treats them as a single feature to machine without lifting the tool. Due to rounding or drawing error, two entities that are meant to be joined may not actually touch end to end. The engraving command will automatically join any entities whose endpoints are less than the Join Tolerance apart. This setting is particularly important when you use a custom-designed font.

Ellipse / Spline Chord Error – When a character contains an ellipse or spline, Techno generates many short linear moves to closely approximate the original curve. This setting controls how closely the set of lines will match the original curve; it's the maximum deviation between the curve and each generated line. The smaller the number, the more linear moves Techno will create for each ellipse or spline, and the smoother the characters will be cut.

M-Code Executed Before Each Character (Z) – While executing a G141 command, Techno executes this M-Code before engraving each character with the Z axis spindle.

M-Code Executed Before Each Character (W) – While executing a G141 command, Techno executes this M-Code before engraving each character with the W axis spindle.

Optional Commands

Execute G54 – G59 – Directs Techno to execute the G54-G59 commands in a G-Code program. Uncheck this option if your G-Code files include a G54 as part of the boilerplate at the beginning of the file, and you're not using fixture

offsets. If you uncheck this option, make sure any G-Code program you run does not rely on G54-G59 to set Program Zero.

Execute Optional ‘/’ Lines – Directs Techno to execute lines in a G-Code program that begin with a “/” (optional lines). If you uncheck this option, Techno will ignore lines that begin with “/”.

Execute M01 – Directs Techno to execute M01 commands in a G-Code program (optional program pause). If you uncheck this option, Techno will ignore M01 commands.

Scale

X, Y and Z – Sets the default scaling factor for each axis, for all G-Code files loaded. These factors are overridden by any scaling commands included in a G-Code file. For more information on scaling G-Code, see “G50, G51 Scaling” in the System Programming section.

Spindle

Techno uses these settings when processing the G74 and G84 commands.

Turn On Clockwise – M-Code to turn the spindle on clockwise.

Turn On Counterclockwise – M-Code to turn the spindle on counterclockwise.

Turn Off – M-Code to turn the spindle off.

User Variable Settings

User variables are created on this panel and may be used in G-Code programs or macros like any other variable. They are stored in a file called UserVars.dat, not the Setup file, so they are always accessible on your computer no matter what Setup file is loaded.

Display on screen – If you select Yes, Techno will display this variable on the main screen in the User Variables Box.

Editable on Screen – If you select Yes, you will be able to change the value on the main screen.

Counter Settings

Counters let you track the number of times something has occurred, such as the number of times a particular part has been machined. You'll create counters here, and increment them using the G150 command in a G-Code program. Techno will update a counter only if the system is connected to the Signal Generator.

Counters are stored in a file called Counters.dat, not the Setup file, so they are always accessible on your computer no matter what Setup file is loaded.

The Counters command on the Configuration menu displays the following panel:

Counters				
Master List for All Setups			Usage in Current Setup	
#	Description	Value	Display on Screen	Editable on Screen
1	Counter	1	Yes	Yes
2		0	No	Yes
3		0	No	Yes
4		0	No	Yes
5		0	No	Yes
6		0	No	Yes
7		0	No	Yes
8		0	No	Yes
9		0	No	Yes
10		0	No	Yes
11		0	No	Yes
12		0	No	Yes
13		0	No	Yes
14		0	No	Yes
15		0	No	Yes
16		0	No	Yes
17		0	No	Yes
18		0	No	Yes
19		0	No	Yes
20		0	No	Yes

– Counters are identified by this number. The G150 command uses this number to specify the counter. For more information see “G150 Increment Counter” in the System Programming section.

Description – A description of the counter. The description will appear on the main screen if you display the counter there.

Value – Enter the initial value of the counter. This value may be changed by your program or on the main screen.

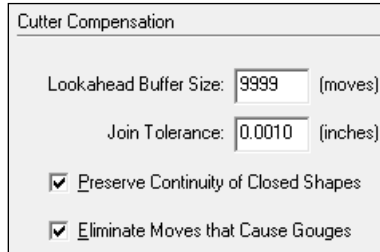
Display on screen – If you select Yes, Techno will display this counter on the main screen in the User Variables Box.

Editable on Screen – If you select Yes, you will be able to change the value on the main screen.

Cutter Compensation Settings

Techno has cutter compensation available to offset the toolpath by the tool radius. There are several ways to configure this feature.

The Cutter Compensation command on the Configuration menu displays the following panel:



The screenshot shows a dialog box titled "Cutter Compensation". It contains two input fields: "Lookahead Buffer Size" with a value of "9999" and units "(moves)", and "Join Tolerance" with a value of "0.0010" and units "(inches)". Below these fields are two checked checkboxes: "Preserve Continuity of Closed Shapes" and "Eliminate Moves that Cause Gouges".

Lookahead Buffer Size – Sets the number of moves Techno looks forward when calculating the offset toolpath. The further it looks, the more potential gouges Techno can eliminate. See “G40, G41, G42 Cutter Compensation” in the System Programming section for more details.

Join Tolerance – Sets the maximum distance between two points in order for those points to be considered equal. The tolerance gives the software some flexibility when looking for intersections and tangent points between lines and arcs. See “G40, G41, G42 Cutter Compensation” in the System Programming section for more details.

Preserve Continuity of Closed Shapes – When checked, Techno uses advanced logic to automatically close the offset toolpath when processing a closed shape (e.g. a square). See “G40, G41, G42 Cutter Compensation” in the System Programming section for more details.

Eliminate Moves that Cause Gouges – When checked, Techno uses advanced logic to prevent gouges, by eliminating the lines and arcs that cause them. See “G40, G41, G42 Cutter Compensation” in the System Programming section for more details.

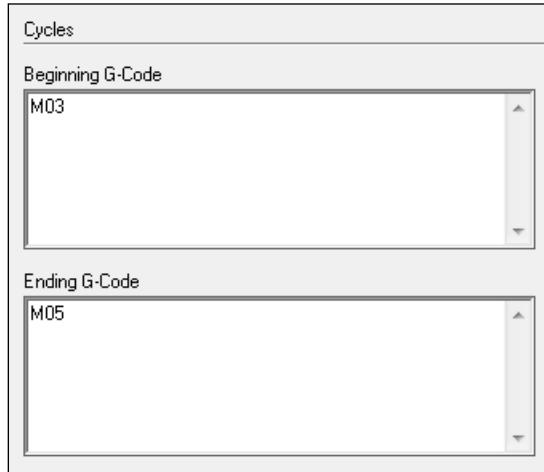
Import Settings

These settings are described under “Importing a DXF File” in the System Programming section.

Cycle Settings

The Cycles Control Panel on the main screen lets you generate G-Code for standard operations such as hole drilling, pocketing, and so on. There are several configuration settings that affect generation of the G-Code.

The Cycles command on the Configuration menu displays the following panel:



The screenshot shows a window titled "Cycles" with two text input fields. The first field is labeled "Beginning G-Code" and contains the text "M03". The second field is labeled "Ending G-Code" and contains the text "M05". Both fields have vertical scroll bars on their right sides.

Beginning G-Code – Techno adds this G-Code to the beginning of every cycle.

Ending G-Code – Techno adds this G-Code to the end of every cycle.

Tool Library Settings

Each tooling file contains settings for up to 100 tools. Each tool has an associated tool number, description, diameter and XYZ offset. The diameter is used by the cutter compensation commands (G41 and G42). The XYZ offset is used by the tool length/geometry compensation commands (G43 and G44).

The Tool Library command on the Configuration menu displays the following panel:

Tool #	Description	Diameter (inches)	X Offset (inches)	Y Offset (inches)	Z Offset (inches)
1		0.0000	0.0000	0.0000	0.0000
2		0.0000	0.0000	0.0000	0.0000
3		0.0000	0.0000	0.0000	0.0000
4		0.0000	0.0000	0.0000	0.0000
5		0.0000	0.0000	0.0000	0.0000
6		0.0000	0.0000	0.0000	0.0000
7		0.0000	0.0000	0.0000	0.0000
8		0.0000	0.0000	0.0000	0.0000
9		0.0000	0.0000	0.0000	0.0000
10		0.0000	0.0000	0.0000	0.0000
11		0.0000	0.0000	0.0000	0.0000
12		0.0000	0.0000	0.0000	0.0000
13		0.0000	0.0000	0.0000	0.0000
14		0.0000	0.0000	0.0000	0.0000
15		0.0000	0.0000	0.0000	0.0000
16		0.0000	0.0000	0.0000	0.0000
17		0.0000	0.0000	0.0000	0.0000
18		0.0000	0.0000	0.0000	0.0000
19		0.0000	0.0000	0.0000	0.0000
20		0.0000	0.0000	0.0000	0.0000
21		0.0000	0.0000	0.0000	0.0000
22		0.0000	0.0000	0.0000	0.0000
23		0.0000	0.0000	0.0000	0.0000
24		0.0000	0.0000	0.0000	0.0000
25		0.0000	0.0000	0.0000	0.0000
26		0.0000	0.0000	0.0000	0.0000
27		0.0000	0.0000	0.0000	0.0000
28		0.0000	0.0000	0.0000	0.0000
29		0.0000	0.0000	0.0000	0.0000
30		0.0000	0.0000	0.0000	0.0000

Tool # – Used throughout the software to identify the tool (e.g. in the M06 tool change command).

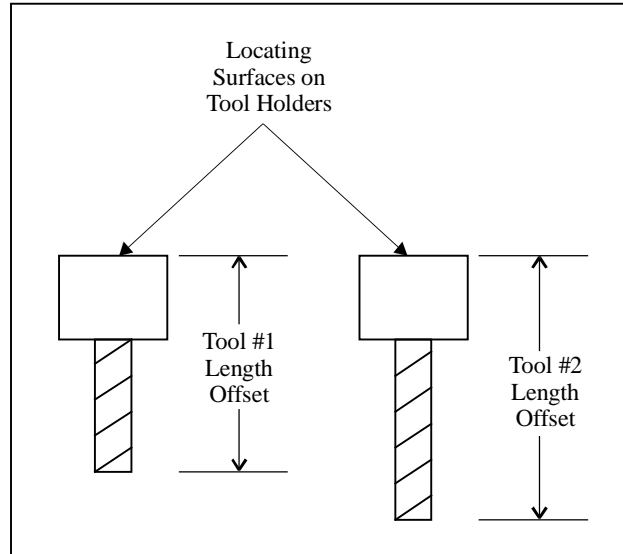
Description – Used throughout the software to describe the tool.

Diameter – The diameter of the tool, if applicable. The diameter is used by the G41 and G42 commands, and by the viewports to graphically represent the tool diameter. This field is not used for lathe applications.

X, Y, Z Offsets – The offset used by the G43 and G44 commands.

To set these values, make sure each tool is mounted in its own tool holder. Then enter the offset from the tip of the tool to the tool holder's locating feature as follows:

- **Milling Applications** – The most commonly used offset is Z, for tool length compensation. To use tool length compensation, each tool holder must have a feature that locates it relative to the spindle nose. Enter the axial distance between the tip of the tool and the tool holder's locating feature.



If you're using tool length sensing, Techno will automatically set the Z offset for you.

- Lathe Applications – Offsets are generally required for both X and Z. To use tool geometry compensation, each tool holder must have a feature that locates it relative to the tool post. Enter the X and Z distances between the tip of the tool and the tool holder's locating feature.

Tool Life Management Settings

Techno has a Tool Life Management system that can monitor and manage tool wear. The system offers both time-based and counter-based monitoring, which makes it useful in a variety of CNC applications. Please see "Tool Box" in the Main Screen Features section for more information on the Tool Life Management system.

The Tool Life Management command on the Configuration menu displays the following panel:

Tool Life Management <input type="checkbox"/> Display Shortcut								
Tool		Track Time			Track Counter			
Tool #	Description	Active	Elapsed Cutting Time (hrs)	Life (hrs)	Active	Description	Curr Count	Life
1	1/8 milling	Yes	0.000	20.000	No		0	0
2		No	0.000	0.000	No		0	0
3		No	0.000	0.000	No		0	0
4		No	0.000	0.000	No		0	0
5		No	0.000	0.000	No		0	0
6		No	0.000	0.000	No		0	0
7		No	0.000	0.000	No		0	0
8		No	0.000	0.000	No		0	0
9		No	0.000	0.000	No		0	0
10		No	0.000	0.000	No		0	0
11		No	0.000	0.000	No		0	0
12		No	0.000	0.000	No		0	0
13		No	0.000	0.000	No		0	0
14		No	0.000	0.000	No		0	0
15		No	0.000	0.000	No		0	0
16		No	0.000	0.000	No		0	0
17		No	0.000	0.000	No		0	0
18		No	0.000	0.000	No		0	0
19		No	0.000	0.000	No		0	0
20		No	0.000	0.000	No		0	0
21		No	0.000	0.000	No		0	0
22		No	0.000	0.000	No		0	0
23		No	0.000	0.000	No		0	0
24		No	0.000	0.000	No		0	0
25		No	0.000	0.000	No		0	0
26		No	0.000	0.000	No		0	0
27		No	0.000	0.000	No		0	0
28		No	0.000	0.000	No		0	0
29		No	0.000	0.000	No		0	0
..	

Tool

Tool # - The tool number as shown on the Tool Library panel.

Description – The description entered in the Tool Library panel (not editable here).

Track Time

Active – If you select Yes, Techno will track the amount of time the tool has been used and alert the operator when a program will exceed the life of the tool.

Elapsed Cutting Time (hrs) – The total elapsed cutting time for the tool. Generally Techno will update this value for you as appropriate. Techno automatically adds time when the tool is used, and resets the value to zero when you change the tool. However, you may edit this value if necessary. For example, when first implementing Tool Life Management, you may want to set an initial value for the tools currently in use.

Life (hrs) – The life of the tool in hours. When you run a G-Code program that will cause the tool to exceed its life, Techno displays a warning message.

Track Counter

Active – If you select Yes, Techno will track the number of times an event occurs while the tool is loaded. The event may be anything you want, for instance, stamping a part on a CNC punch press. The event must be marked in the G-Code program by a G160 command. Every time a G160 is performed in the program, the counter for the current tool will increase by one.

Description – A description of the process that is to be monitored and counted by Techno. For example, on CNC punch press, the Description could be “Punch Count”.

Curr Count – The current count for the tool. Generally Techno will update this value for you as appropriate. Techno increments the counter when it executes the G160 command, and resets the value to zero when you change the tool. However, you may edit this value if necessary. For example, when first implementing Tool Life Management, you may want to set an initial value for the tools currently in use.

Life – The life of the tool measured in counts. When you run a G-Code program that will cause the tool to exceed its life, Techno displays a warning message.

Tool Change Settings

Techno provides a great deal of flexibility in how it handles a tool change. It can accommodate any manual or automatic tool change scheme.

The Tool Change command on the Configuration menu displays the following panel:

Display Shortcut

Tool Change Mode

Silent (No Pause)
 Manual (Pause With Message)
 Automatic Tool Changer (M06 Macro)

Type: Fixed Rack

Sense Tool Length after Tool Change

Automatic Tool Change

Approach Feedrate: 5.00 (in/min) Monitor Tool Chuck Power

Input Line: 10

Power On State: Normal

Tool Rack Positions (Machine Coordinates, inches)

Pos #	X	Y	Z Raised	Z Near Grip	Z Near Release	Z Grip	Z Release
1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
3	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
7	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
8	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
9	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
10	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
11	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
12	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Tool Change Mode

Silent (No Pause) – The tool change occurs in the Techno software, but the G-Code processing does not stop and no physical tool change occurs. This mode is useful if you’re using the same tool for a long time, or if you’re running a program offline.

Manual (Pause with Message) – Techno pauses G-Code processing and displays a message to the operator to change tools. Techno will also execute M06 if you’ve defined one in the M-Codes to Control Output Lines table, on the M-Code Definitions panel (described later in this section). A typical use is to turn off the spindle.

Automatic Tool Changer (M06 Macro) – Techno executes the macro defined for M06 on the M-Code Definitions panel (described later in this section). The macro contains the logic required to drive an automatic tool changer. Using the advanced programming features (described in “System Programming”), you can create a macro to control any automatic tool changer.

Also, you can use the macro to perform a customized sequence of operations for a manual tool change.

Type – If you selected Automatic Tool Changer, choose the type:

- **Fixed Rack** – A stationary rack of tools is located somewhere in the machine envelope. The spindle is equipped with an automatic tool chuck. To pick up a tool, the machine moves to a rack position, closes the tool chuck and raises the Z axis. This is the most common automatic tool change method used on smaller machine tools. When you select this option, Techno automatically creates a standard M06 macro for you. You cannot edit the macro unless you switch to Custom Macro mode (below).
- **Custom Macro** – You’ll write a custom M06 macro to perform a tool change on other types of tool changers, or a customized manual tool change sequence. Also, you’ll use this mode if you want to start with the Fixed Rack macro, and tailor it for your tool changer.

If you chose Custom Macro, go to the M-Code Definitions panel and edit the existing M06 macro (or create a new one if one hasn’t been defined). See “M-Code Definitions” below for more details.

Any custom M06 macro must assign a value to a Techno system variable “#CurrTool”. The typical G-Code line for this is

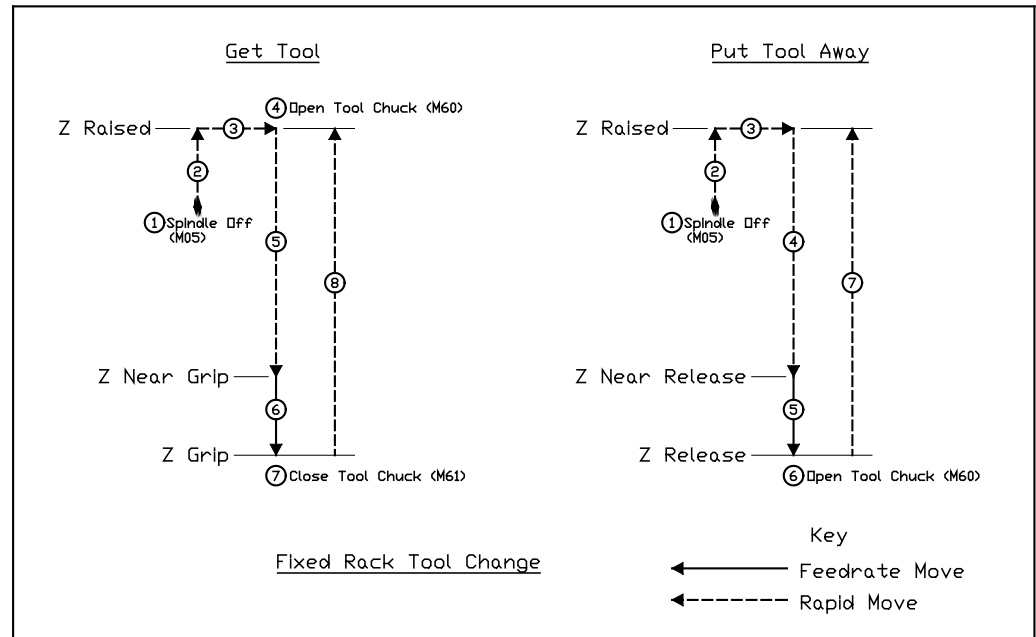
```
#CurrTool = #NextTool
```

See “Advanced Programming Features” in the System Programming section to learn more about system variables and other advanced commands useful for controlling tool changers. You can skip the rest of this section, which describes the Fixed Rack tool change in detail.

Sense Tool Length after Tool Change – Tells Techno to sense the tool length immediately after each tool change. This applies to manual or automatic tool changes. For more information see “Tool Length Sensing” later in this section.

Automatic Tool Change

The diagram below summarizes the sequence for the Fixed Rack tool change.



Tool Rack Positions

Set the following parameters for each tool rack position (in machine coordinates), referring to the diagram above.

X, Y – The machine coordinates of the tool rack position (destination for move 3 in the diagram).

Z Raised – Z machine coordinate that's high enough to allow the longest tool to clear the tool rack and any workpiece or fixture when the machine moves to or from the tool rack position. **WARNING: IT IS VERY IMPORTANT TO SET THIS VALUE HIGH ENOUGH TO ACCOMMODATE THE LONGEST TOOL. OTHERWISE A CRASH WILL OCCUR.** The safest value to use is zero (Z axis completely raised).

Z Near Grip – When getting a tool, the Z machine coordinate to which the machine moves at the rapid rate. **WARNING: IT IS VERY IMPORTANT TO SET THIS VALUE HIGH ENOUGH TO ACCOMMODATE THE LONGEST TOOL. OTHERWISE A CRASH WILL OCCUR.** The safest value to use is zero (Z axis completely raised).

Z Grip – Z machine coordinate where the tool chuck closes to grip the tool.

Z Near Release – When putting away a tool, the Z machine coordinate to which the machine moves at the rapid rate. **WARNING: IT IS VERY IMPORTANT TO SET THIS VALUE HIGH ENOUGH TO**

ACCOMMODATE THE LONGEST TOOL. OTHERWISE A CRASH WILL OCCUR. The safest value to use is zero (Z axis completely raised).

Z Release – Z machine coordinate where the tool chuck opens to release the tool.

Approach Feedrate – The feedrate for all downward moves to the Z Grip and Z Release level.

Monitor Tool Chuck Power – Tells Techno to monitor an input line that indicates the presence of tool chuck power. Typically this is used to monitor air pressure for the spindle chuck. If the input line indicates no power, Techno halts motion and displays an error message. This prevents a crash when the system attempts to change tools.

Input Line – The input line monitored for tool chuck power.

Power On State – The state of the input line when the tool chuck has power.

- ▶ To Finish Setting Up Automatic Tool Changing
 1. On the Output Lines panel, enter the description “Tool Chuck” for the tool chuck output line. Also, create a Safety Interlock so the tool chuck cannot be open when the spindle is on (see Output Lines above for more details).
 2. On the M-Code Definitions panel, define three M-codes to set output lines as follows (see “M-Code Definitions” below for more details):
 - M05: Spindle Off
 - M60: Open Tool Chuck (be sure to include a delay time, and set the delay to occur AFTER the output lines are set)
 - M61: Close Tool Chuck (be sure to include a delay time, and set the delay to occur AFTER the output lines are set)
 3. On the Tools in Changer panel, identify the tool in each rack position. Whenever you change the tools in the rack, you’ll need to update this panel. See “Tools in Changer Settings” below for more details.

Tools in Changer Settings

The Tools in Changer command on the Configuration menu displays the following panel:

Pos #	Tool
1	None
2	None
3	None
4	None
5	None
6	None
7	None
8	None
9	None
10	None
11	None
12	None
13	None
14	None
15	None
16	None
17	None
18	None
19	None
20	None
21	None
22	None
23	None
24	None
25	None
26	None
27	None
28	None
29	None
30	None

This panel lets you specify which tool is in each position of an automatic tool changer. The tool arrangement is stored in the Tooling file (“TLG” extension).

Changer Positions

Pos # – The tool changer position, which corresponds to the same column in the Tool Rack Positions table on the Tool Change panel.

Tool – The tool loaded in the tool changer position. Use the pull-down menu to select any tool in the tool library (or None if there is no tool loaded.)

Restrict System to Use Only These Tools – Tells Techno to list only the tools in the changer in the Tool pull-down menu on the main screen.

Tool Length Sensing Settings

Techno supports tool length sensing to automatically set the Z offset for any tool (displayed on the Tool Library panel of the Configuration dialog box). It works for milling-type machine configurations only.

The Tool Length Sensing command on the Configuration menu displays the following panel:

Use Tool Length Sensing – Check to activate tool length sensing.

The sequence of operations for tool length sensing is listed below. All parameters on this panel are underlined. Note that all X, Y and Z positions must be specified in *machine* coordinates.

- The Initial M-Code is executed (optional, typically used to turn off the spindle)
- Z axis moves up to the Raised Position (rapid move)
- X and Y axes move to the Sensor Position (rapid move)
- Z axis moves down to the Sensing Start Position (rapid move)
- Z axis moves in the Sensing Direction towards the sensor, moving at the Sensing Feedrate, until the sensor trips or the Sensing Limit is reached. The sensor must be wired to the input line specified by Monitor Input Line.
- If the Sensing Method is set to Sense on Retract, the Z axis retracts from the sensor until it's not tripped.
- The Z axis offset for the current tool is automatically entered into the Tool Library panel of the Configuration dialog box, according to the Tool Length Formula. The default formula is simply the current Z machine coordinate (“#Machine.Z”). To change the formula, see “Advanced Programming Features” in the System Programming section for details on using variables and creating mathematical expressions.

- Z axis moves up to the Raised Position (rapid move).

Program Zero Sensing Settings

You can use program zero sensing to detect program zero for any axis. For information on using this feature, see “DRO Box” in the Main Screen Features section of this manual.

The Program Zero Sensing command on the Configuration menu displays the following panel:

Program Zero Sensing

Use Program Zero Sensing

Activate for Axes: X Y Z A B C

Sensing Direction:

Sensing Feedrate: (in/min)

Monitor Input Line:

Max Distance to Move: (inches)

Retract Distance: (inches)

Sensor Offset: (inches)

Initial Rapid to Sensor Location:
(Optional, Machine Coords)

X (in)

Y (in)

Z (in)

A (deg)

B (deg)

C (deg)

Use Program Zero Sensing – Activates this feature.

Activate for Axes – For each axis you select, Techno adds a corresponding menu command to the program coordinate Set button in the DRO Box.

Sensing Direction – The direction of the move towards the sensor.

Sensing Feedrate – The feedrate for the move towards the sensor.

Monitor Input Line – The input line for the program zero sensor.

Max Distance to Move – The maximum distance the axis will move to find the sensor.

Retract Distance – The distance the axis retracts after tripping the sensor.

Sensor Offset – The thickness of the sensor.

Initial Rapid to Sensor Location – Select axes to include in an optional rapid move to the sensor location. Values are in machine coordinates.

Touch Probe Settings

The Touch Probe command on the Configuration menu displays the following panel:

Touch Probe Display Shortcut

Use Touch Probe

Sensing Feedrate: 3.00 (in/min) Sensing Axis: Z

Monitor Input Line: 1 Sensing Direction: Negative

Retract Distance: 0.0200 (inches) Positioning Axes: X

Probe Tip Diameter: 0.0787 (inches) Y

Point File Type: XYZ Scan Pattern: Zig Zag

Point File Delimiter: Space Reversal Distance: 0.0200 (inches)

Create Output Files:

- SCN (Raw scan data, FlashCut CNC format)
- XYZ (Points)
- XYZ (Points, compensated for probe tip diameter)
- STL (Surface)
- STL (Surface, compensated for probe tip diameter)

Use Touch Probe – Check to activate the touch probe feature.

Sensing Feedrate – The feedrate for surface-sensing moves.

Monitor Input Line – The input line for the touch probe.

Retract Distance – The distance the Sensing Axis retracts after detecting a surface.

Probe Tip Diameter – The diameter of the touch probe tip.

Point File Type – The extension of the output file(s) created at the end of the scan that contain all of the scanned points.

Point File Delimiter – The character that separates the each coordinate in a point file.

Create Output Files – Techno always creates an SCN file to store the raw scan data. Techno will also generate up to four additional files at the end of each scan, based on your selections here.

Sensing Axis – The axis that moves when detecting a surface.

Sensing Direction – The direction of motion when detecting a surface.

Positioning Axes – The two axes that move the probe between sensing locations. Together they form the plane perpendicular to the Sensing Axis.

Scan Pattern – Select either the Zig Zag or Single Direction pattern for sensing. For more information on the two scan patterns, see “Probe Control Panel” in the Main Screen Features section.

Reversal Distance – This setting only applies to single direction scanning. See “Probe Control Panel” in the Main Screen Features section for information on this setting.

Main Screen Settings

The Main Screen command on the Configuration menu displays the following panel:

Main Screen

Show DRO's

- Program Coordinates
- Machine Coordinates
- Relative Coordinates
- Distance To Go

Show Boxes

- Tool Box
- Teach Box
- Fiducial Box

Show Control Panels

- G-Code
- Jog
- Point
- Home
- Aux
- MDI
- Probe
- Cycles

Select All

Deselect All

Show DRO's

Select which DRO's are visible on the main screen. The DRO command on the View menu let's you toggle between showing one or all of the DRO's checked here.

Show Control Panels

Select which control panels are visible on the main screen.

Show Boxes

Select which boxes are visible on the main screen. Note that visibility of the User Variable, Digital Output and Variable Output Boxes is controlled by other areas of the configuration.

Viewport Settings

There are 11 different viewports available in Techno. You can display as many as you like, and set parameters for each one.

The Viewports command on the Configuration menu displays the following panel:

View On	Automatic Zoom Extents	Tool Icon	Tool Trace Width
<input type="checkbox"/> X	Toolpath	N/A	N/A
<input type="checkbox"/> Y	Toolpath	N/A	N/A
<input checked="" type="checkbox"/> Z	Toolpath	N/A	N/A
<input checked="" type="checkbox"/> XY	Toolpath	Tool Diameter	Match Icon
<input checked="" type="checkbox"/> XZ	Toolpath	Dot	Match Icon
<input type="checkbox"/> YX	Toolpath	Tool Diameter	Match Icon
<input type="checkbox"/> YZ	Toolpath	Dot	Match Icon
<input type="checkbox"/> ZX	Toolpath	Dot	Match Icon
<input type="checkbox"/> ZY	Toolpath	Dot	Match Icon
<input checked="" type="checkbox"/> XYZ	Toolpath	N/A	Match Icon
<input type="checkbox"/> XYZA	Toolpath	N/A	Match Icon

General
<input type="checkbox"/> Activate Scrolling, Zooming and Rotation
<input checked="" type="checkbox"/> Use Video RAM to Optimize Performance
<input checked="" type="checkbox"/> Draw Tool Trace
<input type="checkbox"/> Use Finer Grid Spacing

View On – Turns on the viewport.

Automatic Zoom Extents – Sets the scaling mode for the viewport. Toolpath causes the toolpath for the current G-Code File to expand as much as possible within the viewport. Machine causes the machine envelope to expand as much as possible within the viewport.

Tool Icon – Allows you to select the size of the Tool Icon.

- Dot – The tool is represented by a small dot.
- Tool Diameter – The tool is represented by circle with the same diameter as the current tool.

Tool Trace Width – Determines how the viewport's blue trace is drawn (applies to viewports with 2 or more axes).

- **Thin Line** – The trace is drawn just thick enough to cover the red toolpath preview lines.
- **Match Icon** – The trace is drawn as wide as the tool icon. If the tool icon is set to Tool Diameter, the blue trace shows an accurate representation of what has been cut.

General

Activate Scrolling, Zooming and Rotation – Checking this box causes Techno to replace the standard (original) viewports with advanced viewports on the main screen. The advanced viewports offer the following:

- Scrolling
- Zooming
- Rotation
- Integration with the Jump to Line feature (Techno jumps to the line or arc you select in the viewport)

If you generally use small G-Code files (under 500 KB), you can use the advanced viewports regardless of your PC's graphics hardware. If you use larger G-Code files (larger than 500 KB), you should only use the advanced viewports if you have a high-performance graphics card in your PC. If your PC's graphics capabilities are limited, inexpensive graphics cards are readily available and easy to add to a desktop PC.

Use Video RAM to Optimize Performance – When checked, Techno improves viewport performance by storing graphics information directly on your computer's video hardware.

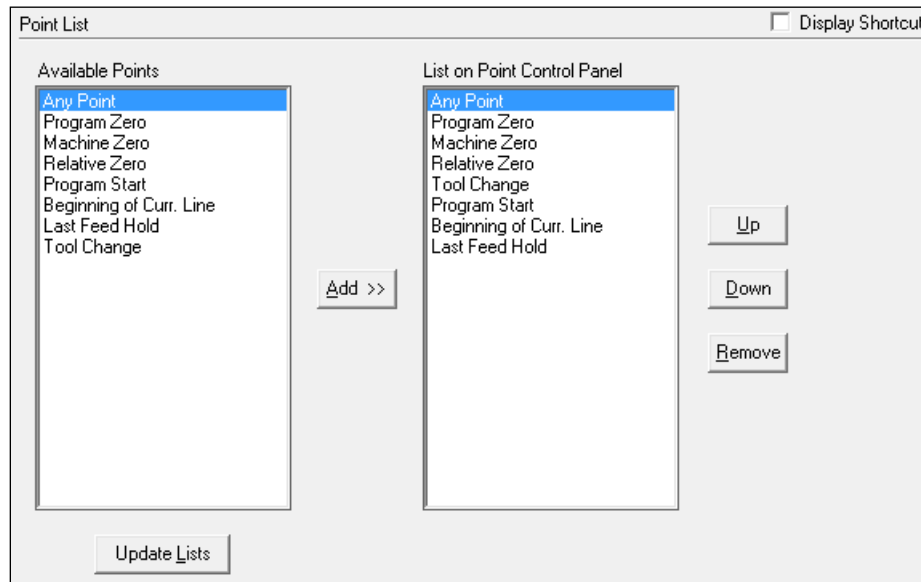
Draw Tool Trace – When checked, the tool icon creates a blue tool trace as it moves.

Use Finer Grid Spacing – When checked, Techno will use finer grid spacing at all zoom levels.

Point List Settings

You can customize the list of points displayed in the Point List on the Point Control Panel.

The Point List command on the Configuration menu displays the following panel:



Available Points – The points available to add to the List on the Point Control Panel:

- System Points: Any Point, Program Zero, Machine Zero, Relative Zero, Program Start, Current Line Start, Last Feed Hold
- Reference Points: Any reference point that has a description entered (on the Reference Points panel)
- Tool Rack Positions: Any tool rack position with a value other than zero entered in any column (on the Tool Change panel)

List on Point Control Panel – The points that are displayed on the Point Control Panel.

Add – Adds the selected Available Point to the list.

Up, Down – Moves the selected item in the list by one position.

Remove – Removes the item from the list.

Update Lists – Updates the Available Points list based on changes you’ve made on other configuration panels, and updates the names in the List on Point Control Panel.

After you exit the Configuration dialog, choose the Point button on the main screen to see your changes in the Point List. For more information, see “Point Control Panel” in the Main Screen Features section.

Jogging Settings

The Jogging command on the Configuration menu displays the following panel:

Jogging
 Display Shortcut

Jog Rates

	Slow	Medium	Fast	
X	10.00	50.00	200.00	(in/min)
Y	10.00	50.00	200.00	(in/min)
Z	5.00	25.00	100.00	(in/min)
A	90.00	360.00	1440.00	(°/min)
B	90.00	360.00	1440.00	(°/min)
C	90.00	360.00	1440.00	(°/min)

General

Initial Jog Rate: Slow

Keyboard Jog Requires Control Key

Runtime Jog

X
 Y
 Z
 A
 B
 C

Jog Distances

	Jog Panel Radio Buttons	Actual Distances Moved					
		X (inches)	Y (inches)	Z (inches)	A (degrees)	B (degrees)	C (degrees)
Dist. 1:	0.100000	0.100000	0.100000	0.100000	0.090000	0.090000	0.090000
Dist. 2:	0.010000	0.010000	0.010000	0.010000	0.000000	0.000000	0.000000
Dist. 3:	0.001000	0.001000	0.001000	0.001000	0.000000	0.000000	0.000000

Jog Rates

Slow, Medium, Fast – These columns correspond to the jog rate radio buttons on the Jog Control Panel. Techno executes all jog moves using the rates you enter in these fields.

Jog Distances

Jog Panel Radio Buttons – Techno displays these values on the three jog mode buttons with discreet distances on the Jog Control Panel (the middle three radio buttons).

Actual Distances Moved – Techno rounds all discreet distances to the nearest step for each axis. These fields show the actual distance each axis will move after rounding.

General

Initial Jog Rate – When you start the program, this is the initial jog rate setting on the Jog Control Panel.

Keyboard Jog Requires Control Key – When checked, the operator must hold down the Control key when using the keyboard to jog.

Runtime Jog

X, Y, Z, A, B, C – For any axis checked, runtime jog buttons appear on the G-Code Control Panel on the main screen. The runtime jog buttons allow you to jog the axis while a G-Code file is running. The axis moves at the Slow Jog Rate. Note that the runtime jog axis may not be specified in any G-Code file you run.

Pendant Settings

The Flex Pendant is a versatile accessory that you can customize to your particular application. You can assign functions to each button to create your own custom layout.

The Pendant command on the Configuration menu displays the following panel:

Layer	Button	Function
Green	1	None
Green	2	None
Green	3	None
Green	4	None
Green	5	None
Green	6	None
Green	7	None
Green	8	None
Green	9	None
Green	10	None
Green	11	None
Green	12	None
Green	13	None
Green	14	None
Green	15	None
Green	16	None
Green	17	None
Green	18	None
Green	19	None
Green	20	None
Red	1	None
Red	2	None
Red	3	None
Red	4	None
Red	5	None
Red	6	None
Red	7	None
Red	8	None
Red	9	None
Red	10	None
Red	11	None

Display Shortcut
 Use Flex Pendant
 Pendant Buttons:
 1 2 3 4
 5 6 7 8
 9 10 11 12
 13 14 15 16
 17 18 19 20

Use Flex Pendant – Tells Techno to search for the pendant each time you start the software.

Use Standard Layout – Sets the entire Function column to match the standard layout (overwriting all current Function settings).

Pendant Table

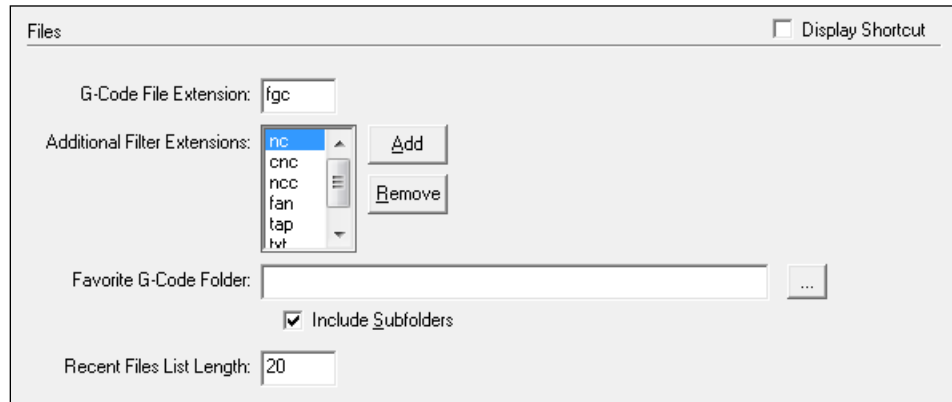
Layer – The pendant layer (Red or Green).

Button – The button number as shown on the diagram.

Function – The function for the specified layer and button. You’ll create a custom layout by setting the function for every button.

File Settings

The Files command on the Configuration menu displays the following panel:



G-Code File Extension – Sets the file extension used when opening and saving G-Code files. If you normally open files made by the Techno DXF import or files you’ve written in the Techno editor, set this to “FGC”. If you normally open files made by another CAM program, enter its standard G-Code file extension (such as “NC”).

This setting is also used by the favorite G-Code menu described below and in the Main Screen Features section.

Additional Filter Extensions – Defines a list of file extensions used in the filter when opening G-Code files. This feature lets you easily open files with a variety of extensions.

Favorite G-Code Folder – Sets the folder used for the favorite G-Code menu. Use the browse (“...”) button to locate the desired folder. See the Main Screen Features section for more information on this menu.

Include Subfolders – Tells Techno whether or not to add all G-Code files found in subfolders of the Favorite G-Code Folder.

Recent Files List Length – Sets the number of recently used G-Code files displayed in the File menu.

Message Settings

Techno lets you turn off some warning messages. This is a convenience feature intended for experienced Techno users only. Don’t turn off any warning messages until you’re very familiar with operation of the software and your machine tool.

The Messages command on the Configuration menu displays the following panel:



Reset Program – Directs Techno to display a warning message whenever you choose Reset on the G-Code Control Panel.

Not Starting at Program Zero – Directs Techno to display a warning message whenever you choose Start on the G-Code Control Panel, and the program coordinates have not been zeroed (at start of file only).

Not Starting on Toolpath – Directs Techno to display a warning message whenever you choose Start on the G-Code Control Panel, and the machine tool is not on the toolpath.

Zero Program Coordinates, Zero Machine Coordinates, Zero Relative Coordinates – Directs Techno to display a warning message whenever you zero the coordinates.

Set Machine Coordinates to Home – Directs Techno to display a warning message whenever you set machine coordinates to home.

Clear Machine Coordinates – Directs Techno to display a warning message whenever you choose Clear (on the DRO Box).

Changed Tool Rack Z Settings – Directs Techno to display a warning message whenever you change data in the Z Raised, Z Near Grip or Z Near Release columns on the Tool Change panel of the Configuration dialog box. The message appears only if the Tool Change Mode is set to Automatic Tool Changer.

Changed Tools in Changer – Directs Techno to display a warning message whenever you make changes on the Tools in Changer panel of the Configuration dialog box.

Feed Hold Input Line Tripped – Directs Techno to display a message whenever a Feed Hold input line triggers a feed hold.

Move After Zeroing Program Coordinates – Directs Techno to display a message before the optional automatic move that follows zeroing program coordinates.

Changed Tooling Configuration – Directs Techno to display a confirmation message asking whether or not to save changes to the tooling file when the tooling file is closed (including exiting the program).

Feedrate Override Not 100% - Directs Techno to display a message when you load a G-Code program and the Feedrate Override is not at 100%.

Exit Program – Directs Techno to display a warning message when you exit the program.

Security Settings

Techno lets you password protect all configuration panels and some features of the software.

The Security command on the Configuration menu displays the following panel:

The screenshot shows the Security configuration window. At the top right is a checkbox for 'Display Shortcut'. Below it is a checkbox for 'Use System Password'. Underneath are two input fields for 'Password:' and 'Password Confirmation:', with a '(a-z, 0-9)' restriction. The window is divided into two main sections: 'Password protects these configuration panels:' and 'Password protects these features:'. Each section contains a list of checkboxes for various system components. In the 'features' list, the 'Security' checkbox is checked, while all others are unchecked.

Use System Password – Turns on the password protection system. When you use password protection:

- Techno presents a password dialog whenever a protected feature is used.
- Techno removes all protected configuration panels from the Configuration menu and adds an “Edit Protected” command near the top of the menu. The Edit Protected command provides access to all protected panels, including the Security panel, after the password is entered.

Password – Sets the password for all password-protected features. The password may contain only letters and numerals. It is not case sensitive (capital and lowercase letters are interchangeable). If you forget your password, please contact technical support for assistance.

Password Confirmation – A confirmation of the Password is required to ensure correct spelling.

Configuration Panel Checkboxes – Indicate which configuration panels are password-protected. When a configuration panel is checked, Techno removes it from the Configuration menu on the main screen. When you choose Configuration : Edit Protected, you'll be prompted to enter the password to continue. Once you enter the correct password, the Configuration dialog appears, with all configuration panels displayed.

Feature Checkboxes – Indicate which features are password-protected. When a feature is checked, you'll be prompted to enter the password when using that feature. The checkboxes restrict access to features as follows.

Switch G-Code File

File Menu : New G-Code (Main Screen & Editor)

File Menu : Open G-Code (Main Screen & Editor)

File Menu : Close G-Code (Main Screen & Editor)

File Menu : Recently Used Files (Main Screen & Editor)

File Menu : Editor (Main Screen, only when there's no file open)

File Menu : Import DXF (Main Screen)

File Menu : Save G-Code As (Editor)

Switch Setup File

File Menu : New Setup

File Menu : Open Setup

File Menu : Save Setup As

File Menu : Recent Setup

Switch Tooling File

File Menu : New Tooling

File Menu : Open Tooling

File Menu : Save Tooling As

File Menu : Recent Tooling

Editor

File Menu : Editor

Edit User Variables

User Variable Box : Edit

Change DRO's

View Menu : DRO

General Preference Settings

The Preferences:General command on the Configuration menu displays the following panel:

Copy Coordinates

These settings affect the behavior of the coordinate-copying commands on the Edit menu (Copy Program Coordinates, Copy Machine Coordinates, and Copy Relative Coordinates.) These commands copy the current coordinates to the Windows clipboard so you can paste them into the editor or into another software application.

Format as Plain Text – Tells Techno to format the coordinates in a simple text format (see example under Include Axis Letters below).

Include Axis Letters – When checked, Techno includes axis letters before each coordinate. This is handy when you're creating a G-Code program from the copied coordinates.

Example: The current program coordinate is X=1, Y=2, Z=-3 and Copy Program Coordinates is chosen from the Edit menu.

Axis Letters Included: “X1.0000 Y2.0000 Z-3.0000” goes on the clipboard

Axis Letters Not Included: “1.0000 2.0000 -3.0000” goes on the clipboard

Format as G-Code Data (XML) – Tells Techno to format the coordinates as required for G-Code Data files. For more information see “G-Code Data Files” in the System Programming section.

Create Complete Line – Tells Techno to create a complete XML line that includes the required tags. For more information see “G-Code Data Files” in the System Programming section.

Clear the Clipboard Before Each Copy – When checked, Techno deletes the previous coordinates on the clipboard before copying new coordinates there. If you want to collect a series of points before pasting the entire group, uncheck this checkbox so the points accumulate on the clipboard.

Teach Box

Coordinate Decimals – The number of decimal places for coordinates displayed in the Teach Box, and for coordinates in the generated G-Code file.

Feedrate – The feedrate for the G-Code file generated using the Teach Box.

Language – The language displayed for all controls and messages. After you select a new language you must restart the software for the new language to take effect.

Setting Machine Zero

Techno lets you set Machine Zero two ways: Homing to switches or using the Zero All command on the Machine Set button. Either method establishes the machine tool envelope, which provides these benefits:

- The software will safely ramp down the machine tool to a complete stop at the defined limits of the machine tool envelope.
- The software checks each G-Code file before running to make sure the toolpath fits completely within the machine tool envelope.

It's better to home to switches since it provides these added benefits:

- It's much more convenient, since Techno can find Machine Zero automatically.
 - You can establish Machine Zero at the same repeatable location anytime you like (within the repeatability tolerance of the switch). A repeatable Machine Zero lets you restart unfinished jobs, and reliably position the tool relative to work-holding fixtures.
 - You can use the homing process to check for accuracy of the machine tool's position.
- ▶ To Set Machine Zero by Homing to Switches
1. Make sure you've correctly entered the homing-related setup parameters described in the “Basic Homing Settings” and “Input Line Settings” sections above.
 2. Choose the Home button on the Control Selection Box.

3. If Machine Zero has not been set, make sure all axis checkboxes are checked.
 4. Choose the Seek Home button. One axis at a time, Techno drives the axis until it finds and backs off the switch. To stop the process, choose the Feed Hold button, or hit any key on the keyboard besides Ctrl or Shift.
 5. If Machine Zero was already set before homing, Techno displays a dialog showing the discrepancy between the previous Machine Zero and the new Machine Zero just found. Choose Yes if you want to use the new Machine Zero just found, or No if you want to keep the existing Machine Zero.
 6. If you chose Yes, Techno zeros the machine coordinates and displays the machine tool envelope in the Viewport Box.
- ▶ To Set Machine Zero Using the Zero Button
1. Make sure you've correctly entered the homing-related setup parameters described in the "Machine Tool Settings" and "Homing Settings" section above.
 2. Choose the Jog button on the Control Selection Box.
 3. Jog the tool to the home end of each axis (as defined on the Basic Homing panel of the Configuration dialog box). Move each axis to within a short distance of its physical limit.
 4. If this is the first time you're setting Machine Zero on your machine, it might be helpful to scratch a reference line or attach a Vernier scale to two adjacent moving parts on each axis. This will help you find the same home position again in the future.
 5. Choose Zero All from the Set button next to the Machine label in the DRO Box. When the confirmation dialog appears, choose Yes (see "Message Settings" above for details on suppressing this dialog if desired).
 6. If Machine Zero was already set, Techno displays a dialog showing the discrepancy between the previous Machine Zero and the new Machine Zero. Choose Yes if you want to use the new Machine Zero, or No if you want to keep the existing Machine Zero.
 7. If you chose Yes, Techno zeros the machine coordinates and displays the machine tool envelope in the Viewport Box.

4. System Programming

Techno supports ANSI standard G-Code to control machine tool movement and peripheral devices. This section describes how to create and open G-Code files, and the G-Codes supported.

There are five ways you can open or create G-Code files:

- Open an existing G-Code file created by a CAM program, Techno or any other source.
- Import a DXF file created by a CAD or drawing program.
- Write a G-Code program directly in the Techno editor.
- Create a file using a cycle. See “Cycle Control Panel” in the Main Screen Features section for more information.
- Teach points and create a point-to-point G-Code file. See “Teach Box” in the Main Screen Features section for more information.

Opening a G-Code Program

- ▶ To open an existing program manually
 1. Choose Open G-Code from the File menu. The Open G-Code File dialog box appears.
 2. Navigate to the G-Code file you wish to open.
 3. Double-click the file name, or click on the file name and choose OK.

You can configure Techno to open a program when an input line is tripped. For more information see “Input Line Settings” in the Initial Setup section.

Importing a DXF File

Techno provides a very useful 2D DXF import feature. The DXF import automatically arranges all lines and arcs that have common endpoints into features. The entire toolpath is then optimized to reduce total machining time for the part. The DXF import assumes the part surface is at a Z program height of 0.0. All geometry is treated as either holes or cutter paths with no offsets.

- ▶ To import a DXF file
 1. Choose Import DXF from the File menu. The Open dialog box appears.
 2. Navigate to the DXF file you wish to import.
 3. Double-click the file name, or click on the file name and choose OK.

4. The Save As dialog box appears, asking you the name of the new G-Code file created by the DXF import. By default, it will use the DXF file name with an “FGC” extension in the same drive and folder in which the DXF file resides. If this is acceptable, choose OK. Otherwise, you can rename the file and/or choose an alternate folder.
5. The Import panel of the Configuration dialog box will appear.

6. Fill in the values for each of the following fields:

General

Scale – A multiplication factor for the toolpath in the XY plane only. For example, if you enter 2, the toolpath generated will be double the size of the original geometry defined in the DXF file. Note that the values you enter for positioning the Z axis are unaffected by the scale factor.

Decimals – The number of decimal places to use for all coordinates. A higher number can help eliminate extra backlash compensation moves caused by rounding error.

Join Tolerance – If two drawing entities, such as two lines, are touching end to end, Techno treats them as a single feature to machine

without lifting the tool. Due to rounding or drawing error, two entities that are meant to be joined may not actually touch end to end. The DXF import will automatically join any entities whose endpoints are less than the Join Tolerance apart.

Positioning – Choose the positioning mode you prefer: G90 or G91. For more information see G90 Absolute Positioning Mode and G91 Incremental Positioning Mode later in this section.

Optimize Toolpath – When checked, Techno optimizes the toolpath for shortest overall cutting time. When unchecked, Techno cuts part features in the order they appear in the DXF file. This gives you more control over the order in which part features are cut. (A feature is a contiguous set of lines, arcs and polylines, or a standalone circle).

Line Numbers – Check this box if you want the DXF import to number all of the G-Code lines in the program it creates.

Combine Layers – When checked, Techno imports the DXF file as if all layers were combined into a single layer. When unchecked, Techno generates a separate toolpath for each layer, one layer at a time.

Return to Program Zero – Check this box you want the machine tool to move back to Program Zero at the end of the program.

Z Axis Coordinates

From File – Check this box if you want Techno to use Z axis values from the DXF file for all tool-down Z positions, instead of generating these values based on your settings.

Tool Up – The safe Z program coordinate to which the tool will move before rapid moves between two features.

Incremental Depth of Cut – The incremental depth for each milling pass. For example if the final tool down is -0.2500” and the incremental depth of cut is 0.0625” then four passes would be cut on each feature to get to the final depth of cut. (-0.0625, -0.1250, -0.1875, -0.2500). If the final tool down is -0.3000” and the incremental depth of cut is 0.0625” then five passes would be cut on each feature to get to the final depth of cut (-0.0625, -0.1250, -0.1875, -0.2500, -0.3000).

Final Tool Down (Milling) – The final depth (program coordinates) to which the tool will cut each feature.

Final Tool Down (Holes) – The final depth (program coordinates) to which the tool will drill each hole.

Program Zero Location

This setting determines where Techno will place Program Zero relative to the coordinate system of the DXF file. There are two options:

- X, Y of Import File

You specify the X and Y location in the DXF file's coordinate system that Techno will place at Program Zero in the G-Code file.

- Relative to Toolpath

Techno sets Program Zero based on the specified point on the imaginary rectangle that contains all DXF file entities.

Toolpath Point – The point on the imaginary rectangle that contains all entities.

X Offset from Zero to [Point], Y Offset from Zero to [Point] – Allows you to offset the toolpath from Program Zero.

Feedrates

XY Feedrate – The feedrate for cutting moves in the XY plane.

Plunge Feedrate – The feedrate for all Z axis plunges.

Circles

Techno can handle circles several ways. The choices are:

- Mill All – Tells Techno to move the tool along the perimeter of all circles.
- Drill All – Tells Techno to drill a hole at the center point of all circles.
- Drill in Diameter Range – Tells Techno to drill at the center of each circle whose diameter is in the specified range. Techno will move along the perimeter of each circle outside of the range.

Ellipses and Splines

Chord Error – When Techno imports an ellipse or spline, it generates many short linear (G01) moves to closely approximate the original curve. This setting controls how closely the set of lines will match the original curve; it's the maximum deviation between the curve and each generated line. The smaller the number, the more G01 moves Techno will create in the G-Code file for each ellipse or spline.

Cutter Compensation

Techno can automatically apply cutter compensation to your toolpath.

Offset Direction

- None – Techno will not apply cutter compensation.
- Outside – Techno will offset the tool toward the outside of the outermost closed shape.
- Inside – Techno will offset the tool toward the inside of the outermost closed shape.

When there are features contained within other features (eg. a plate with cutouts), the Offset Direction applies to the outermost feature. Any feature that is contained within another feature will be offset in the opposite direction. For example, if the DXF file had two concentric circles and the Offset Direction were set to Outside, the outer circle would be offset to the outside and the inner circle would be offset to the inside.

Tool – The tool that will be used.

Cut Innermost Features First – Techno will start by cutting the features that are contained within other features. For example, if the DXF file had two concentric circles, Techno would cut the inner circle before the outer circle.

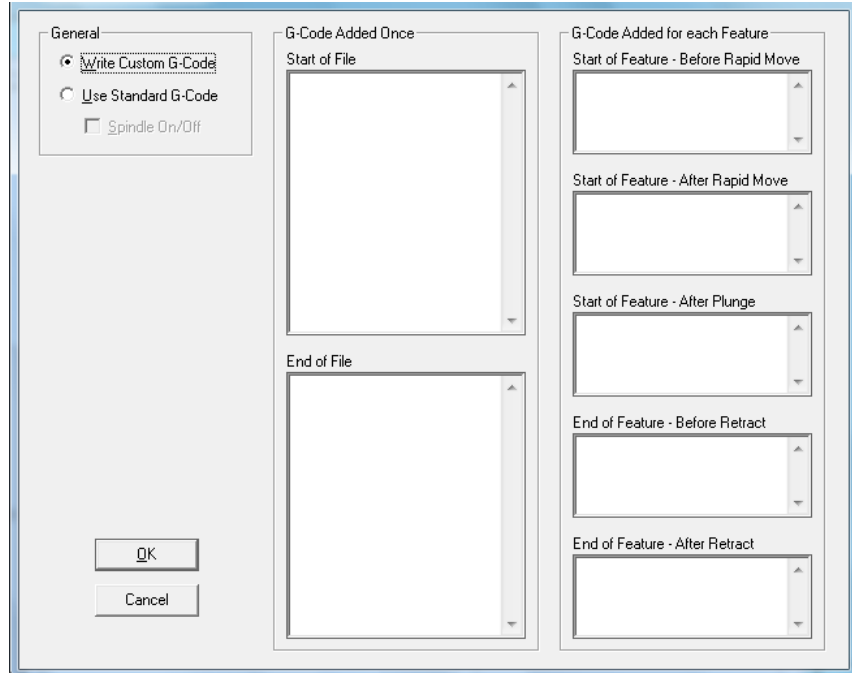
Milling Direction – Choose between conventional and climb milling.

Spindle Direction – Choose between clockwise (CW) and counterclockwise (CCW). Techno must know this to determine the direction of motion required for the specified Milling Direction.

Additional G-Code

You can specify additional lines of G-Code to be inserted into the file. You can insert G-Code at the start and end of the file, and the start and end of each feature.

To enter additional G-Code, click Edit to display the Additional G-Code dialog box. The following dialog will appear:



General

Write Custom G-Code – Makes all G-Code fields available for you to enter any number of G-Code lines.

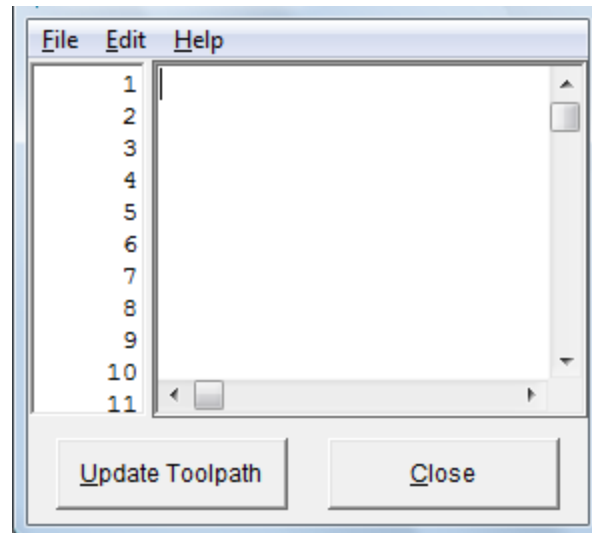
Use Standard G-Code – Techno automatically enters G-Code into the fields.

7. Choose OK.
8. Techno generates and loads the G-Code file.

Using the Program Editor

Techno provides a full-featured editor for creating or modifying G-Code files. If you prefer, you can also use your own editor such as WordPad (which comes with Windows) or Microsoft Word. If you use a different editor, make sure you save the file as 'text only' and add an "FGC" extension to the file name.

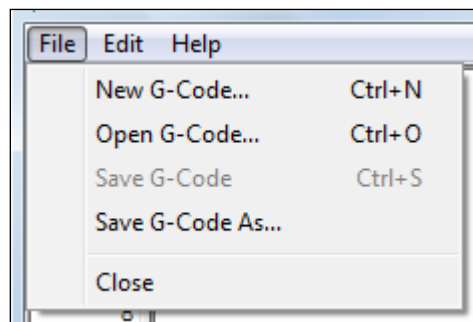
- ▶ To open the editor
 1. Choose Editor from the File menu, or double-click the Program Listing Box. The editor dialog box will appear.



2. Type or edit your program in the scrolling text box. Refer to the “G and M-Code Reference” section below to learn more about using G-Code.

The remainder of this section describes each menu command and button available in the editor.

File Menu



New G-Code – Opens a new, empty file.

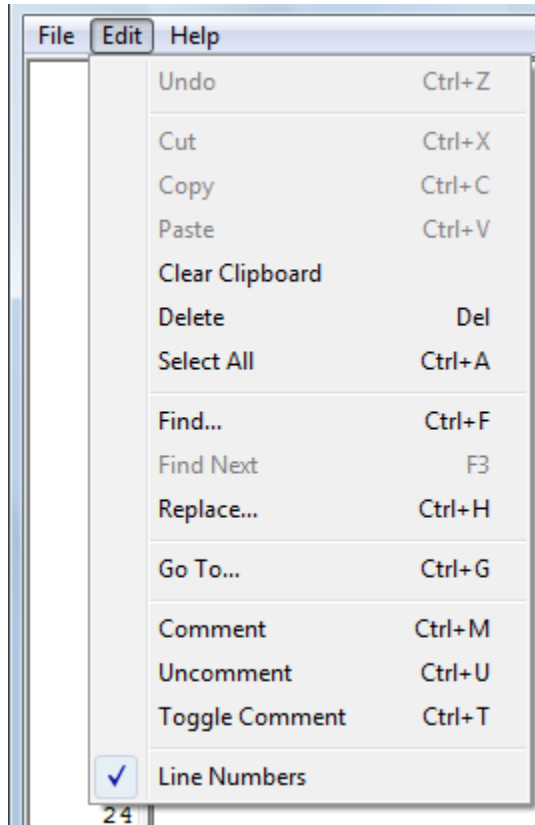
Open G-Code – Opens an existing G-Code file, checks the file for errors, draws the toolpath in the Viewport Box, and displays the program in the Program Listing Box.

Save G-Code – Saves the G-Code file.

Save G-Code As – Saves the G-Code file under a new name.

Close – Closes the editor.

Edit Menu



Undo – Retracts the last edit you made.

Cut – Moves the currently selected text to the Windows clipboard.

Copy – Copies the currently selected text to the Windows clipboard.

Paste – Inserts the text on the Windows clipboard into the editor, at the current cursor location.

Clear – Deletes the currently selected text.

Select All – Selects all the text in the editor.

Find – Opens the Find dialog box, which lets you search for text.

Find Next – Finds the next occurrence of the search text.

Replace – Opens the Replace dialog box, which lets you search for text and replace it with new text.

Go To – Opens the Go To dialog box, which lets you jump to any line in the file.

Comment – Adds an open parenthesis to the beginning of all selected lines, so Techno will ignore the lines.

Uncomment – Removes the open parentheses from the beginning of all selected lines, so Techno will execute the lines.

Toggle Comment – Either adds or removes an open parenthesis to the beginning of all selected lines depending on their current state (e.g. the selected line(s) is commented out this command will remove this open parenthesis).

Line Numbers – Turns editor line numbering on and off. Note that these numbers are different than the “N” numbers found in some G-Code Files.

Help Menu

The help menu is identical to the help menu on the main screen. For more details, see “Pull-Down Menu Bar” in the Main Screen Features section of this manual.

Buttons

Update Toolpath – Redraws the toolpath to reflect all changes you’ve made to the G-Code file.

Close – Closes the editor.

G-Codes Supported

G00	Rapid Tool Positioning
G01	Linear Interpolated Feedrate Move
G02	Clockwise Circular Feedrate Move
G03	Counter Clockwise Circular Feedrate Move
G04	Dwell
G17	XY Plane Selection
G18	XZ Plane Selection
G19	YZ Plane Selection
G20	Inch Units (same as G70)
G21	Metric Units (same as G71)
G27	Home to Switches
G28	Move to Reference Point 1
G29	Rapid Return from Reference Point
G29.1	Feedrate Return from Reference Point
G30	Move to Reference Points 2-20
G31	Seek Sensor
G40	Cancel Cutter Compensation
G41	Cutter Compensation (Left)
G42	Cutter Compensation (Right)
G43	Tool Length/Geometry Compensation (Plus)
G44	Tool Length/Geometry Compensation (Minus)
G49	Cancel Tool Length/Geometry Compensation
G50	Cancel Scaling
G51	Scaling
G52	Use Local Coordinate System
G53	Linear Move to Machine Coordinates (Rapid)
G53.1	Linear Move to Machine Coordinates (Feedrate)
G54-59	Set Program Zero (Fixture Offsets)
G54.1	Set Program Zero (Extended Fixture Offsets)
G70	Inch Units (same as G20)
G71	Metric Units (same as G21)

G73	High Speed Peck Drilling Cycle
G74	Left-Handed Tapping Cycle
G76	Thread Cutting Cycle
G80	Cancel Canned Cycle
G81	Drilling Cycle
G82	Counterboring Cycle
G83	Peck Drilling Cycle
G84	Standard Tapping Cycle
G85	Reaming Cycle
G90	Absolute Positioning Mode
G91	Incremental Positioning Mode
G92	Set Program Zero
G93	Inverse Time Feedrate
G94	Standard Feedrate
G98	Return to Initial Level (Canned Cycles)
G99	Return to Rapid Level (Canned Cycles)
G120	Sense Tool Length
G140	Set Engraving Font
G141	Engrave Text
G150	Increment Counter
G151	Set Counter
G160	Increment Tool Use Counter
G161	Set Tool Use Counter
G170	Find Fiducials
G179	Cancel Fiducial Correction
G180	Turn Off Safe Envelope
G181	Turn On Safe Envelope
G190	Set PWM/PFM Output
G210	Select Variable Output Port

M-Codes Supported

M00	Program Pause
M01	Optional Program Pause

M02	End of Program
MXX	Custom Programmable (See “M-Code Definitions” in the Initial Setup section of this manual). Typical functions are: <ul style="list-style-type: none">M03 Spindle OnM05 Spindle OffM06 Tool ChangeM07 Mist Coolant OnM08 Flood Coolant OnM09 Coolant OffM50 Plasma OnM51 Plasma Off
M30	End of Program (Reset)
M30.1	End of Program (Reset and Restart)
M98	Subroutine Call
M99	Return From Subroutine
M100	Wait for Input Line (Normal State)
M101	Wait for Input Line (Tripped State)

Other Commands Supported

F	Set Feedrate
T	Select Tool
S	Set Variable Output Port (eg. spindle speed)
()	Comment
/	Optional Line
BEGINGRID	Begin a Grid of Parts
ENDGRID	End a Grid of Parts

Advanced Keyword Commands and Functions

EnableFO	Enable Feedrate Override
DisableFO	Disable Feedrate Override
SetFO	Set Feedrate Override Percent
SaveState	Save Current Modal States
RestoreState	Restore Saved Modal States

GetRackIndex Get Tool Rack Index

Key Programming Concepts

There are two basic programming concepts you should understand before learning the G and M codes: Mode and Absolute vs. Incremental.

Mode

Most G-code commands supported by Techno are modal, meaning they put the system into a particular mode of operation and don't need to be repeated on every program line. A modal command stays in effect until another command changes the mode. Related modal commands that affect one aspect of program execution are called a mode group.

The following list shows the mode groups supported by Techno.

Movement

- G00 Rapid Tool Positioning
- G01 Linear Interpolated Feedrate Move
- G02 Clockwise Circular Feedrate Move
- G03 Counter Clockwise Circular Feedrate Move

Circular Interpolation

- G17 XY Plane Selection
- G18 XZ Plane Selection
- G19 YZ Plane Selection

Units

- G20 Inch Units (also G70)
- G21 Metric Units (also G71)

Cutter Compensation

- G40 Cancel Cutter Compensation
- G41 Cutter Compensation (Left)
- G42 Cutter Compensation (Right)

Tool Length/Geometry Compensation

- G43 Tool Length/Geometry Compensation (Plus)
- G44 Tool Length/Geometry Compensation (Minus)
- G49 Cancel Tool Length/Geometry Compensation

Scaling

G50 Cancel Scaling

G51 Scaling

Set Program Zero

G54-59 Set Program Zero (Fixture Offsets)

G54.1 Set Program Zero (Extended Fixture Offsets)

G92 Set Program Zero (XYZA Parameters)

Canned Cycles

G73 High Speed Peck Drilling Cycle

G74 Left-Handed Tapping Cycle

G76 Thread Cutting Cycle

G80 Cancel Canned Cycle

G81 Drilling Cycle

G82 Counterboring Cycle

G83 Peck Drilling Cycle

G84 Standard Tapping Cycle

G85 Reaming Cycle

Canned Cycle Return

G98 Return to Initial Level

G99 Return to Rapid Level

Positioning

G90 Absolute Positioning Mode

G91 Incremental Positioning Mode

Safe Envelope

G180 Turn Off Safe Envelope

G181 Turn On Safe Envelope

Feedrate

G93 Inverse Time Feedrate

G94 Standard Feedrate

Independent Modal Commands

G52	Use Local Coordinate System
F	Set Feedrate
S	Set Spindle Speed
G140	Set Engraving Font

Absolute vs. Incremental

All moves are either absolute or incremental. In an absolute move, the ending point is defined relative to Program Zero. In an incremental move, the ending point is defined relative to the current tool location. The G90/G91 commands tell the system which of these two modes to use (described below).

While there will be cases where incremental programming is useful, generally you should define your moves as absolute since it is a less error prone method of programming. All of the examples in the following section use absolute positioning unless otherwise noted.

G and M-Code Reference

G00 Rapid Tool Positioning

The G00 command moves the tool to the designated coordinate at the rapid rate using linear interpolation. The rapid rate is calculated from the Maximum Feedrates defined on the Feedrate/Ramping panel of the Configuration dialog box.

Example:

```
G00 X1.0 Y2.0 Z1.5
```

Moves the tool to program coordinates X=1.0, Y=2.0, Z=1.5 at the rapid rate

When using G00, there are several things to keep in mind:

- You do not need to specify all coordinates, only the ones for which you want movement.

Example:

```
G00 X4.0 Y3.0
```

Moves the tool to program coordinates X=4.0, Y=3.0, leaving the Z position unchanged

- This is a modal command, meaning that all successive moves will be treated as rapid moves until another modal move command (G01, G02 or G03) occurs.

Example:

```
G00 X1.0 Y2.0 Z1.5      Rapid Move
X4.0 Y6.5 Z1.0         Rapid Move
G01 X3.0 Y3.0 Z1.4     Feedrate Move
X2.8 Y1.4 Z0           Feedrate Move
```

- The interpretation of the coordinates depends on the G90/G91 command in effect.
- For rotary axes, Techno will make the shortest possible move in the direction indicated by the command (always less than 360 degrees.)

Example (starting at program coordinate A = 0):

```
G00 A365                Rapid move 5 degrees, after which the DRO
                        reads A=365
```

- Express Mode is a setting that affects how the Signal Generator processes motion commands. It's located on the Controller panel of the Configuration dialog box. When the Signal Generator is running in Express Mode, it can move up to five axes simultaneously.

When **not** in Express Mode, Techno can only move up to three axes simultaneously. If you specify a move on four axes, Techno will perform two separate moves, one for the Z axis and one for the X-Y-A axes. The Z move will occur first if the direction is towards the home end of the axis (specified by the Home End setting on the Basic Homing panel of the Configuration dialog box). Otherwise the X-Y-A move will occur first. If this behavior doesn't suit your needs, use two G00 commands.

Example (starting at program zero, home end for Z axis is positive):

```
G00 X1.0 Y2.0 Z1.5 A0   Moves the tool to program coordinates
                        X=1.0, Y=2.0, Z=1.5, leaving the A position
                        unchanged (the A parameter is included but
                        does not specify any motion)

G00 X2.0 Y3.0 Z1.0 A90 First moves the tool to X=2.0, Y=3.0, A=90,
                        leaving the Z position unchanged, then
                        moves the tool to Z=1.0, leaving the X, Y
                        and A axes unchanged
```

G01 Linear Interpolated Feedrate Move

The G01 command moves the tool to the designated program coordinate at the specified feedrate using linear interpolation.

Example:

```
G01 X2.0 Y1.0 Z-1.5 F2.0      Moves the tool to program coordinates X=2.0,  
                              Y=1.0, Z=-1.5 at a feedrate of 2.0 in/min
```

When using G01, there are several things to keep in mind:

- You do not need to specify all coordinates, only the ones for which you want movement.

Example:

```
G01 X4.0 Y3.0 F2.0          Moves the tool to program coordinates  
                              X=4.0, Y=3.0, leaving the Z position  
                              unchanged
```

- This is a modal command, meaning that all successive moves will be treated as linear feedrate moves until another modal move command (G00, G02 or G03) occurs.
- The interpretation of the coordinates depends on the G90/G91 command in effect.
- The F command is used to designate a feedrate. The feedrate set with the F command is modal (stays in effect until another F command occurs).

Example:

```
G01 X4.0 Y3.0 Z1.0 F7.0     Moves the tool to program coordinates  
                              X=4.0, Y=3.0, Z=1.0 at a feedrate of 7.0  
                              in/min  
  
X2.0 Y2.5                   Moves the tool to program coordinates  
                              X=2.0 Y=2.5, leaving the Z axis unchanged  
                              at Z=1.0 (the feedrate remains 7.0 in/min)
```

- For any move that includes a rotary axis (A, B or C), there are two ways Techno can interpret the current F setting. Techno uses your setting for 'F Command Interpretation for Rotary Moves' on the Rotary Axes panel of the Configuration dialog box.

Linear Feedrate (Distance/Minute) – This option directs Techno to interpret the feedrate command (F) as a linear feedrate. The speed of the tool tip relative to its point of contact on the rotating workpiece will be at the specified F value. In this mode, you can use a single F command to cover linear, linear plus rotary, and rotary moves.

Rotary Feedrate (Degrees/Minute) – This option directs Techno to interpret the feedrate command (F) as a rotary feedrate. The rotary

axis controls the timing. The F parameter is interpreted as degrees/minute.

Example:

```
G01 X1 A90 F360
```

The A axis turns 90 degrees while the X axis moves 1 inch. The A axis turns at exactly 360 degrees/minute. The timing for the X axis is based on the A axis motion and is not explicitly listed in the G-code file.

Note that if you choose the rotary feedrate interpretation, the F feedrate in use is always interpreted as degrees/minute for any move that contains motion on a rotary axis. For all other moves, it's interpreted as inches/minute (or mm/min). Therefore, you must be very careful to reset F every time you switch between pure linear and rotary-included moves.

- Express Mode is a setting that affects how the Signal Generator processes motion commands. It's located on the Controller panel of the Configuration dialog box. When the Signal Generator is running in Express Mode, it can move up to five axes simultaneously.

When **not** in Express Mode, Techno can only move up to three axes simultaneously. If you specify a move on four axes, Techno will perform two separate moves, one for the Z axis and one for the X-Y-A axes. The Z move will occur first if the direction is towards the home end of the axis (specified by the Home End setting on the Basic Homing panel of the Configuration dialog box). Otherwise the X-Y-A move will occur first. If this behavior doesn't suit your needs, use two G01 commands.

Example (starting at program zero, home end for Z axis is positive):

```
G01 X1.0 Y2.0 Z1.5 A0 Moves the tool to program coordinates
                        X=1.0, Y=2.0, Z=1.5, leaving the A position
                        unchanged (the A parameter is included but
                        does not specify any motion)

G01 X2.0 Y3.0 Z1.0 A90 First moves the tool to X=2.0, Y=3.0, A=90,
                        leaving the Z position unchanged, then
                        moves the tool to Z=1.0, leaving the X, Y
                        and A axes unchanged
```

G02 Clockwise Circular Feedrate Move

The G02 command moves the tool in a clockwise path from the starting point (the current tool position) to the designated ending point in the currently selected plane (see G17-G19). The I, J, and K parameters represent the incremental X, Y, and Z distances (respectively) from the starting point of the arc to the center point of the arc.

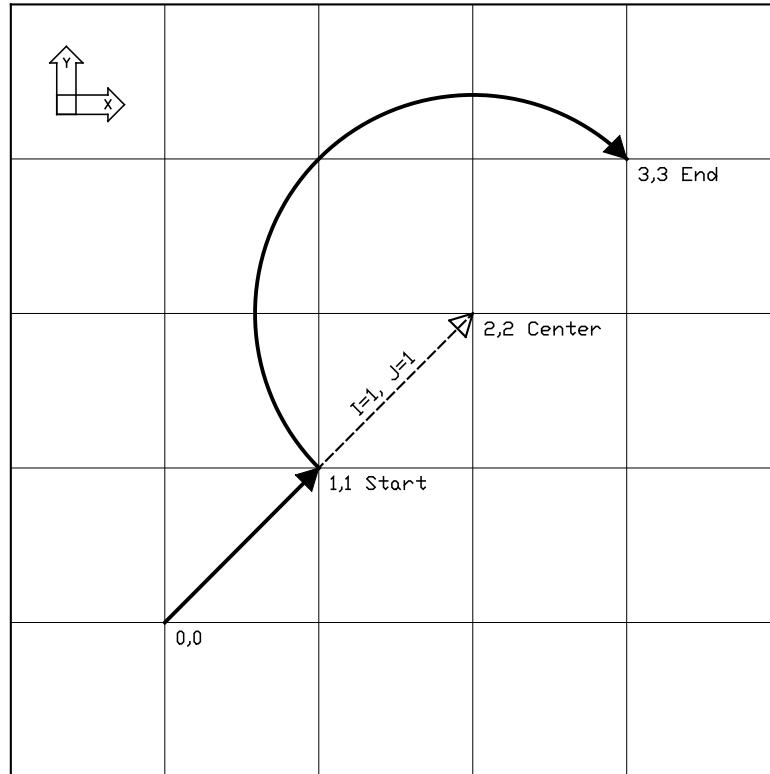
Example:

G01 X1 Y1 F3

Moves the tool to program coordinates X=1, Y=1 at a feedrate of 3 in/min

G02 X3 Y3 I1 J1

Moves the tool using clockwise circular interpolation to program coordinates X=3, Y=3 with a center point of X=2, Y=2 at a feedrate of 3 in/min



An alternative to specifying the distance to the center point is to specify the radius, using the R parameter. Usually this is easier than determining the correct I, J and K values. For any given radius, generally there are two possible arcs: one that sweeps an angle less than 180 degrees, and one that sweeps an angle greater than 180 degrees (see diagram below). To specify an angle less than 180 degrees, make R a positive number; to specify an angle greater than 180 degrees, make R a negative number.

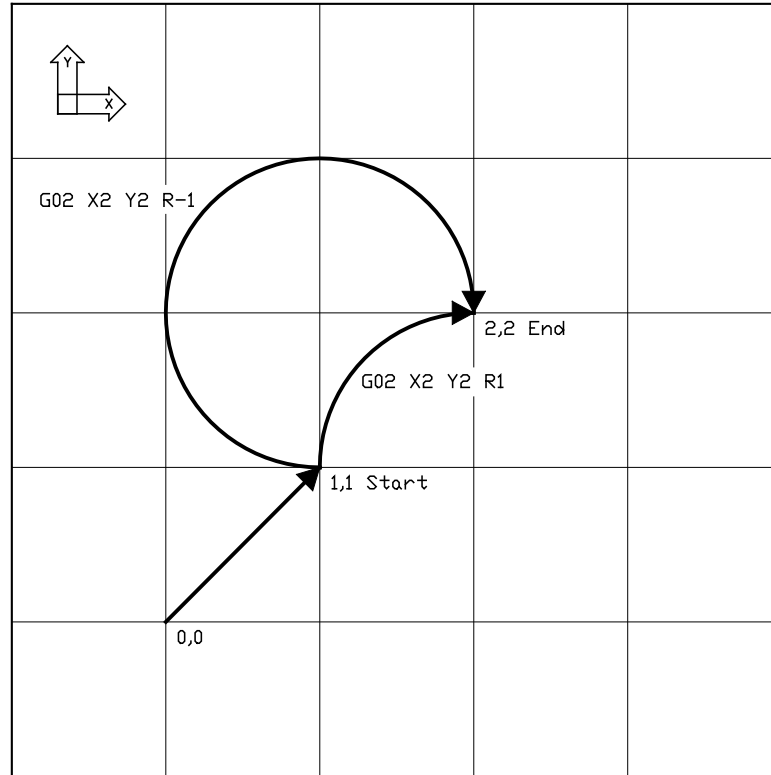
Example:

G01 X1 Y1 F3

Moves the tool to program coordinates X=1, Y=1 at a feedrate of 3 in/min

G02 X2 Y2 R1 (or R-1)

Moves the tool using clockwise circular interpolation to program coordinates X=2, Y=2



When using the R word, please note:

- If the arc sweeps a 180 degree angle, it doesn't matter whether R is negative or positive.
- If the ending point is the same as the starting point, Techno will ignore the command, since the center point cannot be determined.

You can use the G02 command to specify a helical move (helical interpolation). During a helical move, the circular motion described above is combined with linear motion that's perpendicular to the plane containing the arc. For example, circular motion in the XY plane is combined with linear motion along the Z axis to form a helix.

To specify helical motion, add an X, Y or Z parameter to the command, to indicate the ending point of the linear motion. In the following example, a Z parameter has been added to the G02 command for an arc in the XY plane.

Example:

```
G01 X1 Y1 Z1 F3
```

Moves the tool to program coordinates X=1, Y=1, Z=1 at a feedrate of 3 in/min

```
G02 X3 Y3 Z2 I1 J1
```

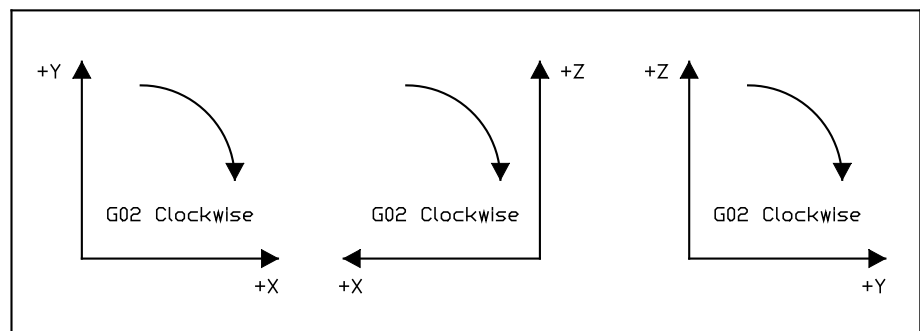
Moves the tool using clockwise circular interpolation to program coordinates X=3, Y=3 with a center point of X=2, Y=2, while simultaneously moving the tool in a straight line along the Z axis to Z=2

The accuracy of the helical move, relative to a theoretically perfect helix, is controlled by the Helical Interpolation Accuracy setting on the G-Code panel of the Configuration dialog box (see “G-Code Settings” in the Initial Setup section).

In knife cutting applications, you can include rotary motion as part of the G02 command, by adding the A, B or C parameter to the G-Code line. The rotary axis will turn proportionately throughout the arc to keep the knife blade tangent to the cutting direction. To use this option you must first configure your machine for knife cutting. Please see “Rotary Axes Settings” in the Initial Setup section for more information. Also note that the Helical Interpolation Accuracy setting on the G-Code panel of the Configuration dialog box applies to these arc moves.

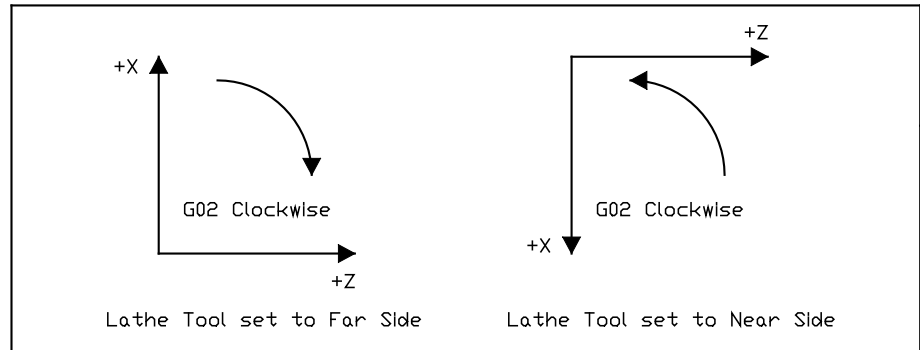
When using G02, there are several things to keep in mind:

- This is a modal command, meaning that all successive moves will be treated as clockwise circular feedrate moves until another modal move command (G00, G01 or G03) occurs.
- The interpretation of the X, Y and Z coordinates depends on the G90/G91 command in effect. The I, J and K values are unaffected by G90/G91.
- The tool will move at the current feedrate set by the last F command.
- Only XY arcs can be cut when G17 is active, only XZ arcs can be cut when G18 is active, and only YZ arcs can be cut when G19 is active. Arcs cannot be specified for the A or W axes.
- The clockwise direction of rotation is as viewed from the positive end of the unused axis (the axis not in the plane of motion). For example, a G02 arc move in the XY plane is clockwise as viewed from the positive end of the Z axis (i.e. from above). The following diagram illustrates this behavior for all three arc planes:



In some cases, a clockwise arc as defined above will be displayed counter clockwise in a viewport. For example, in lathe applications, the Lathe Tool setting (on the Basic Definition panel of the Configuration dialog box) determines whether the X positive direction is up or down in the ZX viewport. When Lathe Tool is set to Near

Side, the X positive direction is down and G02 arcs are displayed counter clockwise on the screen.



G03 Counter Clockwise Circular Feedrate Move

The G03 command is identical to the G02 command, but it moves the tool in a counter clockwise arc instead of a clockwise arc.

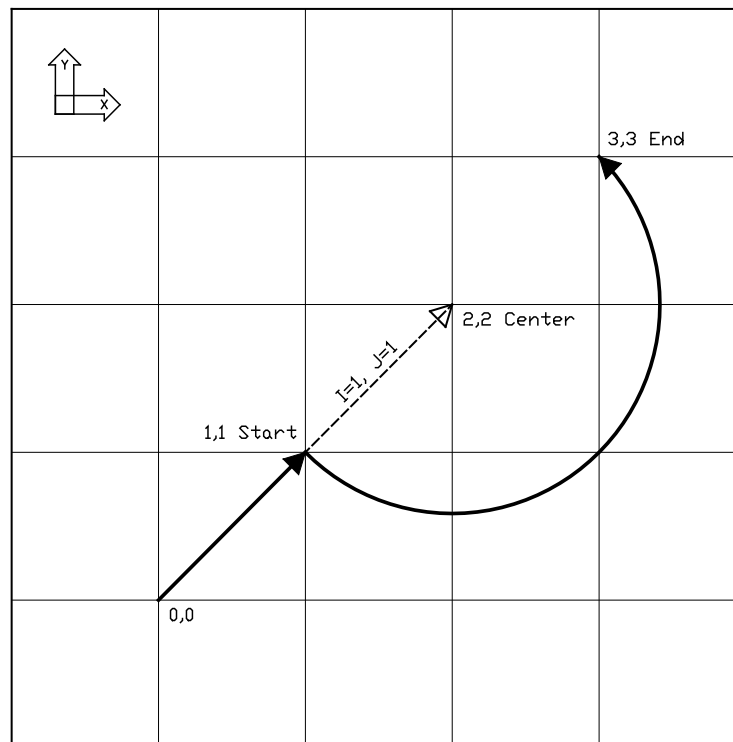
Example:

```
G01 X1 Y1 F3
```

Moves the tool to program coordinates X=1, Y=1 at a feedrate of 3 in/min

```
G03 X3 Y3 I1 J1
```

Moves the tool using counter clockwise circular interpolation to program coordinates X=3, Y=3 with a center point of X=2, Y=2 at a feedrate of 3 in/min



G04 Dwell

The G04 command causes the program to dwell for a specified amount of time. The command format is:

G04 Xn

where n is the number of seconds to wait. For safety reasons there is a maximum time allowed for each dwell command.

Example:

G04 X1.5

Program pauses for 1.5 seconds before moving on to the next line of G-Code

For compatibility reasons, Techno also allows the following format:

G04 Pn

where n is the number of seconds or milliseconds to wait, depending on your “G04 P Parameter Units” setting on the G-Code panel of the Configuration dialog box.

Example (P parameter in milliseconds):

G04 P3000

Program pauses for 3 seconds before moving on to the next line of G-Code

G17, G18, G19 Arc Plane Selection

These commands specify the plane used for circular interpolation as follows:

G17 XY plane

G18 XZ plane

G19 YZ plane

When using G17-G19, there are several things to keep in mind:

- Unless you explicitly use the G18 or G19 command, Techno assumes G17 as the default.
- The three commands are modal, i.e. one command remains in effect until another in the set is used.

G20, G21 Inch Units and Metric Units

The G20 command indicates that all G-Code commands are in inch units. Techno then assumes all distances are in inches and all feedrates are in inches/minute. For compatibility reasons, Techno accepts G70 as equivalent to G20.

The G21 command indicates that all G-Code commands are in metric units. Techno then assumes all distances are in millimeters and all feedrates are in millimeters/minute. For compatibility reasons, Techno accepts G71 as equivalent to G21.

You will only need these commands when the units used in the G-Code file don't match the System Units setting on the System:General panel of the Configuration dialog box. If you don't use either command, Techno assumes all program values are consistent with the configuration setting.

G27 Home to Switches

The G27 command homes the machine tool to its home switches, in a similar manner as the Seek Home button on the Home Control Panel. The purpose of this command is to allow checking for positioning errors, and to allow re-zeroing of any axis. This is especially useful when running large G-Code files. The command lets you home the machine at any point in the file, and optionally have Techno display the homing discrepancy dialog if the positioning error is not within the desired tolerance (which pauses execution of the file).

If you want the G27 command to home only some of the axes on your machine tool, you can limit homing to those axes by adding each axis letter followed by a zero (eg. "X0") after the command. Then, Techno homes only the indicated axes.

Example:

G27	<i>Homes all axes</i>
G27 X0 Y0	<i>Homes the X and Y axes</i>

Note that Machine Zero must be set before you run a G-Code file that uses this command.

For more details, see "Home Control Panel" in the Main Screen Features section, and "Homing Settings" in the Initial Setup section.

G28, G30 Move to Reference Point

The G28 and G30 commands move the tool at the rapid rate to the associated reference point defined on the Reference Points panel of the Configuration dialog box. These positions are defined either in machine or program coordinates. When a position is defined in machine coordinates, Machine Zero must be set for this command to be used. Note that G28 typically is used for moving to the tool change position and/or Machine Zero (reference point 1).

The sequence of axis motions follows a general-purpose scheme that's described under "Point Control Panel" in the Main Screen Features section of this manual.

If you want the G28/G30 command to move only some of the axes on your machine tool, you can limit the movement to those axes by adding each axis letter followed by a zero (eg. "X0") after the command. Then, Techno moves only the indicated axes to the reference point coordinates. A typical use is to raise only the Z axis for a manual tool change ("G28 Z0").

Example (milling application):

Reference point 1 is defined as machine coordinate X=1, Y=1, Z=-1. Reference point 2 is defined as machine coordinate X=2, Y=2, Z=-2.

G01 X1.5 Y2.5 Z-3 F10	<i>Linear move to the program coordinates X=1.5, Y=2.5, Z=-3</i>
G28 Z0	<i>Rapid move in the Z axis only to machine coordinate Z=-1</i>
G01 X1.5 Y2.5 Z-3 F10	<i>Linear move to the program coordinates X=1.5, Y=2.5, Z=-3</i>
G28	<i>Rapid move in the Z axis to machine coordinate Z=-1 followed by a rapid move in the XY plane to machine coordinates X=1, Y=1</i>
G01 X1.5 Y2.5 Z-3 F10	<i>Linear move to the program coordinates X=1.5, Y=2.5, Z=-3</i>
G28 X0 Y0 Z0	<i>Rapid move in the Z axis to machine coordinate Z=-1 followed by a rapid move in the XY plane to machine coordinates X=1, Y=1 (identical to the G28 command with no parameters specified)</i>
G30 P2	<i>Rapid move in the XY plane to machine coordinates X=2, Y=2 followed by a rapid move in the Z axis to machine coordinate Z=-2</i>

G29, G29.1 Return from Reference Point

The G29 command moves the tool to the designated program coordinate at the rapid rate. The G29.1 command moves the tool to the designated program coordinate at the specified feedrate.

The sequence of axis motions follows a general-purpose scheme that's described under "Point Control Panel" in the Main Screen Features section of this manual.

Example:

Reference point 1 is defined as machine coordinate X=1, Y=1, Z=-1.

G01 X1.5 Y2.5 Z-3 F10	<i>Linear move to program coordinates X=1.5, Y=2.5, Z=-3</i>
G28	<i>Rapid move in the Z axis to machine coordinate Z=-1 followed by a rapid move in the XY plane to machine coordinates X=1, Y=1</i>
G29 X2 Y3 Z-2	<i>Rapid move in the XY plane to program coordinates X=2, Y=3 followed by a rapid move in the Z axis to program coordinate Z=-2</i>
G29.1 X4 Y5 Z-3 F10	<i>Feedrate move in the XY plane to program coordinates X=4, Y=5 followed by a feedrate move in the Z axis to program coordinate Z=-3</i>

When using G29, there are several things to keep in mind:

- You do not need to specify coordinates for all machine tool axes, only the ones for which you want movement.

Example:

```
G29 X4.0 Y3.0
```

Moves the tool to program coordinates X=4.0, Y=3.0, leaving the Z position unchanged

- The interpretation of the coordinates depends on the G90/G91 command in effect.

G31 Seek Sensor

G31 tells Techno to move a single axis until a sensor is activated, similar to tool length sensing and homing. The syntax is:

```
G31 Xx [Yy] [Zz] [Aa] [Bb] [Cc] Ii Ss Ee Ff
```

where the parameters are

x, y, z, a, b, or c:

Endpoint for the move (only one axis may be used)

i: Input line to monitor

s: Switch state that ends the move (0 = untripped, 1 = tripped)

e: Coordinate system for the endpoint (0 = program coordinates, 1 = machine coordinates)

f: Feedrate

Example:

```
G31 Z-2.0 I4 S1 E0 F5
```

Moves the Z axis to program coordinate -2.0 at 5 inches/minute, stopping when input line 4 becomes tripped

A common use for the G31 command is to adjust a tool offset, not because the length of the tool has changed, but to allow the system to adjust its motion to match an irregularly shaped workpiece. For example, a milling machine can compensate all Z motion to match a workpiece with an unknown height, by sensing the top surface and adjusting the Z offset for the current tool.

To set a tool offset using G-Code, set the system variable #Tool as shown in these two examples:

```
#Tool{1}.z = #Machine.Z
```

Sets the Z offset for tool 1 to the current Z machine coordinate

```
#Tool{3}.x = #Machine.X + 2
```

Sets the X offset for tool 3 to the current X machine coordinate plus 2

For more details on using variables, see “Advanced Programming Reference” later in this section.

Note: In previous versions of Techno a ‘C’ parameter was used for the coordinate system instead of the ‘E’ parameter. If your machine tool does not have a C axis, Techno still allows the ‘C’ parameter for the coordinate system in the G31 command.

G40, G41, G42 Cutter Compensation

Cutter compensation allows Techno to automatically adjust the toolpath to account for the radius of the cutting device (tool, torch, or other device). This provides the following benefits:

- You can write your G-Code program to reflect the dimensions of the part, rather than calculating where the centerline of the tool will move.
- You can easily compensate for tool wear, when the exact diameter of the tool is not known when the G-Code file is created.
- You can precisely control the final dimensions of a part, by artificially changing the tool diameter to reposition the toolpath.

The commands are defined as:

- G41 Compensate to the left of the programmed path
- G42 Compensate to the right of the programmed path
- G40 Cancel cutter compensation

The format for the G41/42 command is

G41 Dn (or G42 Dn)

where n is the tool number for the current tool. Techno offsets the toolpath by half the diameter you’ve entered for the tool (i.e. the radius) on the Tool Library panel of the Configuration dialog box.

Prior to the first G41/42 command, you should indicate the current tool using the M06 command (e.g. M06 T1 to select tool 1). If you leave this out, Techno will warn you that the tool number in the G41/42 command does not match the current tool (but the file will still run).

The G40 command (cancel cutter compensation) requires no parameters. Note that G54-59 and G92 also cancel cutter compensation.

You’ll activate cutter compensation just before cutting a feature (profile, pocket, etc.) then cancel cutter compensation immediately upon completing the feature. The typical sequence of commands is –

1. Select tool with M06 command (e.g. M06 T1)
2. Move Z axis up to safe height above workpiece
3. Activate cutter compensation using G41 or G42
4. Move X and Y axes together (G00 or G01) to a point on the feature (called the ‘approach’ move)
5. Move Z axis down into the workpiece
6. Do any number of XY moves (lines and arcs) to cut the feature, all at the same Z axis depth
7. Move Z axis up to safe height above workpiece
8. Cancel cutter compensation using G40
9. Repeat steps 3-8 for all features

Be sure to follow these steps for each distinct feature of the part. Cutter compensation should be turned off whenever the machine is not actually cutting a feature (e.g. when returning to Machine Zero or the tool change position).

For information on customizing cutter compensation, please see “Cutter Compensation Setup” in the Initial Setup section of this manual.

The following example demonstrates use of cutter compensation.

Example:

This G-Code file cuts on the inside of two 1-inch squares.

F10.00	<i>Set feedrate</i>
M06 T1	<i>Select tool 1</i>
G00 Z 0.25	<i>Raise Z to safe height</i>
G41 D1	<i>Activate cutter comp left</i>
G00 X 1.00 Y 1.00	<i>Approach move</i>
G01 Z-0.10	<i>Plunge into workpiece</i>
G01 X 2.00 Y 1.00	<i>Cut first side of square</i>
G01 X 2.00 Y 2.00	
G01 X 1.00 Y 2.00	
G01 X 1.00 Y 1.00	<i>Cut last side of square</i>
G00 Z 0.25	<i>Raise Z to safe height</i>
G40	<i>Cancel cutter compensation</i>
G41 D1	<i>Activate cutter comp left</i>
G00 X 3.00 Y 1.00	<i>Approach move</i>
G01 Z-0.10	<i>Plunge into workpiece</i>

```
G01 X 4.00 Y 1.00           Cut first side of square
G01 X 4.00 Y 2.00
G01 X 3.00 Y 2.00
G01 X 3.00 Y 1.00         Cut last side of square
G00 Z 0.25                Raise Z to safe height
G40                       Turn off cutter compensation
G00 X 0.00 Y 0.00        Return to program zero
```

G50, G51 Scaling

The G51 command lets you scale all motion about program zero or any other point you choose.

The simplest syntax is:

G51 Pp

where 'p' is the scale for all axes.

Example:

```
G51 P1.5                    Scales motion on all axes by a factor of 1.5,
                           relative to program zero
```

You can also set the scaling per axis using this syntax:

G51 Ii Jj Kk

where the parameters are

i: X axis scale

j: Y axis scale

k: Z axis scale

Example:

```
G51 I2 J2 K1                Scales motion on the X and Y axes by a factor of
                           2, relative to program zero. Z axis motion is not
                           scaled.
```

Finally, instead of scaling motion about program zero, you can specify the scaling origin using the following syntax:

G51 Xx Yy Zz Ii Jj Kk

where the additional parameters are

x: X axis scaling origin

y: Y axis scaling origin

z: Z axis scaling origin

Example:

```
G51 X3 Y4 Z-1 I2 J2 K0.5
```

Scales motion on the X, Y and Z axes about the point X=3, Y=4, Z=-1

The G50 command turns scaling off.

Example:

```
G51 P4
G01 X1 Y1 F5
```

This move is scaled

```
G50
G01 X2 Y2
```

This move is not scaled

G52 Local Coordinate System

The G52 command activates a local coordinate system that Techno uses in place of your original program coordinates for all absolute positioning moves. The X, Y, Z, A, B and C parameters indicate the offset from your Program Zero location to the origin for the local coordinate system.

For example, "G52 X1 Y2 Z-4" would activate a local coordinate system whose origin is at a distance of 1, 2, -4 from the original Program Zero.

All absolute moves are made relative to the new local coordinate system. To cancel use of the local coordinate system in the middle of a G-code file, use the command "G52 X0 Y0 Z0 A0 B0" (using only the letters for axes used on your machine).

When Techno reads a G52 command, it displays a magenta dot in the Viewport Box showing the origin of the local coordinate system.

Note that the local coordinate system only applies to the G-code file being executed. The G52 command has no effect the Program Zero location. Techno automatically cancels the local coordinate system when it completes execution of a G-code file.

If you reposition Program Zero using the G54-59 or G92 commands, the G52 command remains in effect, with local coordinate system now offset from the new Program Zero location.

Example:

```
G01 X1.0 Y3.0 Z-1.5 F12
```

Moves the tool to program coordinates X=1.0, Y=3.0, Z=-1.5

```
G52 X3 Y-7 Z0
```

Activates a local coordinate system with origin at X=3, Y=-7, Z=0 relative to Program Zero (the machine tool does not move)

```
G01 X1.0 Y10.0 Z2.0
```

Moves the tool to the point X=1.0, Y=10.0, Z=2.0 relative to the local coordinate system as defined by the G52 command above

G52 X0 Y0 Z0

Cancels use of the local coordinate system. All absolute moves are again relative to Program Zero as you set it before running the program

G53, G53.1 Linear Move to Machine Coordinates

The G53 and G53.1 commands move the tool to the designated Machine coordinate using 3-axis linear interpolation. The G53 command causes a rapid move, while the G53.1 command causes a feedrate move. These commands are identical to the G00 and G01 commands except for the following:

- The destination is specified in Machine coordinates.
- The commands are not modal. You must include the G53 or G53.1 command on any G-Code line where you want motion to a Machine coordinate.
- You may include an optional 'D' parameter to limit each axis to one direction of motion, as follows:

D1: Motion must be positive. Any axis that would move in a negative direction doesn't move.

D-1: Motion must be negative. Any axis that would move in a positive direction doesn't move.

A typical use for the 'D' parameter is to suppress motion in an unwanted direction, when the move is part of a subroutine or M-Code macro (e.g. for automatic tool changing).

Example:

G53.1 X2 Y3 Z-4 F12

Moves the tool to machine coordinates X=2.0, Y=3.0, Z=-4.0 at a feedrate of 12

G53 X3 Y4 Z-5 D1

Moves the tool to machine coordinates X=3.0, Y=4.0, Z=-4.0 at the rapid rate (Z axis motion is suppressed since it would be in the negative direction)

For more information, please see "G00 Rapid Tool Positioning" and "G01 Linear Interpolated Feedrate Move" above.

G54-59, G54.1, G92 Set Program Zero Commands

The G54-59, G54.1 and G92 commands reset the Program Zero location.

The G54-59 and G54.1 commands set Program Zero to a predefined offset away from Machine Zero. They are particularly useful when using a workpiece fixture; the offset from Machine Zero to the fixture can be set once, and then used over and over. These commands use the offsets defined on the Fixture Offsets panel of the Configuration dialog box.

Example:

If the offset for G54 is set to

X = 1

Y = 2

Z = -3

A = 90

on the Fixture Offsets panel, the G54 command would place Program Zero at machine coordinate X=1, Y=2, Z=-3, A=90. The result is the same as if you moved the machine tool to machine coordinate X=1, Y=2, Z=-3, A=90, then clicked the Zero button for program coordinates. The G54-59 and G54.1 commands require Machine Zero to be set.

The G54.1 command allows access to 100 fixture offsets using the following syntax:

G54.1 Pn

where 'n' is any number 1 through 100. This command raises the total number of available fixture offsets to 106 (6 for G54-59, plus 100).

The G92 command is similar to the G54-59 and G54.1 commands, except *the new program coordinates for the current location* are specified in the command itself. The command syntax is:

G92 [Xx] [Yy] [Zz] [Aa] [Bb] [Cc]

where 'x', 'y', 'z', 'a', 'b' and 'c' specify the new current program coordinates. These parameters are optional but you must include at least one.

Example:

G92 X1 Y2 Z-3 A90

sets the current program coordinates to X=1, Y=2, Z=-3, A=90. *The new offset from Machine Zero to Program Zero is not specified, so its value depends on where the machine happens to be positioned.* For this reason, the 54-59 and G54.1 commands are generally preferred to the G92 command, since they use a more reliable method for setting the Program Zero location.

For the G54-59, G54.1 and G92 commands, the Program Zero that's established remains in effect after the G-Code file has finished execution. The function of these commands is identical to that of the Program Coordinates Zero and Set buttons on the main screen.

M06 Tool Change and T Select Tool Commands

To indicate tool changes in the G-Code file, use the M06 and T commands as follows:

```
M06 Tn
```

where n is the tool number on the Tool Library panel of the Configuration dialog box.

Example:

```
M06 T3
```

The T command can be used on any line prior to the M06 command; it does not need to be on the same line as M06.

Techno handles the M06 command according to the settings on the Tool Change panel of the Configuration dialog box. See “Tool Change” in the Initial Setup section for details.

The M06 command does not move the machine tool to the tool change position. This is done using the G28 command described above. If you’re changing tools manually, it’s good practice to place the G28 command (and the M05 spindle off command) on the lines directly preceding the M06 command.

For compatibility with Fanuc G-Code for lathes, Techno also supports the Fanuc-style lathe tool change command. The command is:

```
Txxyy
```

where the parameters are

xx: tool number

yy: wear offset number

When Techno encounters a T command formatted this way, it does the following:

1. Performs a tool change to tool number xx
2. Applies the geometry offset for tool number xx

Example:

```
T0404
```

Changes to tool #4 and applies the geometry offset for tool #4 (the ending “04” is ignored)

G43, G44, G49 Tool Length/Geometry Compensation Commands

Tool length/geometry compensation lets Techno account for differences in tool lengths and geometry when the G-Code file is executed. This lets you create a G-Code program without knowing the exact geometry of the tools that will be used to cut the part.

In milling mode, compensation is generally used to account for differences in tool lengths in the Z direction. Generally, X and Y offsets are not used. However, to accommodate a milling-style machine with multiple spindles, the offset between spindles can be handled using the X and Y components of the tool offset.

In lathe mode, compensation is generally used to account for differences in tool tip positions, in both X and Z directions.

THESE COMMANDS ARE NOT FOR THE NOVICE CNC USER. WHEN NOT PROPERLY USED, TOOL LENGTH AND GEOMETRY COMPENSATION CAN CRASH THE MACHINE TOOL, CAUSING SERIOUS DAMAGE TO YOUR WORKPIECE OR MACHINE TOOL.

When applying tool length and geometry compensation, Techno uses the offsets defined on the Tool Library panel of the Configuration dialog box. See “Tooling Settings” in the Initial Setup section for more information on defining your tool offsets.

For tool length and geometry compensation to work properly, Techno must know what tool is in use at all times. Therefore, you must indicate every tool change with an M06 command in the G-Code file. Similarly, whenever you change a tool between running G-Code files, you should select it in the Current Tool pull-down menu. The Compensation Mode pull-down menu is set to Positive (G43) by default.

Technically, the M06 command indicating your starting tool is optional, but we recommend you include it so the file is self-contained and less prone to operator error. However, if you choose not to include an M06 command for your starting tool, you must make certain you choose your starting tool from the Current Tool pull-down menu on the main screen, before running the file.

Once the M06 command has set the current tool, the G43 command applies the proper offset to account for the current tool's length or geometry as follows:

G43 Hn

where n is the tool number for the current tool.

Milling Mode (Tool Length Compensation) – The G43 command tells Techno to shift all subsequent Z axis moves away from the workpiece (in the positive Z direction) by an amount equal to the Z offset for tool ‘n’.

Lathe Mode (Tool Geometry Compensation) – The G43 command tells Techno to shift all subsequent moves away from the workpiece (in the positive X and Z directions) by an amount equal to the X and Z offset for tool ‘n’.

If tool length/geometry compensation is already in effect for a previous tool, the resulting shift away from the workpiece equals the difference between offsets for tool ‘n’ and the previous tool.

Milling Mode Example:

G43 H3

Shifts all subsequent Z axis moves away from the workpiece (in the positive Z direction) by the Z offset for Tool #3

Lathe Mode Example:

G43 H3

Shifts all subsequent X and Z axis moves away from the workpiece (in the positive X and Z directions) by the X and Z offset for Tool #3

The G44 command is identical to the G43 command, except that it shifts all moves in the direction opposite from G43. Unless you are an experienced CNC programmer and know how to use G44 correctly, G43 is the preferred command.

The G49 command cancels tool length/geometry compensation.

When using tool length compensation there are several important things to keep in mind:

- You must predefine all tools and associated offsets on the Tool Library panel of the Configuration dialog box.
- It's good practice to include an M06 tool change and a G43 compensation command for the first tool used, near the beginning of the G-Code program.
- Techno does not cancel tool offset when it finishes processing a G-Code file. **TO AVOID CRASHES, IT IS VERY IMPORTANT THAT YOU UPDATE THE CURRENT TOOL DROP-DOWN MENU, WHENEVER YOU MANUALLY CHANGE A TOOL.**
- The G43, G44 and G49 commands are modal, so the current tool offset remains active until Techno executes another one of these three commands, or you set the Compensation Mode pull-down menu.
- It's good practice to use the G43 command in the line directly following the M06 command.

The following example illustrates proper use of the tool change and tool length compensation commands.

Milling Mode Example:

There are two basic methods for setting Program Zero, and both will be covered in this example.

- Use G54-G59 to apply a fixture offset ("Fixture Offset Method"). This method requires Machine Zero and the correct fixture offset to be set.
- Touch the tip of the tool to the top of the workpiece and click the Program Coordinates Zero button on the main screen ("Tool Touch-Off Method"). This method doesn't require Machine Zero or a fixture offset to be set.

In this example, the tool change position (reference point 1 on the Reference Points panel of the Configuration dialog box) is defined as machine coordinates X=0, Y=0, Z=0 (Machine Zero). Tool #1 is loaded in the machine tool, and selected in the Current Tool pull-down menu on the main screen. The Compensation Mode pull-down menu is set to Positive (G43). Machine Zero has been set. The machine tool is at Machine Zero. The tooling configuration is shown below.

Tool #	Description	Diameter (inches)	X Offset (inches)	Y Offset (inches)	Z Offset (inches)
1	1/4" End Mill	0.2500	0.0000	0.0000	1.5000
2	1/8" Ball End Mill	0.1250	0.0000	0.0000	2.2500
3		0.0000	0.0000	0.0000	0.0000
4		0.0000	0.0000	0.0000	0.0000
5		0.0000	0.0000	0.0000	0.0000
6		0.0000	0.0000	0.0000	0.0000
7		0.0000	0.0000	0.0000	0.0000
8		0.0000	0.0000	0.0000	0.0000

Program Zero is set as follows:

Tool Touch-Off Method:

Program zero is set at machine coordinate X=1, Y=1, Z=-4, with the tip of Tool #1 touching the top surface of the workpiece.

Fixture Offset Method:

Program zero is temporarily set at Machine Zero. The fixture offset for G54 is set to X=1, Y=1, Z=-5.5, A=0 on the Fixture Offsets panel of the Configuration dialog box. The fixture offset is the machine coordinate when the spindle nose touches the top surface of the workpiece. When the G54 command in the G-Code file is executed, Program Zero will be set at its associated offset value.

The G-Code file is as follows:

G54	<i>Sets Program Zero (Fixture Offset Method Only). Program coordinates change to X=-1, Y=-1, Z=4</i>
G28	<i>The machine is already at the tool change position so nothing happens</i>
M05	<i>Turns off the spindle</i>
M06 T1	<i>Changes the tool to Tool #1, which is already loaded</i>
G43 H1	<i>Applies tool length compensation for Tool #1. Since Tool #1 was already loaded, and Compensation Mode was set to Positive (G43) when the program started, this command doesn't</i>

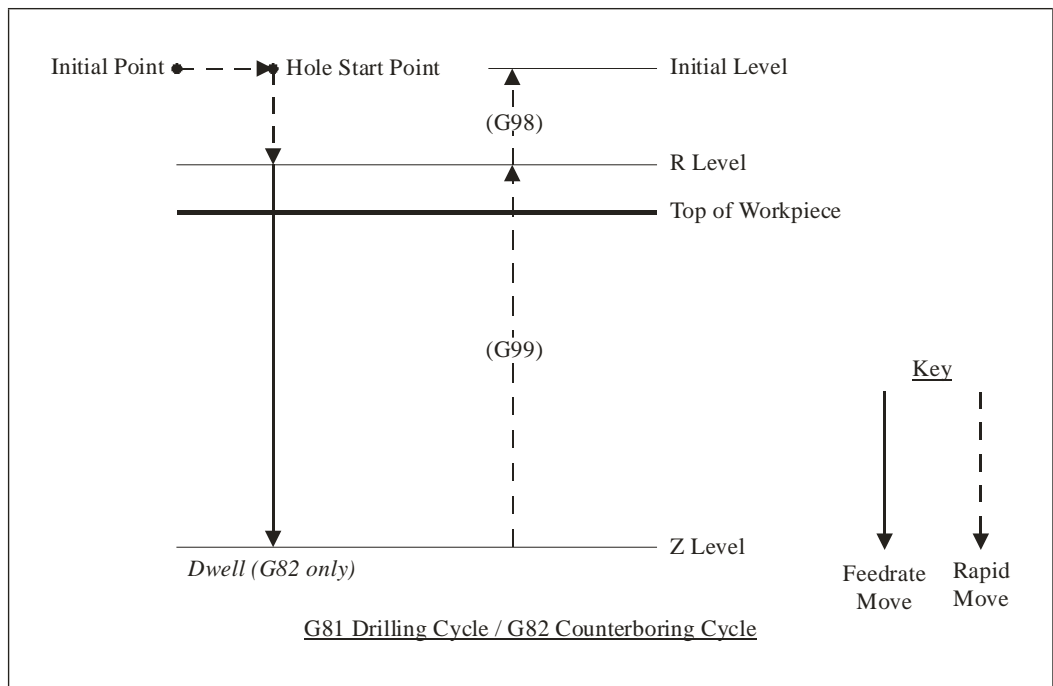
	<i>do anything. Program coordinates and machine coordinates remain unchanged</i>
M03	<i>Turns on the spindle</i>
G29 X0 Y0 Z.25	<i>Moves the X and Y axes across, then the Z axis down to program coordinates X=0, Y=0, Z=0.25, machine coordinates X=1, Y=1, Z=-3.75</i>
G01 Z-1.0 F10	<i>Moves the Z axis down to program coordinates X=0, Y=0, Z=-1, machine coordinates X=1, Y=1, Z=-5</i>
G01 X3 Y3 Z-1 F20	<i>Linear interpolation to program coordinates X=3, Y=3, Z=-1, machine coordinates X=4, Y=4, Z=-5</i>
G28	<i>Moves the Z axis up, then the X and Y axes across to the tool change position, program coordinates X=-1, Y=-1, Z=4, machine coordinates X=0, Y=0, Z=0</i>
M05	<i>Turns off the spindle</i>
M06 T2	<i>Changes the tool to Tool #2</i>
G43 H2	<i>Applies tool length compensation for Tool #2. The Z program coordinate shifts by -0.750 (1.500 - 2.250). The program coordinates are now X=-1, Y=-1, Z=3.25. The machine coordinates remain unchanged at X=0, Y=0, Z=0</i>
M03	<i>Turns on the spindle</i>
G29 X0 Y0 Z0.25	<i>Moves the X and Y axes across, then the Z axis down to program coordinates X=0, Y=0, Z=0.25, machine coordinates X=1, Y=1, Z=-3.0</i>
G01 Z-1.0 F10	<i>Moves the Z axis down to program coordinates X=0, Y=0, Z=-1, machine coordinates X=1, Y=1, Z=-4.25</i>
G01 X3 Y3 Z-1 F20	<i>Linear interpolation to program coordinates X=3, Y=3, Z=-1, machine coordinates X=4, Y=4, Z=-4.25</i>
G28	<i>Moves the Z axis up, then the X and Y axes across to the tool change position, program coordinates X=-1, Y=-1, Z=3.25, machine coordinates X=0, Y=0, Z=0</i>
M05	<i>Turns off the spindle</i>

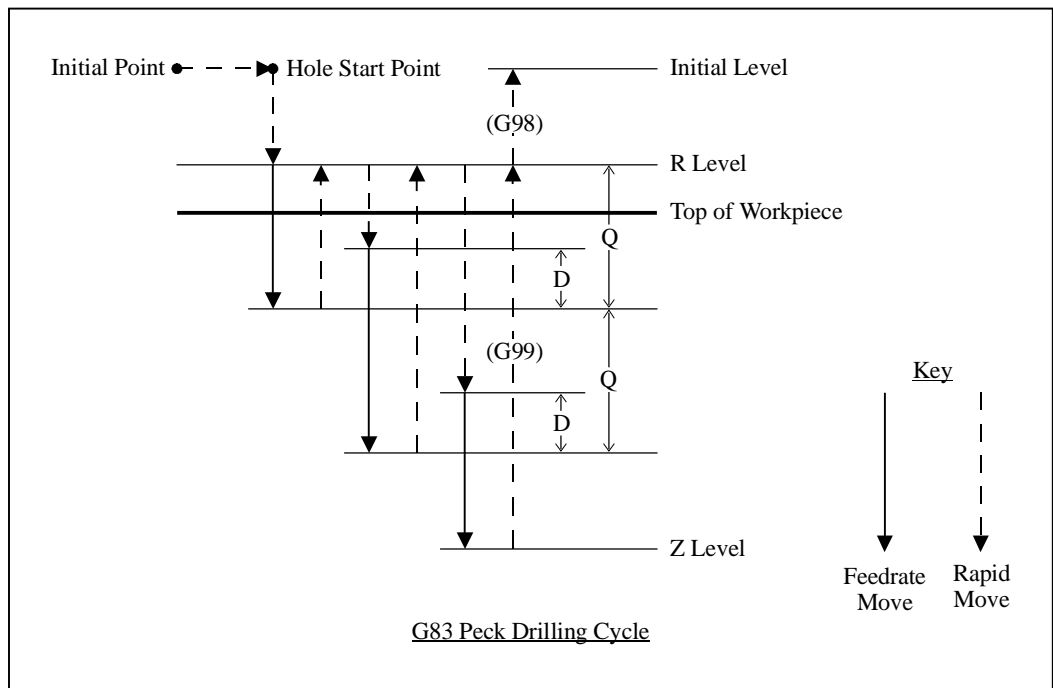
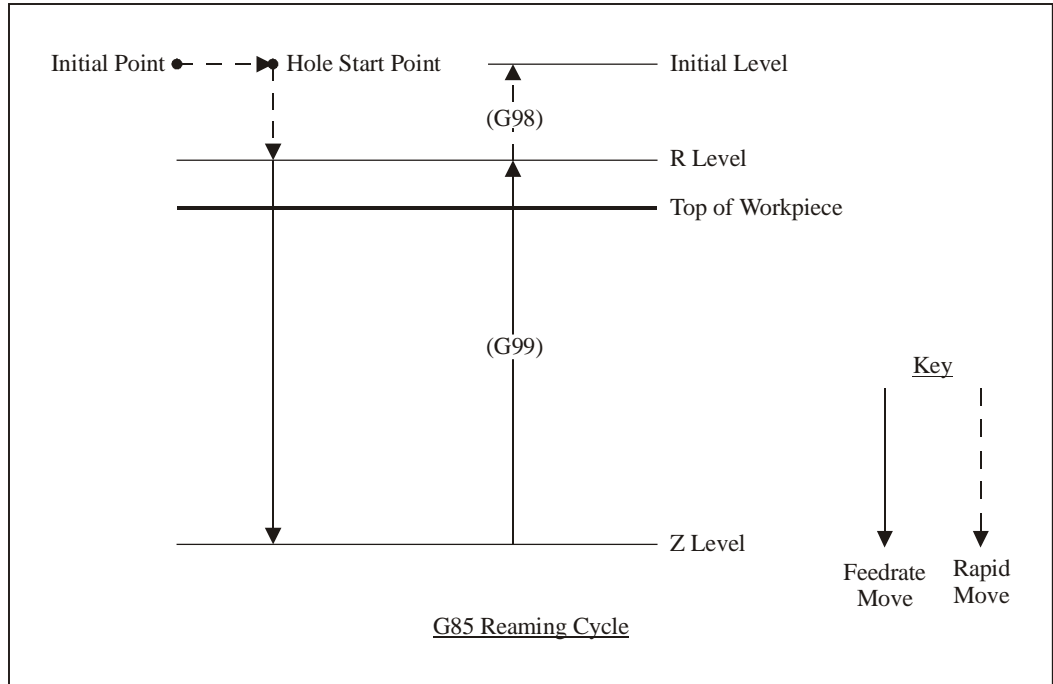
G73, G80, G81, G82, G83, G85, G98, G99 Drilling Canned Cycle Commands

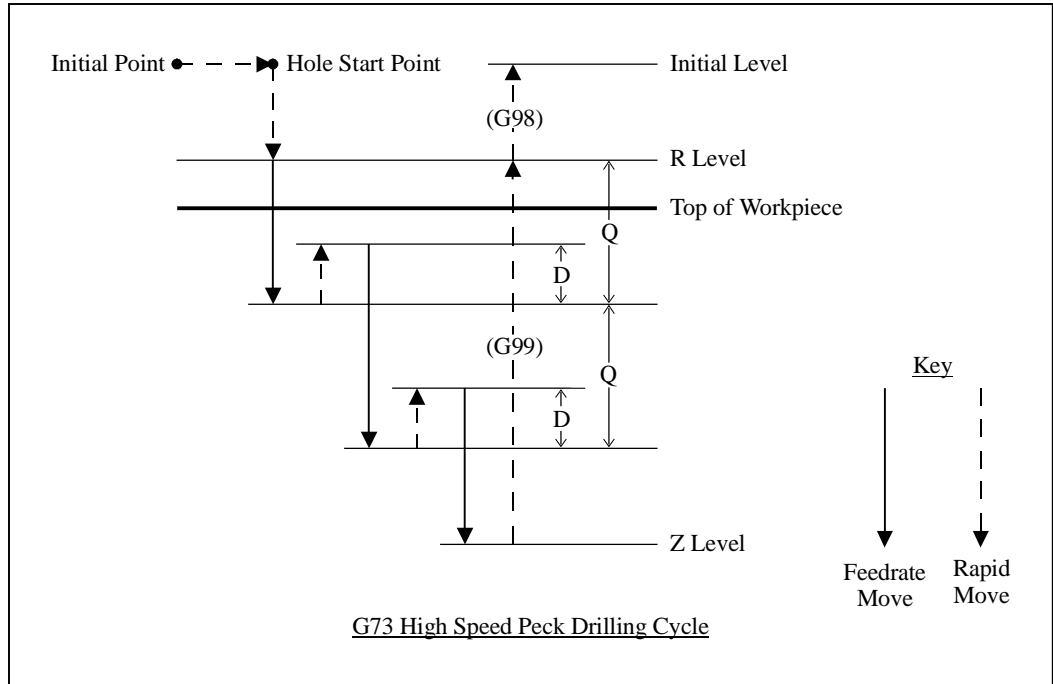
Drilling canned cycles simplify programming by letting you specify standard drilling operations with a single command. Techno supports the following drilling cycle commands:

- G73 High Speed Peck Drilling Cycle
- G80 Cancel Canned Cycle
- G81 Drilling Cycle
- G82 Counterboring Cycle
- G83 Peck Drilling Cycle
- G85 Reaming Cycle

The following diagrams illustrate the sequence of moves for all drilling cycles.







The general command syntax for all drilling cycles is as follows:

$$G_n X_x Y_y R_r Z_z Q_q P_p F_f L_l$$

where the parameters are

- n: Drilling cycle number (73, 81, 82, 83 or 85)
- x: X coordinate of Hole Start Point (G90 mode) or distance for rapid move from Initial Point to Hole Start Point (G91 mode)
- y: Y coordinate of Hole Start Point (G90 mode) or distance for rapid move from Initial Point to Hole Start Point (G91 mode)
- r: Z coordinate of R Level (G90 mode) or distance for rapid move from Hole Start Point to R Level (G91 mode)
- z: Z coordinate of Z Level (G90 mode) or distance from R Level to Z Level (G91 mode)
- q: Depth of cut for each downward cutting move (G73/G83 only)
- p: Dwell time at Z level (milliseconds, G82 only)
- f: Feedrate for each downward cutting move (and retract move for G85 only)
- l: Number of times to repeat the canned cycle (G91 mode only)

The required parameters are R, Z, F (plus Q for G73/G83, and P for G82).

The D distance is specified in the G73/G83 Retract Distance field on the G-Code panel of the Configuration dialog box.

Example (assumes D is 0.050 in.):

G98

Optional

G00 X0.0 Y0.0 Z1.0

Moves the tool to the Initial Point, program coordinate X=0.0, Y=0.0, Z=1.0 (not required, for illustration only)

G83 X1.0 Y2.0 R0.1 Z-1.0 Q0.5 F8.0

First moves the tool to the Hole Start Point, program coordinates X=1.0, Y=2.0, Z=1.0. Then peck drills a 1.0" deep hole (rapid down to 0.1, feedrate down to -0.4, rapid up to 0.1, rapid down to -0.350, feedrate down to -0.9, rapid up to 0.1, rapid down to -0.850, feedrate down to -1.0, rapid up to 1.0)

When using the drilling cycle commands, there are several things to keep in mind:

- For all drilling canned cycles, the G98 and G99 commands affect the final Z axis rapid move, upward out of the completed hole (rightmost move in each diagram), as follows:

G98: Z axis moves up to the Initial Level

G99: Z axis moves up to the R Level

G98 and G99 compose a mode group, with G98 being the default if neither is specified in the G-Code file. You can include either command on the same line as the canned cycle command (before or after the canned cycle command and all parameters).

- G73, G74, G76, G80, G81, G82, G83, G84 and G85 compose a mode group.
- G80 cancels the active canned cycle command. The G00, G01, G02 and G03 commands also cancel the active canned cycle command.
- While a drilling cycle command is active, the command and most of its parameters do not need to be repeated on every G-Code line. However, the X and/or Y positioning parameters must be included, and must start the G-Code line for the line to be interpreted as a drilling cycle command.

Example (drills 3 holes in a row along the X axis using the G81 cycle):

G81 X1.0 Y2.0 R0.25 Z-0.5 F5

X2.0

X3.0

- When you switch to a new command in the mode group, you must include all parameters required for the command.

- The feedrate set by the F parameter for any canned cycle command remains as the current feedrate for subsequent G01, G02 or G03 moves, unless you explicitly set F to another value for the next move.

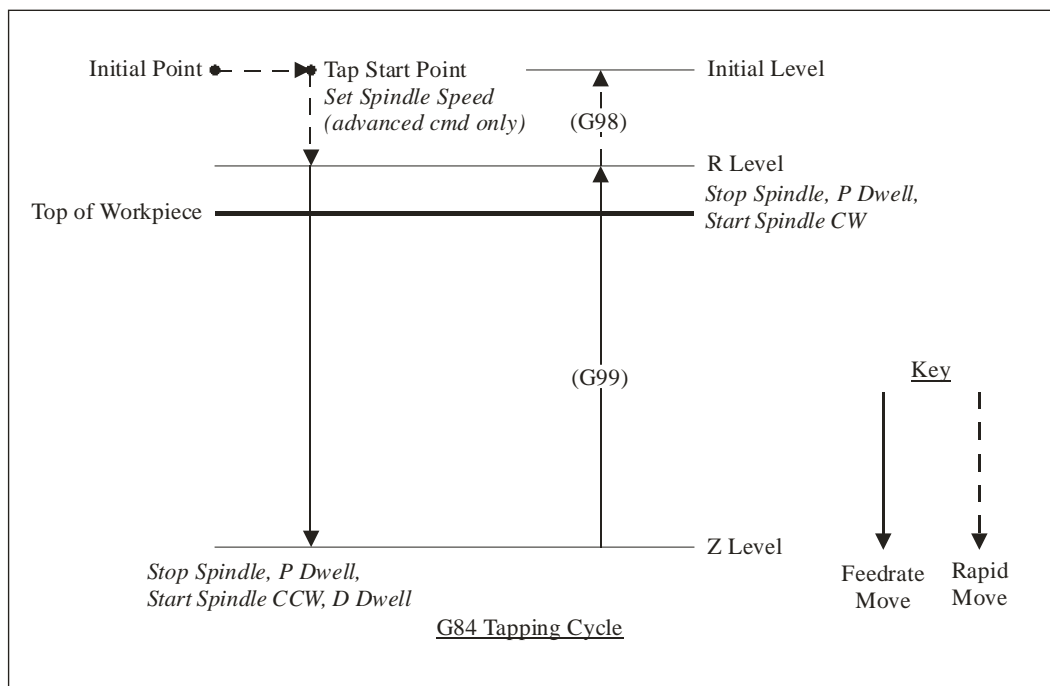
G74, G84 Tapping Canned Cycle Commands

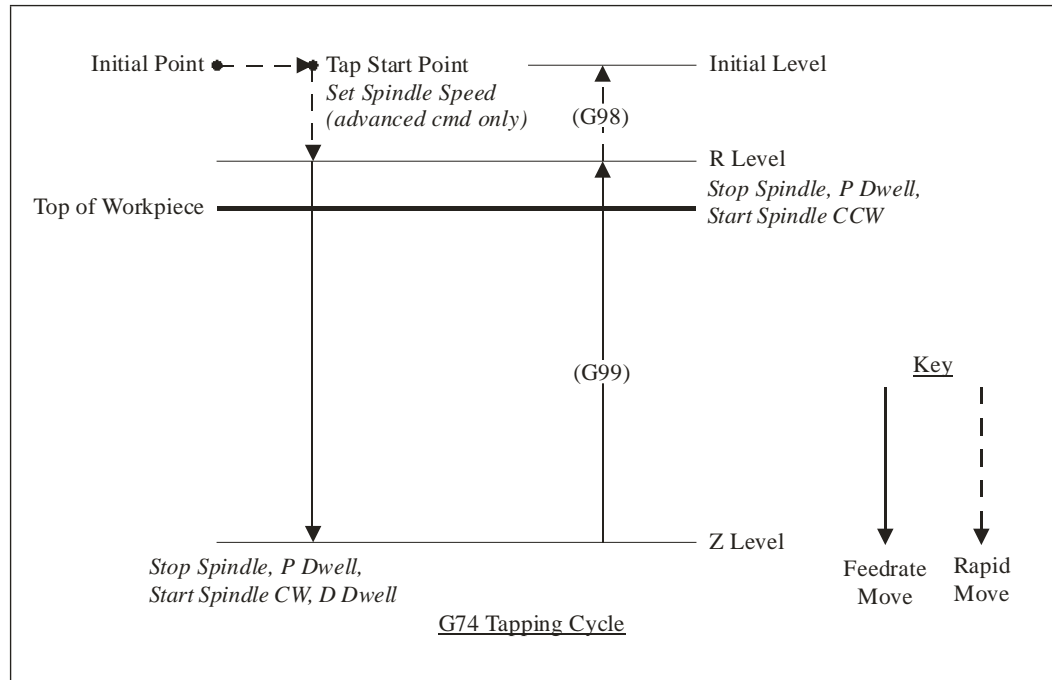
Tapping canned cycles simplify programming by letting you specify standard tapping operations with a single command. Techno supports the following tapping cycle commands:

G84 Standard Tapping Cycle

G74 Left-Handed Tapping Cycle

The following diagrams illustrate the sequence of moves for the tapping cycles.





There are two versions of the tapping cycle commands: Advanced and Fanuc-compatible. We recommend using the advanced command as the Fanuc command is provided for compatibility purposes only.

Advanced Tapping Cycle

Gn Xx Yy Rr Zz Qq Ss Ee Pp Dd Ll

where the parameters are

- n: Tapping cycle number (84 for standard tapping or 74 for left-handed tapping)
- x: X coordinate of Tap Start Point (G90 mode) or distance for rapid move from Initial Point to Tap Start Point (G91 mode)
- y: Y coordinate of Tap Start Point (G90 mode) or distance for rapid move from Initial Point to Tap Start Point (G91 mode)
- r: Z coordinate of R Level (G90 mode) or distance for rapid move from Tap Start Point to R Level (G91 mode)
- z: Z coordinate of Z Level (G90 mode) or distance from R Level to Z Level (G91 mode)
- q: Thread pitch
- s: Spindle speed (rpm)
- e: % increase in feedrate for the retract move (eg. E2 means the Z axis will retract 2% faster than it plunged into the hole)

- p: Dwell time after spindle is stopped (milliseconds)
- d: Dwell time after spindle is reversed (milliseconds)
- l: Number of times to repeat the canned cycle (G91 mode only)

The required parameters are R, Z, Q and S.

Fanuc-Compatible Tapping Cycle

Gn Xx Yy Rr Zz Ff Pp Ll

where the parameters are

- n: Tapping cycle number (84 for standard tapping or 74 for left-handed tapping)
- x: X coordinate of Tap Start Point (G90 mode) or distance for rapid move from Initial Point to Tap Start Point (G91 mode)
- y: Y coordinate of Tap Start Point (G90 mode) or distance for rapid move from Initial Point to Tap Start Point (G91 mode)
- r: Z coordinate of R Level (G90 mode) or distance for rapid move from Tap Start Point to R Level (G91 mode)
- z: Z coordinate of Z Level (G90 mode) or distance from R Level to Z Level (G91 mode)
- f: Feedrate (must be calculated based on spindle speed previously set by S command, and desired thread pitch)
- p: Dwell time after spindle is stopped (milliseconds)
- d: Dwell time after spindle is reversed (milliseconds)
- l: Number of times to repeat the canned cycle (G91 mode only)

The required parameters are R, Z, and F.

Before using the tapping cycles you must configure the system for tapping as follows:

1. Enter M-Codes for controlling the spindle on the G-Code panel of the Configuration dialog box as follows:
 - Turn the spindle on clockwise (typically M3)
 - Turn the spindle on counterclockwise (typically M4)
 - Turn off the spindle (typically M5)
2. If you want to use the Fanuc-compatible tapping cycle, check Use Fanuc-Compatible G74/G84 Tapping Cycles on the G-Code panel of the Configuration dialog box.

WARNING: The Feed Hold button (and feed hold input lines) are disabled during a tapping cycle, since it is not safe to stop moving the tool while the spindle is still turning. If you need to stop motion during a tapping cycle, turn off power to the spindle and Signal Generator.

When using the tapping cycle commands, there are several things to keep in mind:

- The spindle must be turned on before a tapping cycle command is executed (clockwise for G84, counterclockwise for G74).
- The G98 and G99 commands affect the final Z axis rapid move as follows:

G98: Z axis moves up to the Initial Level

G99: Z axis remains at the R Level (no rapid move)

G98 and G99 compose a mode group, with G98 being the default if neither is specified in the G-Code file. You can include either command on the same line as the canned cycle command (before or after the canned cycle command and all parameters).

- G73, G74, G76, G80, G81, G82, G83, G84 and G85 compose a mode group.
- G80 cancels the active canned cycle command. The G00, G01, G02 and G03 commands also cancel the active canned cycle command.
- While a tapping cycle command is active, the command and most of its parameters do not need to be repeated on every G-Code line. However, the X and/or Y positioning parameters must be included, and must start the G-Code line for the line to be interpreted as a tapping cycle command.

Example (taps 3 holes in a row along the X axis using the G84 cycle):

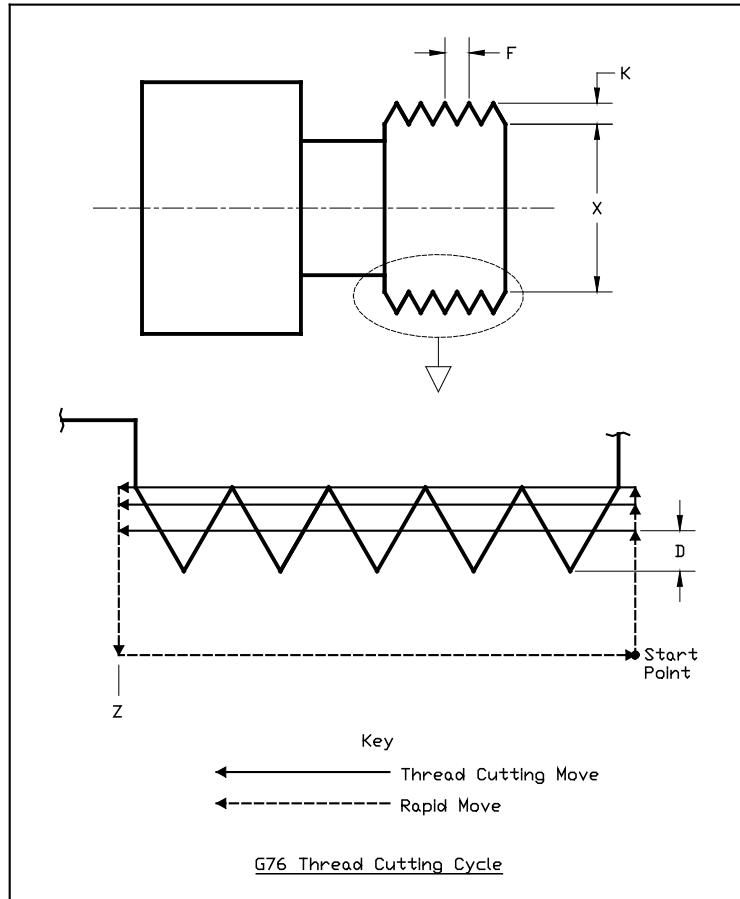
```
G84 X1.0 Y2.0 R0.25 Z-0.5 Q0.05 S200 P500
X2.0
X3.0
```

- When you switch to a new command in the mode group, you must include all parameters required for the command.
- The feedrate set by the F parameter for any canned cycle command remains as the current feedrate for subsequent G01, G02 or G03 moves, unless you explicitly set F to another value for the next move.

G76 Thread Cutting Canned Cycle Command

The thread cutting canned cycle lets you cut threads on a lathe. Your lathe must be equipped with a spindle encoder that is plugged into the I/O Expansion Board.

The following diagram illustrates the sequence of moves for the thread cutting cycle. The rest of this section refers to this diagram.



The general command syntax is as follows:

`G76 Xx Zz Kk Dd Ff [Aa] [Ii] [Pp]`

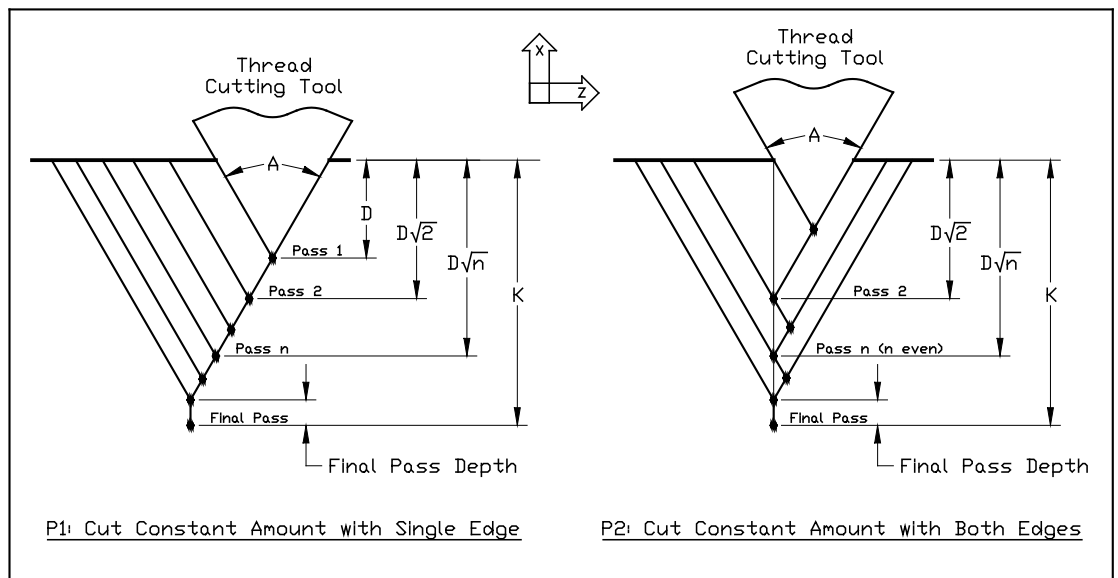
where the parameters are

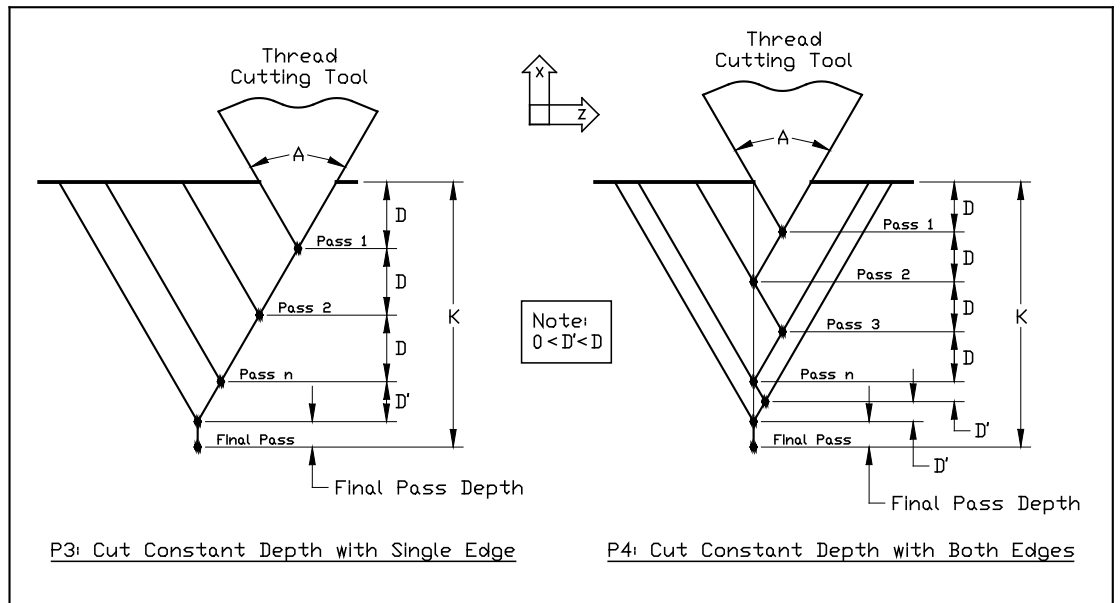
- x: Final diameter (minor diameter for external threads, major diameter for internal threads)
- z: Z Coordinate at end of thread
- k: Height of thread
- d: Depth of first pass specified as an integral number of ten-thousandths of an inch or mm, e.g. '2000' means 0.2000 inches or mm (depending on the units you're using)
- f: Thread pitch
- a: (Optional) Tool angle (determines lead-in angle, default is zero)

- i: (Optional) Amount of taper (difference between initial radius and ending radius, default is zero)
- p: (Optional) Cutting method (values 1-4, sets strategy for successive plunges, default is '1')

Note that in the diagram above, the 'A' and 'I' parameters are zero, and the cutting method is '1' (all default values).

Cutting a thread generally requires several passes. The 'P' parameter controls the depth, starting position and plunge angle of each pass, as shown in the following diagrams. For all four cutting methods shown in the diagrams below, the 'A' parameter is shown as nonzero to illustrate its affect. When 'A' is zero, the plunges are straight into the workpiece, along the X axis, as shown in the diagram above.





The settings on the Threading panel of the Configuration dialog box also affect the cutting passes as follows.

Minimum Depth of Cut – The incremental depth of any pass will be greater than or equal to this value (except for the final pass and any spring passes).

Final Pass Depth of Cut – Incremental depth of the final pass (finishing pass).

Spring Passes – Number of additional passes performed at the same depth as the final pass.

WARNING: The Feed Hold button (and feed hold input lines) are disabled during a threading pass, since it is not safe to stop moving the tool while the part is still turning. If you need to stop motion during a cutting pass, turn off power to the spindle and Signal Generator.

Example (assumes Minimum Depth of Cut is 0.010, Final Pass Depth of Cut is 0.005, Spring Passes is set to 1):

```
G00 X0.75
```

Move to the Start Point

```
G76 X1.0 Z-2 K0.125 D500 F0.2
```

Performs a total of 8 threading passes (including 1 spring pass) to make a two inch long thread with minor diameter 1.0, major diameter 1.25, at 5 turns per inch. By default the cutting method is P1, the tool angle is 0, and there is no taper. The first pass is cut at a depth of 0.05.

Note that the F parameter does not set the feedrate for subsequent non-threading moves, since its meaning is different from the standard F command.

G90 Absolute Positioning Mode

The G90 command puts the system into absolute positioning mode. All XYZABC coordinates are treated as points relative to Program Zero (or a local coordinate system set by the G52 command). This command stays in effect until a G91 command occurs.

Note that absolute positioning is the default positioning mode for Techno. It is not necessary to include this command in your G-code file if all your moves are absolute.

G91 Incremental Positioning Mode

The G91 command puts the system into incremental positioning mode. All XYZABC coordinates are treated as incremental move distances. This command stays in effect until a G90 command occurs.

Example:

G01 X1.0 Y3.0 Z-1.5 F12	<i>Moves the tool to program coordinates X=1.0, Y=3.0, Z=-1.5 (G90 is assumed)</i>
G91	<i>All XYZA coordinates after this command will be interpreted as incremental distances</i>
G01 X1.0 Y2.0 Z-0.5	<i>Moves the tool a distance of X=1.0, Y=2.0, Z=-0.5 from the current tool location (this corresponds to the program coordinates X=2.0, Y=5.0 Z=-2.0)</i>
G02 X1.0 Y-1.0 I0.5 J-0.5	<i>Moves the tool using counter-clockwise circular interpolation to the program coordinates X=3.0, Y=4.0, Z=-2.0 with a center point at program coordinates X=2.5, Y=4.5, Z=-2.0</i>
G90	<i>All XYZA coordinates after this command will be interpreted as program coordinates</i>
G01 X1.0 Y2.0 Z-0.5	<i>Moves the tool to program coordinates X=1.0, Y=2.0, Z=-0.5</i>

G120 Sense Tool Length

G120 runs the tool length sensing routine, the same as if you clicked the Length button in the Tool Box. There are no parameters.

Example:

G120	<i>Senses the length of the current tool and stores the result in the Tool Library</i>
------	--

See “Tool Box” in the Main Screen Features section for more information.

G140, G141 Engraving

Techno CNC supports engraving with the G140 and G141 commands.

G140 is used to select the font. The syntax is:

```
G140 "font" Hh [Kk] [Ss]
```

where the parameters are

font: The name of the font. Currently one font “SLF1” is built into Techno. Techno lets you create your own plug-in fonts. Please contact technical support for more information.

h: The height of the font

k: (Optional) The kerning, or spacing, between characters in the text to engrave (default is 1/10th of the height)

s: (Optional) The character width style to use (0 = fixed, 1 = variable, default is variable)

Example:

```
G140 "slf1" H1.5 K0.1 S1
```

The font specified in the G140 command must have been loaded by Techno CNC upon startup or an error will result.

The height parameter specifies the height of the tallest character in the font. All other characters are scaled as appropriate, and the width of each character is scaled so that the proper aspect ratio of the character is maintained.

Fixed character width is determined by the widest character in the font. For fixed character widths, each character is centered in a cell that is the width of the widest character. The kerning value, then, is the space between each cell. For variable character width, the cell for each character is as wide as the character itself.

The G140 command selects the font for all engraving (G141) commands that follow. The first G141 command in a G-code file must be preceded by a G140 command or an error will result.

The G141 command specifies the desired text to engrave. The syntax is:

```
G141 "text" ["text2"] [Xx] [Yy] [Zz] [Ww] Ff Pp Rr [Dd]
```

where the parameters are

text: The text to engrave

text2: (Optional) The text to engrave with a second spindle (see below)

x, y: (Optional) The XY position of the lower left corner of the text (program coordinates). If you leave out these parameters Techno will engrave the text at the current XY location. In G91 mode these parameters represent the offset from the current XY location to the lower left corner of the text.

z: (Z or W is required) The depth of the engraving by the Z spindle (Z program coordinate)

- w: (Z or W is required) The depth of the engraving by the W spindle (W program coordinate)
- f: The feedrate at which to engrave
- p: The feedrate at which to plunge the tool during engraving
- r: The height to which the tool is retracted between cuts (Z/W program coordinate)
- d: (Optional) The direction (or rotation) of the engraving, in degrees (rotated about the point specified as the lower left corner, default is zero)

Example:

```
G141 "Happy Birthday" X1.1 Y0.5 Z-0.2 F10 P2 R0.1 D45
```

The text will be engraved in the font selected by the most recent G140 command. If no G140 command was specified, an error will result.

Techno supports engraving with up to 2 spindles simultaneously. The supported axes for the two spindles (for engraving purposes) are Z and W. For multi-spindle engraving, the following rules apply:

- Either Z or W can be specified, or both
- Both depths, if specified, must be the same. In other words, z and w must be equal.
- The plunge feedrate applies to both axes, if both Z and W are specified.
- As long as the characters in text and text2 are the same, engraving will occur simultaneously. Once the first difference is encountered, each spindle will engrave the remaining characters separately (see example below).
- If only one spindle (Z or W) is specified in the G141 command and two strings are specified, an error will result.
- If both spindles are specified and only one string is specified, the string will be engraved simultaneously by both spindles.

Example:

```
G141 "Serial 12356" "Serial 12456" Z-0.2 W-0.2 F10 P2 R0.1
```

In this case, “Serial 12” will be engraved simultaneously, “356” will be engraved by the Z spindle, and “456” will be engraved by the W spindle.

G150 Increment Counter

G150 increments the count of the specified counter. The syntax is:

```
G150 Cc [Ii]
```

where the parameters are

c: The counter to increment (1–20).

i: (Optional) The increment amount (default is 1)

Example:

G150 C2 *Increments counter #2 by 1*

G150 C2 I3 *Increments counter #2 by 3*

See “Counter Settings” in the Initial Setup Section for more information.

G151 Set Counter

G151 sets the specified counter to the supplied value. The syntax is:

G151 Cc Vv

where the parameters are

c: The counter to increment (1–20)

v: The new value for the counter

Example:

G151 C2 V0 *Sets counter #2 to zero*

See “Counter Settings” in the Initial Setup Section for more information.

G160 Increment Tool Use Counter

G160 increments the tool usage counter for the current tool. The syntax is:

G160 [Ii]

where the parameters are

i: (Optional) The increment amount (default is 1)

Example:

M06 T3 *Change to tool #3*

.

.

.

G160 *Increments the tool usage count for tool #3 by 1*

G160 I4 *Increments the tool usage count for tool #3 by 4*

G161 Set Tool Use Counter

G161 sets the tool usage count for the specified tool to the supplied value. The syntax is:

G161 Tt Vv

where the parameters are

t: The tool number

v: The new value for the tool usage counter

Example:

G161 T2 V0

Sets the tool usage count for tool #2 to 0

G170 Find Fiducials

The G170 command is part of Techno's Fiducial Correction System. If a G-Code file has a fiducial file associated with it, the G170 command will attempt to locate the fiducials defined. This command has no parameters.

When Techno encounters G170 it does the following for each fiducial:

1. Moves rapidly to its defined location
2. Captures an image from the camera and calculates the offset from the center of the ROI to the fiducial

Techno then uses the offset(s) to compensate the toolpath as follows:

- First fiducial: Adjust for linear error
- First and second fiducial: Adjust for rotational error in the XY plane

If, at some point in the process, Techno does not detect a fiducial, it stops and displays an error message to the user. Issuing a G170 command in a file with no associated fiducials has no effect.

For more information see "Fiducial Box" in the Main Screen Features section.

G179 Cancel Fiducial Correction

The G179 command cancels any fiducial correction created by the G170 command. This command has no parameters.

G180, G181 Safe Envelope Commands

Use of the Safe Envelope can be controlled from G-code by using the G180 and G181 commands. G180 turns off safe envelope restrictions, and G181 turns safe envelope restrictions on. These commands allow the G-Code to temporarily deactivate safe envelope restrictions when necessary.

Neither command has parameters. When Techno encounters G180 in the G-code, safe envelope restrictions are ignored and normal machine envelope restrictions apply. When Techno encounters G181, the motion of the machine is bounded by the the safe envelope (defined on the Advanced Definition panel of the Configuration dialog box.)

For more information, see “Coordinates Menu” in the Main Screen Features section. Note that the menu commands take priority over G180 and G181.

G190 Set PWM/PFM Output

G190 sets the current PWM/PFM port to the supplied value. The syntax is:

G190 Dd Qq

where the parameters are

d: Duty cycle (%)

q: Frequency (hz)

Example:

G190 D50 Q1000

Sets current PWM/PFM port to 1000hz at 50% duty cycle

G210 Select Variable Output Port

G210 sets the current variable output port. The syntax is:

G210 Pp

where the parameter is

p: Port number (1 or 2)

The system applies subsequent S commands to the selected port.

Example:

G210 P1

Selects variable output port 1

S1000

Sets port 1 to value corresponding to 1000

G210 P2

Selects variable output port 2

S500

Sets port 2 to value corresponding to 500

See “Variable Output Settings” in the Initial Setup section, and “S Set Variable Output Port” below for more details.

M00 Program Pause

The M00 command pauses processing of the G-Code program. The syntax is:

```
M00 Pp "message"
```

where the parameters are

p: (Optional) Display message to operator (0 = no message, 1 = display message)

message:

(Optional) Specific message to the operator (replaces standard system message).

If you don't include the P parameter, Techno uses the Message on M00 Program Pause checkbox, on the G-Code panel of the Configuration dialog box, to determine whether or not to display a message. The P parameter overrides the configuration setting.

Example:

```
M00 "Place a new blank in the fixture"
```

M01 Optional Program Pause

The M01 command is identical to the M00 command, except that it may be deactivated when desired. To control activation of the M01 command, use the Execute M01 checkbox on the G-Code panel of the Configuration dialog box. When the checkbox is unchecked, Techno ignores the command.

M30, M30.1 End of Program

The M30 command ends processing of the G-Code program and automatically resets the program to the first executable line. The M30.1 command is similar to M30 except it also restarts the G-Code program. This allows you to run a program over and over indefinitely.

M98, M99, M02 Subroutine Commands

Subroutines allow you to eliminate repetitive programming. Techno supports the use of subroutines with the M98, M99, and M02 (or M30) commands. Use of these commands is best explained through a simple example. The following G-code program uses one subroutine called "mysub":

Example:

G01 X1 Y1 F10	<i>First line of main program</i>
M98 Pmysub	<i>Jump to subroutine "mysub"</i>
G01 X0 Y0	<i>Continued execution after "mysub" ends</i>
M02	<i>End of main program</i>
Omysub	<i>First line of the subroutine called "mysub"</i>

```
G01 X2 Y2           Continued execution within the subroutine
M99                End of subroutine "mysub"
```

In the main program, the M98 command causes program execution to jump to the first line of the subroutine named "mysub". Notice that the letter "P" must immediately precede the name of the subroutine with no spaces.

The subroutine definition begins with the letter "O" followed immediately by the subroutine name with no spaces. The subroutine must end with the M99 command as shown. M99 causes program execution to jump back to the main program, continuing with the line immediately following the M98 line (G01 X0 Y0 above).

When subroutines are used, the main program must end with M02 or M30, the "End of Program" commands. Each subroutine must have a unique name, consisting of any number of alpha-numeric characters. You can "nest" subroutines as much as you like, meaning one subroutine may call another subroutine, which in turn may call another subroutine, and so on.

To call the same subroutine multiple times, include the L parameter as shown below.

Example:

```
M98 Pmysub L10      Execute subroutine "mysub" 10 times
```

This feature provides a simple means to repetitively loop through any block of G-Code. Just place the G-Code in a subroutine, then call the subroutine with the L parameter set to the number of loops required.

It's sometimes convenient to place subroutines in files separate from the main G-Code file. One use of this feature is to create a library of reusable subroutines that you can apply to various projects and jobs. You'll call the subroutines using the same M98 syntax used for subroutines listed in the main program file.

Techno recognizes two types of subroutine files: 'single subroutine' files and 'included' files.

Single Subroutine Files

You may put a single subroutine in a file and name the file as follows:

```
O[subroutine name].[extension]
```

The extension must match the main program file's extension, and the file must be located in the same folder as the main program file. Since the file solely consists of the subroutine, you do not need to include the subroutine definition line (starting with 'O') or the M99 line marking the end of the subroutine.

Using the example above, the subroutine 'mysub' could be stored in a separate file as follows:

File Name: Omysub.fgc

File Contents:

```
G01 X2 Y2
```

Included Files

You may put any number of subroutines in an included file. Each included file must be listed at the top of the main G-Code file using the following syntax:

```
#[file path and name]
```

Once a file has been included in the main G-Code file, Techno treats the included file's subroutines the same as if they were listed at the bottom of the main G-Code file. If you leave out the path, Techno will look for the file in the same folder as the main G-Code file.

Example:

Main G-Code File (Engraves "A B")

```
#C:\GCODE\ALPHABET.FGC
G00 X0 Y0
M98 PmakeA
G00 X1 Y0
M98 PmakeB
M02
```

ALPHABET.FGC File

```
OmakeA
G91
[Incremental G-Code to cut the letter "A"]
G90
M99

OmakeB
G91
[Incremental G-Code to cut the letter "B"]
G90
M99
```

M100, M101 Wait for Input Line

The M100 and M101 commands wait for an input line to reach a desired state as follows:

M100 Wait for normal state

M101 Wait for tripped state

The Wiring setting on the Input Lines panel of the Configuration dialog box affects the behavior of these commands as follows:

Wiring Setting	M100 Waits For Input Line to be...	M101 Waits For Input Line to be...
Normally Closed	Closed	Open
Normally Open	Open	Closed

The command syntax is as follows:

M100 Ix “[optional custom message to operator]”

M101 Ix “[optional custom message to operator]”

where x is the number of the input line to monitor.

Examples:

M100 I2

System waits for input line 2 to reach the normal state, displays standard message if timeout is reached

M101 I7 “Low air pressure”

System waits for input line 7 to become tripped, displays customized message if timeout is reached

You must define the specified input line as “Control” in the Switch Function pull-down menu, on the Input Lines panel of the Configuration dialog box (see “Input Line Settings” in the Initial Setup section for more details).

Techno uses debounce for these commands. For a switch to successfully reach a desired state, it must hold that state continuously for the debounce duration. M100 and M101 use the Debounce when Idle time on the Input Lines panel of the Configuration dialog box.

The M100 and M101 commands time out once the maximum allowable time has passed and the input line has not reached the desired state. When the commands time out, Techno halts processing of the G-Code program and displays a message to the operator. You can set the M100/M101 Timeout duration on the G-Code panel of the Configuration dialog box.

A typical use for these commands is detecting completed motion of an auxiliary device, such as an indexer or air cylinder.

M03, M05, M07, M08, M09, M50, M51, MXX Auxiliary Device Control

Using the M-Code Definitions panel of the Configuration dialog box you can define up to 50 M-Codes to turn on or off various devices via the output lines. You can also define M-Codes to digitally control external devices (using a group of output lines as digital input into the control lines of the device). See “M-Code Definitions” in the Initial Setup section for details on how to set up the M codes.

When Techno processes an M-Code that sets output lines, if the current output line state already reflects the M-Code’s intended action, Techno skips the M-Code, including any associated delay. For example, if the spindle is on, and Techno processes an M-Code to turn on the spindle, the delay associated with turning on the spindle (e.g. 2 seconds) is skipped.

Typical M codes include:

M03	Spindle On
M05	Spindle Off
M07	Mist Coolant On
M08	Flood Coolant On
M09	Coolant Off
M50	Plasma On
M51	Plasma Off

F, G93, G94 Feedrate Commands

The F command specifies the feedrate. The feedrate is modal, which means it stays in effect until another F command occurs.

For all linear moves, specify the feedrate in inches/minute for English units and millimeters/minute for Metric units.

Example:

```
G01 X4.0 Y3.0 Z1.0 F7.0           Moves the tool to program coordinates X=4.0,  
                                  Y=3.0, Z=1.0 at a feedrate of 7.0 in/min
```

Please note the following cases:

- For any move that includes a rotary axis, specify the feedrate either as a linear feedrate (length/minute) or rotary feedrate (degrees/minute), depending on your setting for ‘F Command Interpretation for Rotary Moves’ on the Rotary Axes panel of the Configuration dialog box. For more information see the detailed explanation of rotary feedrates under “G01 Linear Interpolated Feedrate Move” above.
- Techno supports two feedrate modes: standard feedrate (G94) and inverse time feedrate (G93). The explanation above is for G94 mode, which is the predominant mode in CNC machining and the default in Techno. In G93

mode, Techno interprets all feedrate (F) commands as inverse time feedrates. The definition of the F command is

$$F = 1 / \text{Time to complete the move in minutes}$$

When using inverse time feedrates, there must be an F command at the end of every line containing a G01, G02 or G03 command. G93 is used primarily for simultaneous 4 and 5 axis machining.

Example:

G93

G01 X1 Y1 Z-1 A90 F10

Moves the tool to program coordinates X=1, Y=1, Z=-1, A=90 in 6 seconds (0.1 minutes)

S Set Variable Output Port

The S command sets an analog, PWM or PFM output port signal to correspond to the supplied value. Common uses include setting the speed for a spindle or the power for a laser.

The S command affects the current variable output port (1 or 2). The current port is selected by:

- Configuration settings on the Variable Output panel of the Configuration dialog box.
- The G210 command (see “G210 Select Variable Output Port” above).

Example:

S1000

On a mill or lathe, sets the spindle speed to 1000 rpm

See “Variable Output Settings” in the Initial Setup section for more details, including how Techno converts the desired S value to an output signal.

Program Comments

You can add comments to your program by enclosing them in parentheses. Techno ignores anything enclosed in parentheses as shown below.

Example:

```
(Move to beginning of the next feature)
G00 X1.0 Y3.0 (Ready to move Z axis down)
G00 Z-1.5
(Begin next feature)
G01 Z-1.6 F8
G01 X3.0 Y7.5
```

Optional Line

The optional line command is a forward slash (/) at the start of a G-Code line. Techno only executes the line if the Execute Optional G-Code Lines checkbox is checked, on the G-Code panel of the Configuration dialog box. If the checkbox is unchecked, Techno ignores the line.

Example:

```
G00 X2 Y4
/G00 Z1 This line is not always executed
G00 X3 Y5
```

BEGINGRID, ENDGRID

You can generate a grid of parts using G-Code for a single part and the BEGINGRID and ENDGRID commands.

All G-Code that appears between the BEGINGRID and ENDGRID commands will be repeated as appropriate to produce a grid of parts. The syntax for the BEGINGRID command is:

```
BEGINGRID Cols Rows X Y IncrX IncrY ByRow ZigZag
```

where the parameters are

Cols: The number of columns in the grid

Rows: The number of rows in the grid

X: The X program coordinate of the lower left corner of the first part

Y: The Y program coordinate of the lower left corner of the first part

IncrX: The grid spacing in the X direction for each column of the grid

IncrY: The grid spacing in the Y direction for each row in the grid

ByRow: Indicates whether the grid will be formed column-by-column (0) or row-by-row (1)

ZigZag: Indicates whether the grid will start each new row/column at the first row/column (0) or zigzag through the grid (1)

Example:

```
BEGINGRID 3 5 1.2 1.4 0.3 0.7 0 1
```

Produces a grid that is 3 columns by 5 rows, the first element in the grid is at program coordinate X=1.2, Y=1.4, columns in the grid are offset 0.3 and rows are offset 0.7, the grid is formed column-by-column and in a zigzag pattern

The grid must end with an ENDGRID command. Then ENDGRID command takes no parameters.

Example:

```
G00 X0 Y0 Z0.1
```

```
F10
```

```
BEGINGRID 3 4 1 1.5 2.0 2.5 0 1
```

```
G00 X0.1 Y0.1
```

Move to the start of the square

```
G01 Z-0.1
```

Plunge the tool

```
G01 X0.9 Y0.1
```

Cut the square

```
G01 X0.9 Y0.9
```

```
G01 X0.1 Y0.9
```

```
G01 X0.1 Y0.1
```

```
G01 Z0.1
```

Raise the tool

```
ENDGRID
```

```
G00 X0 Y0 Z0.1
```

This is a 3 x 4 grid of squares. The lower left of the first square in the grid is at (1.1, 1.6). This is because the starting position of the square (0.1, 0.1) is added to the starting position of the grid (1.0, 1.5). The next square in the grid will start at (1.1, 4.1) because the grid is being created column by column. Techno builds each column completely before moving onto the next column. Once all four rows are built in the first column, with squares starting at (1.1, 1.6), (1.1, 4.1), (1.1, 6.6), and (1.1, 9.1), the next square will start at (3.1, 9.1) on the 4th row of the 2nd column. This is because the grid is being built in a zigzag pattern. If it were not in a zigzag pattern, the next column would have been started at (3.1, 1.6), which is the 1st row of the 2nd column.

Advanced Programming Reference

Techno provides a number of advanced programming features, allowing you to create more flexible and powerful G-Code programs.

Values

In every G-Code example up to this point, all values were *literal* values. A literal value is one that's typed into the G-Code program and cannot change. Examples of literal values are:

```
1
2.33
"Please close the door"
```

An example of these values used in G-Code is:

```
T1
G00 X2.33
M00 "Please close the door"
```

Variables

In addition to literal values, the Techno G-Code interpreter understands variables.

A variable is a placeholder for a value. There are two categories of variables in Techno: System, Program and User. System variables are defined and set by the Techno software itself. These variables represent status values in the software, and therefore allow you to access information about the current state of the program. Most system variables cannot be modified in a G-Code program. However, all system variables can be read in a G-Code program. The system variables currently supported are:

- `#CurrTool` – This variable represents the number of the tool currently loaded. Its value is 0 if no tool is loaded. This value can be modified in a G-Code program. It is typically modified in a tool change macro.
- `#NextTool` – This variable represents the number of the tool that will be loaded on the next tool change command (M06). Its value is 0 if no tool has been specified with the T command. This value can be modified in a G-Code program.
- `#CurrFeedrate` – This variable represents the current feedrate. This value can be modified in a G-Code program.
- `#TlChgFeedrate` – This variable represents the feedrate at which the controlled rate portions of an automatic tool change will occur.
- `#TlChgCheckPower` – This variable represents the state of the Monitor Tool Chuck Power checkbox on the Tool Change panel of the Configuration dialog box (checked = true).

- #TlChgPowerInputLine – This variable represents the setting of the Input Line text box on the Tool Change panel of the Configuration dialog box.
- #TlChgCheckPowerInputNormal – This variable represents the setting of the Power On State pull-down menu on the Tool Change panel of the Configuration dialog box (Normal = true).
- #Tool – This is an array variable that is used to access properties of any configured tool.
- #RefPoint – This is an array variable that is used to access properties of any defined reference point.
- #FixOff – This is an array variable that is used to access properties of any defined fixture offset.
- #TlRackPos – This is an array variable that is used to access the properties of any tool rack position.
- #Machine – This variable represents the current position of the machine tool in machine coordinates.
- #Program – This variable represents the current position of the machine tool in program coordinates.
- #ProgZeroOff – This variable represents the offset from the machine coordinate system to the program coordinate system.
- #ToolOff – This variable represents the tool offset currently in effect.
- #Input – This variable represents the current state of the input lines.
- #Output – This variable represents the current state of the output lines.
- #LastFeedholdPos – This variable represents the location where feed hold most recently occurred.

#Tool, #RefPoint, #FixOff, #TlRackPos, #Input and #Output are array variables. Array variables are discussed later in this section.

#Machine, #Program, #ProgZeroOff, #ToolOff, #Input, #Output and #LastFeedholdPos are runtime variables. Runtime variables are discussed later in this section.

In addition to the above variables, there are several general-purpose system variables available that do not represent any system status values. These variables have been provided for use in macros. You are free to set and read these variables as desired.

- #SysInteger1 - #SysInteger8 – Eight system-declared INTEGER variables.
- #SysReal1 - #SysReal8 – Eight system-declared REAL variables.

- #SysBoolean1 - #SysBoolean8 – Eight system-declared BOOLEAN variables.
- #SysString1 - #SysString8 – Eight system-declared STRING variables.

Program variables are defined and maintained by the G-Code programmer. In order to use a Program variable, you must first declare it with a simple statement that includes the type of the variable and its name, as in this example:

```
GLOBAL INTEGER #Count
```

In the above statement, the word “GLOBAL” indicates the scope of the variable (more on that later), “INTEGER” is the type, and “#Count” is the variable name. The types supported by Techno are:

- INTEGER – These are whole numbers (i.e. numbers without decimal points). They can be positive or negative (or zero).
- REAL – These are rational (or fractional) numbers (i.e. numbers with decimal points). They can be positive or negative (or zero).
- BOOLEAN – Variables of this type can have one of two values: TRUE or FALSE.
- STRING – Variables of this type represent text.

Example declarations:

```
GLOBAL INTEGER #Index
```

```
GLOBAL REAL #X
```

```
GLOBAL BOOLEAN #IsDone
```

```
GLOBAL STRING #Message
```

As with all other words in a Techno G-Code program, the case of the letters does not matter. In other words, these two declarations are the same:

```
GLOBAL REAL #X
```

```
Global REAL #x
```

If these two lines appeared in the same G-Code file, an error would be reported because the same variable is being declared twice.

Variable names can be any sequence of letters or numbers or the underscore character (_), but must always start with a number sign (#). The following are all legal variable names:

```
#X
```

```
#Index
```

```
#3rdAxis
```

```
#Is_Done
```

The word “GLOBAL” in a variable declaration indicates the scope, or context, of the variable. A variable with GLOBAL scope can have its value changed or

retrieved anywhere in the G-Code program, even in a file that is included into the program. GLOBAL variables must be declared before any line of G-Code other than lines that include files using #.

If the word GLOBAL is omitted from the variable declaration, that variable is considered to have local scope. This means that the variable's value can only be changed or retrieved in the same subroutine in which it was declared.

Currently, Techno only supports GLOBAL variables, so the word "GLOBAL" is optional. Future releases of Techno will support local variables. In these releases, the word GLOBAL will be required to indicate variables with global scope.

One or more variables can be declared on the same line. For example,

```
REAL #X #Y #Z
```

declares three variables, #X, #Y, and #Z, which are all of type REAL. When declaring more than one variable on the same line, the variable names must be separated by spaces.

The value of a variable can be set by using the assignment operator. The assignment operator is an equals sign (=). Variables can be assigned literal values, as shown below:

```
INTEGER #Count  
#Count = 0
```

The above code declares an integer variable called #Count, and then assigns #Count the literal value of 0. The process is similar for all variable types.

```
REAL #X #Y  
BOOLEAN #IsDone  
STRING #Message  
#X = 1.0  
#Y = 2.1  
#IsDone = FALSE  
#Message = "Please make sure the enclosure is closed."
```

A variable can be used in place of a literal value in a G-Code file. For instance,

```
REAL #X #Y  
#X = 1.0  
#Y = 2.1  
G00 X#X Y#Y
```

will cause a rapid move to X = 1.0, Y = 2.1. Here are more examples of variable use:

```
INTEGER #ToolNumber  
STRING #Message
```

```
#ToolNumber = 1
M06 T#ToolNumber
#Message = "Please make sure the enclosure is closed."
M00 #Message
```

Some variables types will be automatically converted to the correct type as necessary. For instance, motion commands take REAL type parameters, but INTEGER type parameters will be automatically converted. Example:

```
INTEGER #X #Y
#X = 1
#Y = 2
G00 X#X Y#Y
```

In this case, #X and #Y will be converted automatically to REAL values. Similarly, REAL values will be automatically converted to INTEGER values. In this case, the fractional part of the number is lost.

```
REAL #ToolNumber
#ToolNumber = 2.7
M06 T#ToolNumber
```

The above code will perform a tool change to tool number 2. It is important to note that when a REAL value is converted to an INTEGER value it is ***not*** rounded. In other words, 2.001 converted to an integer is 2, and 2.999 converted to an integer is 2.

Any type of a variable can be converted to a STRING. This means that the following lines of code are all legal:

```
STRING #Message
REAL #X
INTEGER #Count
BOOLEAN #IsDone

#Message = "Test message"
#X = 1.0
#Count = 0
#IsDone = FALSE

M00 #Message
M00 #X
M00 #Count
M00 #IsDone
```

Variables can be assigned to each other as well.

```

REAL #X1 #X2
INTEGER #ToolNumber

#X1 = 3.14
#X2 = #X1
#ToolNumber = #CurrTool

```

Value of #X1 becomes 3.14

Value of #X2 becomes 3.14

Value of #ToolNumber becomes the currently loaded tool (note that #CurrTool is a system variable)

Some system variables represent objects, rather than simple types. The system variables in this category are #Machine, #Program, #ProgZeroOff, and #ToolOff. Here is a description of the properties of each:

- Machine
 - X – the current position on the X axis in machine coordinates
 - Y – the current position on the Y axis in machine coordinates
 - Z – the current position on the Z axis in machine coordinates
 - A (W) – the current position on the A (or W) axis in machine coordinates
- Program
 - X – the current position on the X axis in program coordinates
 - Y – the current position on the Y axis in program coordinates
 - Z – the current position on the Z axis in program coordinates
 - A (W) – the current position on the A (or W) axis in program coordinates
- Program Zero Offset
 - X – the current offset from machine to program coordinates on the X axis
 - Y – the current offset from machine to program coordinates on the Y axis
 - Z – the current offset from machine to program coordinates on the Z axis
 - A (W) – the current offset from machine to program coordinates on the A (or W) axis
- Tool Offset
 - X – the currently applied tool offset on the X axis
 - Y – the currently applied tool offset on the Y axis
 - Z – the currently applied tool offset on the Z axis

Since these variables represent objects rather than simple types, there is a special way to use them. The values represented by these variables can be accessed by

specifying the desired property using a period (dot) followed by the property name. For instance,

```
#Machine.X
```

is how you would access the current position on the X axis in machine coordinates.

Array variables are groups of variable values that are indexed. They can be accessed by supplying the desired index value (position in the array). An array variable is declared in much the same way as any other variable, except that the size of the group, also known as the dimension of the array, must be specified.

Example:

```
REAL #XPositions{5}
```

The above code declares a REAL array variable called #XPosition that contains 5 values. Arrays can be created for any type (INTEGER, REAL, BOOLEAN, or STRING). The name of the array variable must be followed by a pair of braces, {}, that surround a number. The lowest number allowed for an array dimension is 1. An array of 1 is exactly the same as a non-array variable. The highest number allowed for an array dimension is 10,000. In other words,

```
INTEGER #Val{0}
```

```
BOOLEAN #MyArray{10001}
```

will both result in errors when the program is loaded.

An array value is accessed in the same manner as any other variable, except that the index, must be specified. For example,

```
REAL #XPositions{3} #YPositions{3}
```

```
#XPositions{1} = 0.0
```

```
#XPositions{2} = 1.0
```

```
#XPositions{3} = 2.0
```

```
#YPositions{1} = 4.0
```

```
#YPositions{2} = 3.0
```

```
#YPositions{3} = 2.0
```

The lowest index number is 1, and the highest index number is the dimension of the array variable. In other words, if an array variable is declared with a dimension of 3, the highest index number is 3.

The index itself can be a literal value or a variable value.

```
REAL #XPositions{3} #YPositions{3}
```

```
#XPositions{#CurrTool} = 0.0
```

```
#YPositions{#CurrTool} = 2.0
```

In the above example, if no tool is loaded (i.e. #CurrTool = 0), an error will result because the lowest index number is 1. Similarly, if #CurrTool = 4 or higher, an error will result because both array variables were declared with a dimension of 3.

The system variables #Tool, #RefPoint, #FixOff, and #TIRackPos are array variables. However, they have special meaning because their types are not any of the built-in types (INTEGER, REAL, BOOLEAN, or STRING). Instead, these variables refer to objects (defined in the Configuration dialog box) that in turn have properties. Specifically, #Tool refers to tools, #RefPoint refers to reference points, #FixOff refers to fixture offsets, and #TIRackPos refers to tool rack positions. Here are the properties of each object:

- Tool
 - X – the X offset of the tool
 - Y – the Y offset of the tool
 - Z – the Z offset of the tool
 - DIAM – the diameter of the tool
 - DESC – the description of the tool
- Reference Point
 - X – the X coordinate of the reference point
 - Y – the Y coordinate of the reference point
 - Z – the Z coordinate of the reference point
 - A(W) – the A (or W) coordinate of the reference point
 - DESC – the description of the reference point
- Fixture Offset
 - X – the X coordinate of the fixture offset
 - Y – the Y coordinate of the fixture offset
 - Z – the Z coordinate of the fixture offset
 - A(W) – the A (or W) coordinate of the fixture offset
 - DESC – the description of the fixture offset

- Tool Rack
 - X – the X coordinate of the tool rack position
 - Y – the Y coordinate of the tool rack position
 - Z – the Z coordinate of the tool rack position
 - RAISED – the position to which the tool is to be raised
 - NEARGRIP – the position to which the machine tool will move rapidly prior to moving into position to grip the tool
 - NEARRELEASE – the position to which the machine tool will move rapidly prior to moving into position to release the tool
 - GRIP – the position at which the machine tool will grip the tool in the chuck
 - RELEASE – the position at which the machine tool will release the tool from the chuck

Because #Tool, #RefPoint, #FixOff, and #TIRackPos refer to objects, they cannot be accessed directly. However, each individual property of these objects can be retrieved. This is done by adding a period (dot) after the variable name, and then the name of the property. For example,

```
G00 X#RefPoint{1}.X
```

This will cause a rapid move on the X axis to the X coordinate of reference point #1. Similarly,

```
M00 #Tool{1}.DESC
```

will display a message containing the description of tool number 1.

Since #CurrTool and #NextTool are INTEGERS, you could display the description of the currently loaded tool by typing:

```
M00 #Tool{#CurrTool}.DESC
```

User variables are defined on the User Variables panel of the Configuration dialog box. Other than that, Techno treats these variables the same as Program variables. For more information see “User Variable Settings” in the Initial Setup section.

Runtime Variables

In the previous discussion of variables, the value of each variable is known when the program is loaded. However, sometimes it is desirable to know the current state of the system while a program is running. For this reason, Techno CNC has the ability to examine the current values of certain system variables. This gives the user the ability to have the system execute different parts of a G-code file based on the current value of a desired variable. This feature is called runtime branching, but the capabilities actually extend further than just branching.

The following system variables represent values of the current state of the Techno CNC system:

Variable Name	Type	Description
#Input	Boolean Array	Value of the input lines
#Output	Boolean Array	Value of the output lines
#Machine	Position	Current position of the machine in machine coordinates
#Program	Position	Current position of the machine in program coordinates
#LastFeedholdPos	Position	Location where feed hold most recently occurred
#ProgZeroOff	Offset	Current offset from machine coordinates to program coordinates
#ToolOff	Offset	Current tool offset

The examples below illustrate possible uses of these variables.

Example #1 – Branching based on the state of an input line (runtime branching)

If you wish to execute different G-code depending on the state of the first input line, you could do the following:

```
IF #Input{1} THEN
    (Things to do if input line 1 is tripped)
ELSE
    (Things to do if input line 1 is not tripped)
ENDIF
```

Example #2 – Measuring

If you wish to check the distance between two points, you could do the following:

```
GLOBAL REAL #MyVarBegin
GLOBAL REAL #MyVarEnd
GLOBAL REAL #Diff

(Some G-code)
#MyVarBegin = #Machine.X (Current X location, this is our
                          first point)

(Some more G-code)
#MyVarEnd = #Machine.X (Current X location. This is our
                       second point)

#Diff = #MyVarEnd - #MyVarBegin
IF #Diff > 3.0 THEN
```

```
M00 "The spacing is to too large!"  
ENDIF
```

Since runtime variables require information that is only known while the G-Code is actually running, two Techno features behave differently when the G-Code contains runtime variables.

- Toolpath Preview – Techno previews the G-Code file only up to the first use of a runtime variable.
- Running a G-Code file when not connected to the Signal Generator – When Techno encounters a G-Code line that requires the value of #Input or #Output, it displays a dialog that asks you to supply the required value. This allows you to simulate the G-Code file under various runtime conditions.

Runtime variables may also be used in the Tool Length Formula (on the Tool Length Sensing panel of the Configuration dialog box).

Operators

In addition to declaring variables, setting their values, and using their values, Techno supports the use of several operators to make G-Code programs more flexible and powerful. There are mathematical, grouping, string, and comparison operators.

Mathematical

The mathematical operators work for INTEGER and REAL values. They are:

- + adds two values
- subtracts two values
- * multiplies two values
- / divides two values

Examples:

```
INTEGER #Count #Index  
REAL #X #Y #Z  
  
#Count = 0  
#Index = #Count + 1  
#Count = #Count - 2  
  
#X = 1.0  
#Y = #X * 2.0  
#Z = #Y / #X
```

More than one mathematical operator can be used on one line. For example,

```
INTEGER #Val
#Val = 1 + 2 - 3
```

When using mathematical operators, the arithmetic precedence applies. This means that multiplication and division operators are interpreted first, then addition and subtraction. If more than one multiplication or division operator is on the same line, the operators are processed from left to right. The same is true for addition and subtraction operators. Here are some examples:

```
INTEGER #Val
#Val = 2 + 2 - 3           #Val = 1; 2 + 2 is evaluated first, then 3 is
                           subtracted from the result
#Val = 4 / 2 * 3          #Val = 6; 4 / 2 first, then result multiplied by 3
#Val = 1 + 2 * 3          #Val = 7; 2 * 3 first, then added to 1
#Val = 1 - 2 * 3 + 4      #Val = -1; 2 * 3 first, then subtracted from 1,
                           then 4 added to result
#Val = 4 / 2 - 3 * 1      #Val = -1; 4 / 2 first, then 3 * 1, then second
                           result subtracted from first
```

Grouping

If a different order of evaluation is desired, Techno recognizes grouping operators. These are like parentheses in mathematics, except brackets [] are used. Brackets are used because in G-Code the left parenthesis starts a comment. Any expression inside a pair of brackets is evaluated before operators outside the brackets. For example,

```
INTEGER #Val
#Val = 1 + 2 * 3          #Val = 7 as explained above
#Val = [1 + 2] * 3        #Val = 9; 1 + 2 first, then result multiplied by 3
```

Pairs of brackets can be nested to clarify complicated expressions:

```
INTEGER #Val
#Val = [[[1 + 2] * 3 - [4 + 5]] / 6 + 7] * 2
#Val = 14
first 1 + 2 = 3
then 3 is multiplied by 3 = 9
then 4 + 5 = 9
then 9 is subtracted from 9 = 0
then 0 is divided by 6 = 0
then 7 is added to 0 = 7
then 7 is multiplied by 2 = 14
```

String

The only string operator is the concatenation operator.

& combines two strings to form a single string

Example:

```
STRING #Message1 #Message2
#Message1 = "Please change tool"
#Message2 = " and close enclosure"
M00 #Message1 & #Message2
```

The above code will cause the following message to appear:

Please change tool and close enclosure

Since any other type of value can be converted to a STRING, numbers can be displayed very easily. For example,

```
M00 "The currently loaded tool is " & #CurrTool
```

Here, the system variable #CurrTool is automatically converted to a STRING. If #CurrTool is 1, the message is

The currently loaded tool is 1

Note: Do not surround the variable name in quotes, otherwise the variable name itself will be displayed. Surrounding the name in quotes will turn it into a literal value. For example,

```
M00 "The currently loaded tool is " & "#CurrTool"
```

will cause the following message to be displayed:

The currently loaded tool is #CurrTool

Several concatenation operators can be used on the same line. This allows complicated text messages to be built. Example:

```
M00 "Tool #" & #CurrTool &
    " will be replaced by tool #" & #NextTool
```

If the current tool is 1 and the next tool is 2, the message displayed will be:

Tool #1 will be replaced by tool #2

Mathematical operators can appear on the same line as the concatenation operator. If this is the case, all mathematical operations are evaluated first. The results of the mathematical operations are converted to strings and the concatenation takes place. Example:

```
M00 "The result " & 1 + 2 &
    " was obtained by adding 2 to 1"
```

The expression `1 + 2` is evaluated first, then the result is converted to a `STRING` and concatenated with the string literals. The message that is displayed is:

The result 3 was obtained by adding 2 to 1

Comparison

The comparison operators work for any type of value, but both values must be the same type (or must be convertible to the same type). The comparison operators are:

- = TRUE if both values are equal, FALSE if not
- != TRUE if the values are different, FALSE if they are the same
- > TRUE if the value to the left of the operator is greater than the value to the right, FALSE if not
- >= TRUE if the value to the left of the operator is greater than or equal to the value to the right, FALSE if not
- < TRUE if the value to the left of the operator is less than the value to the right, FALSE if not
- <= TRUE if the value to the left of the operator is less than or equal to the value to the right, FALSE if not

Examples of comparison operator use appear in the following sections.

Boolean

Techno also supports Boolean logic operators. These operators can be used wherever a Boolean value is expected.

The AND operator performs a Boolean AND operation (ie. if both operands are TRUE, the expression is TRUE, otherwise it is FALSE).

Syntax:

```
BOOLEAN operand1 AND operand2
```

Example:

```
BOOLEAN #Test1
BOOLEAN #Test2
BOOLEAN #Result
#Test1 = TRUE
#Test2 = FALSE
#Result = #Test1 AND #Test2    #Result is FALSE
IF #Test1 AND #Test2 THEN      The 'IF' condition is FALSE
...
```

The OR operator performs a Boolean OR operation (ie. if either operand is TRUE or both are TRUE, the expression is TRUE, otherwise it is FALSE).

Syntax:

```
BOOLEAN operand1 OR operand2
```

Example:

```
BOOLEAN #Test1
BOOLEAN #Test2
BOOLEAN #Result
#Test1 = TRUE
#Test2 = FALSE
#Result = #Test1 OR #Test2      #Result is TRUE
WHILE #Test1 OR #Test2 DO      The 'WHILE' condition is TRUE
...
```

Flow Of Control

Flow of control commands let you build simple or complex logic into your G-Code programs. Techno provides two types of flow of control: Branches and Loops. Branches make it possible for a G-Code program to execute different sections of the code based on a condition. The syntax for a Branch is:

```
IF condition THEN
...
ENDIF
```

where *condition* is a Boolean value or expression. If *condition* is TRUE, the code between the IF line and the ENDIF line is executed. If *condition* is FALSE, the code is skipped and execution continues with the first line after ENDIF.

At times there is a need to execute some code if a condition is true and execute different code if that condition is FALSE. This can be done by using an ELSE statement:

```
IF condition THEN
...
ELSE
...
ENDIF
```

In this case, the code between the IF and the ELSE is executed if *condition* is TRUE, and the code between ELSE and ENDIF is executed if *condition* is FALSE.

Example:

```
IF #NextTool = 1 THEN
    M00 "Please load tool #1"
ELSE
    M00 "Please load tool #2"
ENDIF
```

In this example, if #NextTool (a system variable) is 1, the user will see a message to load tool number 1; otherwise a message to load tool number 2 will appear. The above example includes an example of the use of the '=' comparison operator as well. Remember that the comparison operator result is BOOLEAN (that is, TRUE or FALSE).

For even greater flexibility, additional conditions can be specified:

```
IF #NextTool = 1 THEN
    M00 "Please load tool #1"
ELSEIF #NextTool = 2 THEN
    M00 "Please load tool #2"
ELSEIF #NextTool = 3 THEN
    M00 "Please load tool #3"
ELSE
    M00 "An invalid tool was selected"
ENDIF
```

The above example lets you display different messages based on the next tool to be loaded. Note that only one ENDIF was necessary. Also, note that THEN is required for each IF or ELSEIF.

IF statements can be nested. This means that inside a block of code between IF and ENDIF, one or more IFs can appear. For example,

```
IF #CurrTool > 0 THEN
    IF #CurrTool = 1 THEN
        M00 "The first tool is loaded"
    ELSEIF #CurrTool = 2 THEN
        M00 "The second tool is loaded"
    ENDIF
ELSE
    IF #CurrTool = 0 THEN
        M00 "No tool is loaded"
    ELSE
        M00 "An invalid tool is loaded"
```



```
ENDIF
ENDIF
```

The lines of code between the first IF and the ELSE will be executed if the currently loaded tool number is greater than zero. The actual message that will appear depends on the number of the tool that is loaded. Notice that each block of code that starts with an IF eventually ends with an ENDIF. These pairings must be present, or an error will result when the G-Code file is loaded.

The other type of flow of control statement is the Loop. The syntax of the Loop is:

```
WHILE condition DO
...
ENDWHILE
```

The lines of code between the WHILE and the ENDWHILE will be executed as long as *condition* is TRUE. When *condition* is FALSE, the loop is exited and execution continues with the line of code following the ENDWHILE. Example,

```
INTEGER #Count
#Count = 0
WHILE #Count < 5 DO
    M00 "Loop count = " & #Count
    #Count = #Count + 1
ENDWHILE
M00 "Not in loop"
```

This code will cause the following messages to appear:

```
Loop count = 0
Loop count = 1
Loop count = 2
Loop count = 3
Loop count = 4
Not in loop
```

If the condition specified in the loop is initially FALSE, the code in the loop will be skipped entirely.

Note: When creating loops, you must be careful to ensure that the loop condition will eventually be FALSE. Otherwise, an infinite loop will result and the program will never load. In the example above, omitting `#Count = #Count + 1` creates an infinite loop.

Loops can appear inside other loops or branches. Branches can appear inside loops as well as other branches. Here's a program that illustrates this as well as most of the other features discussed in this section.

(Use GLOBAL for these variables because their values
(are used in the main routine and in subroutines

```
GLOBAL INTEGER #Row
GLOBAL INTEGER #Col
GLOBAL REAL #XBase #YBase
GLOBAL REAL #RowSpacing #ColSpacing
GLOBAL REAL #XLoc #YLoc
GLOBAL BOOLEAN #MakeMeSmile
GLOBAL INTEGER #Result
```

(Don't need GLOBAL for these variables because they
(are only used in the main routine

```
INTEGER #TotalRows
INTEGER #TotalCols
```

```
#TotalRows = 3
#TotalCols = 4
#XBase = 0.3
#YBase = 0.3
#RowSpacing = 0.8
#ColSpacing = 0.7
#Row=0
#Col=0
```

```
F10
```

```
WHILE #Row < #TotalRows DO
  WHILE #Col < #TotalCols DO
    M98 PUpdateLocation
    #Result = #Col / 2
    IF [#Col - #Result * 2] = 1 THEN
      #MakeMeSmile = TRUE
    ELSE
      #MakeMeSmile = FALSE
    ENDIF
    M98 PSmiley
    #Col=#Col+1
  ENDWHILE
```

```
#Row=#Row+1
#Col=0
ENDWHILE
G00 X0 Y0

OUpdateLocation
#XLoc = #XBase + #Col * #ColSpacing
#YLoc = #YBase + #Row * #RowSpacing
G00 X#XLoc Y#YLoc
M99

OSmiley
G00 Z0
G91
G00 X-0.15 Y0.1
G01 Z-0.5
G02 I0.05
G01 Z0.5
G00 X0.3
G01 Z-0.5
G02 I-0.05
G01 Z0.5
G00 Y-0.2
G01 Z-0.5
IF #MakeMeSmile THEN
    G02 X-0.3 I-0.15 J0.1
ELSE
    G03 X-0.3 I-0.15 J-0.1
ENDIF
G01 Z0.5
G00 X-0.15 Y0.1
G01 Z-0.5
G02 I0.3
G90
G00 Z0
M99
```

File Access

Three file access commands are supported in Techno. These commands allow the G-Code programmer to save data to a file while the program is running.

Creating Files

The CreateFile function creates and opens a new file. If the file already exists, it is overwritten. If the file cannot be created, an error message is displayed. If another file is open, it is closed; only one file can be open at a time.

Syntax:

```
CreateFile[filename]
```

where `filename` is the name of the file you wish to create. The file name can include an absolute path, or a path that's relative to the current G-Code file.

Examples:

```
CreateFile["MyFile.txt"]           Creates MyFile.txt in the same  
                                   folder as the G-Code file  
  
CreateFile["C:\Users\Guest\Documents\AnotherFile.txt"]  
STRING #FileName  
#FileName = "TheFile.abc"  
CreateFile[#FileName]
```

Opening Files

The OpenFile function opens an existing file. If the file does not exist a new one is created. Anything that is saved to the file after it is opened will be saved to the end of the file. If the file cannot be opened, an error message is displayed. If another file is open, it is closed; only one file can be open at a time.

Syntax:

```
OpenFile[filename]
```

where `filename` is the name of the file you wish to open. The file name can include an absolute path, or a path that's relative to the current G-Code file.

Examples:

```
OpenFile["MyFile.txt"]           Opens MyFile.txt located in the  
                                   same folder as the G-Code file  
  
OpenFile["C:\Users\Guest\Documents\AnotherFile.txt"]  
STRING #FileName  
#FileName = "TheFile.abc"  
OpenFile[#FileName]
```

Writing to Files

The WriteLine function writes a line of data to an open file. If a file has not been

opened, an error message is displayed.

Syntax:

```
WriteLine[line]
```

where `line` is the text to write into the file.

Examples:

```
WriteLine["This is a test"]  
STRING #TestLine  
#TestLine = "Another test"  
WriteLine[#TestLine]  
WriteLine["Program X: " & #Program.X]
```

G-Code Data Files

Techno supports G-Code data files. These files can be read while a G-Code program is running to provide data that can be used in the G-Code program. A typical use for G-Code data files is to provide point locations. But due to the flexibility of the design, G-Code data files can be used for any purpose to supply data to a program.

A G-Code data file is an XML file that contains the data format and the data. An XML file is a collection of tags. These tags come in pairs, an opening tag and a closing tag. The tag name can be any string of letters, but cannot contain any spaces. A tag is surrounded by angle brackets (`<tag>`). The opening tag contains only the name inside the brackets (`<tag>`) but the closing tag contains a slash in front of the name (`</tag>`). XML tags must come in pairs (opening and closing). Tags can be nested within each other, but must be closed in the reverse order they were opened. For example:

```
<OuterTag>  
    <InnerTag>  
    </InnerTag>  
</OuterTag>
```

An XML tag defines an element. The element's name is the tag name. So `<tag>` defines an element named "tag". Some elements can have attributes. Attributes are specified inside the angle brackets with the tag, and the value of the attribute must be enclosed in quotes. For instance:

```
<Test Field="XYZ">
```

defines an element named "Test" and sets its "Field" attribute to "XYZ".

The names of elements and the attributes that are valid for each element are defined by the program that reads the XML file. In the case of G-Code data files, that is Techno CNC.

While opening tags must have a closing tag, there is a shorthand way to close a tag. The slash can appear at the end of the tag line (`<tag attribute="X"/>`). Often, the use of this shorthand gives the file a cleaner look. However, it is just as valid to write `<tag attribute="X"></tag>` and the writer of the XML file is free to choose the form that he prefers.

Here is an example of a G-Code data file:

```
<GCodeData>
  <Format>
    <Attributes>
      <Attribute Name="X" Type="REAL"/>
      <Attribute Name="Y" Type="REAL"/>
      <Attribute Name="FLAG" Type="INTEGER"/>
    </Attributes>
  </Format>
  <Data X="1.0" Y="2.0" FLAG="1"/>
  <Data X="2.0" Y="1.0" FLAG="2"/>
  <Data X="0.0" Y="0.0" FLAG="3"/>
</GCodeData>
```

The first tag, `<GCodeData>`, is required. The format and data lines must appear in the file between this tag and its closing `</GCodeData>`.

The next tag, `<Format>`, is required as well. This tag tells Techno how to interpret the data lines in the file.

The `<Attributes>` tag appears between `<Format>` and `</Format>`. This is also required. This is where the fields (or attributes) of each data line are defined.

Each attribute of the data is defined by the `<Attribute>` tag. The data contained in the G-Code data file can have any attribute, as long as it is specified in this section.

There are two attributes of an `<Attribute>`: Name and Type. The Name attribute defines the name of the data attribute. This name can be any sequence of letters, but must not contain any spaces. The Type attribute defines the type of the data attribute. The Type must be one of the following:

- REAL – a floating point number
- INTEGER – a whole number, positive or negative (or zero)
- STRING – text
- BOOLEAN – must be TRUE or FALSE

In the example above, the G-Code data have three attributes: X, which is a real number; Y, which is a real number; and FLAG, which is an integer.

The `<Data>` elements appear after the format section. Any (or all) of the attributes for the `<Data>` may be specified, but it is not an error if an attribute is not specified. However, any attribute that is not specified will have an undefined value.

Any number of <Data> elements can be specified.

Loading Data Files

The G-Code data file is loaded into the running program by calling the LoadData function.

Syntax:

```
LoadData[filename]
```

where filename is the name of the XML file containing the G-Code data.

Examples:

```
LoadData[ "MyData.xml" ]
LoadData[ "C:\Users\Guest\Documents\MoreData.xml" ]
STRING #FileName
#FileName = "StillMoreData.xml"
LoadData[#FileName]
```

If the file name does not include a path, Techno looks for the file in the same folder as the G-Code file.

Multiple data files can be loaded, but the format for the data elements must be the same or errors will occur. The data in each file that is loaded will be appended to the data that have already been loaded.

Using Data Files

The data contained in the G-Code data files are read and stored in an array. This array can be accessed via the predefined variable #DataList. #DataList is an array variable. The order of the data items in #DataList is the same as the order they were specified in the G-Code data file.

The attributes of each item in #DataList are accessed as properties. For example:

```
#DataList{2}.FLAG
```

accesses the FLAG attribute of the second item in the data list.

Another predefined variable, #DataListCount, contains the number of items in #DataList. Using this variable, it's easy to iterate through all items in the data list:

```
INTEGER #Item
#Item = 1
WHILE #Item <= #DataListCount DO
    G01 X#DataList{#Item}.X Y#DataList{#Item}.Y (Move to point)
    #Item = #Item + 1 (Don't forget to increment #Item!)
ENDWHILE
```

Clearing Data

As mentioned above, multiple data files can be loaded, and the data from each file will be appended to the data list. Because of this, a means for clearing the data list has been provided. The ClearData command clears the data list and sets the list count to 0. If ClearData is followed by a LoadData, the data in the file specified by the LoadData function will be the only data in the list.

It is not necessary to clear the data list before loading the first G-Code data file. The data list is automatically cleared each time a new G-Code file is loaded.

Feedrate Override Commands

There are two commands that affect feedrate override.

EnableFO/DisableFO

The EnableFO/DisableFO pair of keywords controls how feedrate override affects the G-Code program. These keywords are provided because there are cases when an override to the programmed feedrate cannot be tolerated. For example, in an automatic tool change macro, feedrate moves are often used. Overriding these feedrates might cause problems, so DisableFO can be used to ensure that the programmed feedrate will be enforced.

Here's an example of EnableFO and DisableFO:

```
G00 X1 Y1
DisableFO                               Make sure the programmed feedrate is not
                                         overridden

G01 Z-0.5 F3
EnableFO                                 Override is okay from this point
...                                     Continue
```

In the above example, the programmer ensures that the tool will plunge at a rate of 3 inches/minute by disabling feedrate override before performing the plunge.

SetFO

The SetFO keyword sets the feedrate override percent to a specified value. A typical use for this command is to reset feedrate override to 100% at certain points in a G-Code file, to override any changes made by the operator.

Example:

F10	
G01 X20 Y0	<i>During this move the operator adjusts feedrate override to 150%, so the machine tool is moving at 15 inches/minute</i>
SetFO[100]	<i>Feedrate override is set to 100%</i>
G01 Y5	<i>The feedrate for this move is 10 inches/minute</i>
...	<i>Continue</i>

Saving and Restoring the System State

The SaveState/RestoreState pair of keywords causes certain state variables to be saved or restored. The specific status items affected are:

- Offset from Machine to Program coordinates (G92/G54/G55/G56/G57/G58/G59/G54.1 /Program Set Button)
- Absolute vs. relative positioning (G90/G91)
- English vs. Metric units (G20/G21 and G70/G71)
- Canned cycle return position (G98/G99)
- Feedrate (F)
- Spindle Speed (S)
- Local coordinate system (G52)
- Arc plane (G17/G18/G19)
- Motion mode (G00/G01/G02/G03/G73/G76/G81/G82/G83/G85)
- Cutter compensation (G41/G42)
- Scaling (G51)
- Feedrate override enable/disable (EnableFO/DisableFO)
- Safe envelope (G180/G181)
- Feedrate mode (G93/G94)

The SaveState/RestoreState keywords are most useful in macros and subroutines where you wish to restore the state of the program before exiting from the macro or subroutine. Here's an example of keyword use:

G91	<i>Relative positioning mode</i>
G20	<i>English units</i>

SaveState	
G90	<i>Absolute positioning mode</i>
SaveState	
G91	<i>Relative positioning mode</i>
RestoreState	<i>This puts the program back into absolute positioning mode</i>
RestoreState	<i>This puts the program back into relative positioning mode</i>

While the above examples only demonstrated setting the G90/G91 mode, Techno would handle all state variables in similar manner.

Mathematical Functions

Techno provides a complete set of mathematical functions for use in your G-Code file.

SIN

This function returns the sine of the supplied angle. The angle is in degrees.

Syntax:

```
REAL SIN[angle]
```

Example:

```
REAL #SineValue  
#SineValue = SIN[30]           #SineValue is set to 0.5
```

COS

This function returns the cosine of the supplied angle. The angle is in degrees.

Syntax:

```
REAL COS[angle]
```

Example:

```
REAL #CosineValue  
#CosineValue = COS[60]        #CosineValue is set to 0.5
```

TAN

This function returns the tangent of the supplied angle. The angle is in degrees. Calling this function will result in an error if the angle is 90 or 270 (or equivalent increments of 360 degrees).

Syntax:

```
REAL TAN[angle]
```

Examples:

```
REAL #TangentValue
#TangentValue = TAN[45]      #TangentValue is set to 1.0
#TangentValue = TAN[90]     Error! The tangent is infinity
```

ASIN

This function returns the arcsine of the supplied number. The angle returned is in degrees and will be from -90 to 90. The number supplied is the normalized size of the side adjacent to the desired angle and must be between -1 and 1 or an error will result.

Syntax:

```
REAL ASIN[number]
```

Example:

```
REAL #ArcsineValue
#ArcsineValue = ASIN[0.707]  # ArcsineValue is set to 45.0
```

ACOS

This function returns the arccosine of the supplied number. The angle returned is in degrees and will be from 0 to 180. The number supplied is the normalized size of the side opposite of the desired angle and must be between -1 and 1 or an error will result.

Syntax:

```
REAL ACOS[number]
```

Example:

```
REAL #ArccosineValue
#ArccosineValue = ACOS[-0.707]  # ArccosineValue is set to 135.0
```

ATAN

This function returns the arctangent of the supplied x and y values. The angle returned is in degrees and will be from -180 to 180. The x and y values supplied are the sizes of the sides adjacent and opposite (respectively) to the desired angle.

Syntax:

```
REAL ATAN[yvalue,xvalue]
```

Example:

```
REAL #ArctangentValue  
#ArctangentValue = ATAN[1,1]  #ArctangentValue is set to 45.0
```

ROUND

This function returns an integer that is the nearest whole number to the value supplied.

Syntax:

```
INTEGER ROUND[origValue]
```

Examples:

```
INTEGER #IntegerValue  
#IntegerValue = ROUND[2.5]  #IntegerValue is set to 3  
#IntegerValue = ROUND[2.49] #IntegerValue is set to 2  
#IntegerValue = ROUND[-2.5] #IntegerValue is set to -3  
#IntegerValue = ROUND[-2.49] #IntegerValue is set to -2
```

ROUNDDOWN

This function returns an integer that is the nearest whole number less than or equal to the value supplied.

Syntax:

```
INTEGER ROUNDDOWN[origValue]
```

Alternate syntax for Fanuc compatibility:

```
INTEGER FIX[origValue]
```

Examples:

```

INTEGER #IntegerValue
#IntegerValue = ROUNDDOWN[2.5]    #IntegerValue is set to 2
#IntegerValue = ROUNDDOWN[2.99]   #IntegerValue is set to 2
#IntegerValue = ROUNDDOWN[-2.5]   #IntegerValue is set to -3
#IntegerValue = ROUNDDOWN[-2.01]  #IntegerValue is set to -3

```

ROUNDUP

This function returns an integer that is the nearest whole number greater than or equal to the value supplied.

Syntax:

```
INTEGER ROUNDUP[origValue]
```

Alternate syntax for Fanuc compatibility:

```
INTEGER FUP[origValue]
```

Examples:

```

INTEGER #IntegerValue
#IntegerValue = ROUNDUP[2.5]       #IntegerValue is set to 3
#IntegerValue = ROUNDUP[2.01]     #IntegerValue is set to 3
#IntegerValue = ROUNDUP[-2.5]     #IntegerValue is set to -2
#IntegerValue = ROUNDUP[-2.99]    #IntegerValue is set to -2

```

SQRT

This function returns the square root of the value supplied. The supplied value must be 0 or greater or an error will result.

Syntax:

```
REAL SQRT[baseValue]
```

Example:

```

REAL #SquareRoot
#SquareRoot = SQRT[81]           #SquareRoot is set to 9.0

```

ABS

This function returns the absolute value of the number supplied.

Syntax:

```
REAL ABS[origValue]
```

Examples:

```
REAL #RealValue
#RealValue = ABS[2.0]           #RealValue is set to 2.0
#RealValue = ABS[-2.0]        #RealValue is set to 2.0
```

MOD

This function returns the remainder of the dividend divided by the divisor.

Syntax:

```
INTEGER MOD[dividend,divisor]
```

Example:

```
INTEGER #Remainder
#Remainder = MOD[27,4]         #Remainder is set to 3
```

POW

This function returns a value that is the base raised to the power of the exponent.

Syntax:

```
REAL POW[base,exponent]
```

Example:

```
REAL #RealValue
#RealValue = POW[4,3]         #RealValue is set to 64.0
```

LN

This function returns the natural logarithm for the value supplied.

Syntax:

```
REAL LN[baseValue]
```

Example:

```
REAL #RealValue
#RealValue = LN[4.0]           #RealValue is set to 1.38629
```

EXP

This function returns a value that is 'e' raised to the power of the exponent.

Syntax:

```
REAL EXP[baseValue]
```

Example:

```
REAL #RealValue
#RealValue = EXP[1.38629]     #RealValue is set to 4.0
```

LOG

This function returns the base 10 logarithm of the value supplied.

Syntax:

```
REAL LOG[baseValue]
```

Example:

```
REAL #LogValue
#LogValue = LOG[100]         #LogValue is set to 2.0
```

REAL

This function converts the integer value supplied to a real value.

Syntax:

```
REAL REAL[integerValue]
```

Alternate syntax for Fanuc compatibility:

```
REAL ADP[integerValue]
```

Example:

```
REAL #RealValue
#RealValue = REAL[2]        #RealValue is set to 2.0
```


Other Functions

The function `GetChangerIndex`, lets you determine the tool changer position for a given tool.

Syntax:

```
INTEGER GetChangerIndex[toolNumber]
```

Example:

```
INTEGER #Index
```

```
#Index = GetChangerIndex[2]     Get the tool changer position of tool #2
```


5. Tutorial

Starting Techno CNC

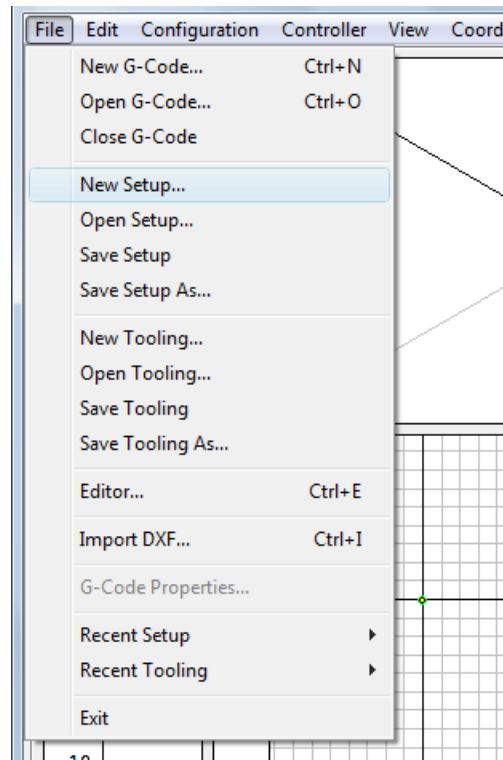
To start Techno CNC, click on the Start button, select Programs, select Techno CNC, then select the Techno CNC icon.

A dialog will appear asking you if you want to start with the Signal Generator connected. Choose the Don't Connect button. If you're running the Demo version, choose Continue.



Configuring Techno CNC

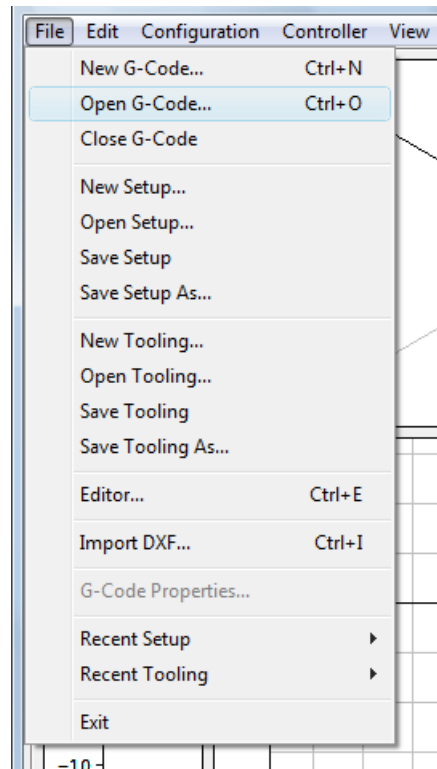
1. Choose New Setup from the File menu.



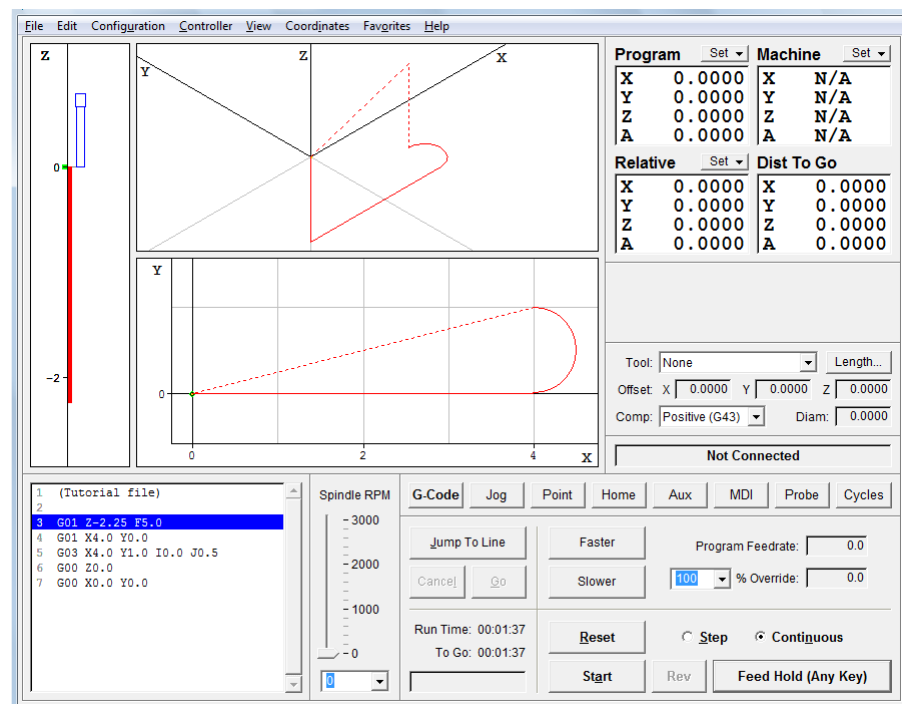
2. The configuration wizard will appear. Answer the questions in the wizard to create a default setup file for your machine. If you're using the Demo version of the software, you don't need to do anything else here. If you're using the live version of the software and running your machine tool, you'll need to finish configuring the setup file for your machine. Please refer to Initial Setup section of this manual for more details.
3. Choose DRO:All Coordinates from the View menu to make sure all four coordinate systems are displayed.
4. Choose Viewports from the Configuration menu. Set the View On checkboxes so only the following viewports are displayed: Z, XY, and XYZ.

Loading a G-Code File

1. Choose Open G-Code from the File menu.



2. Double-click on the file TUTOR.FGC.
Techno will load TUTOR.FGC and the screen should now look like this:

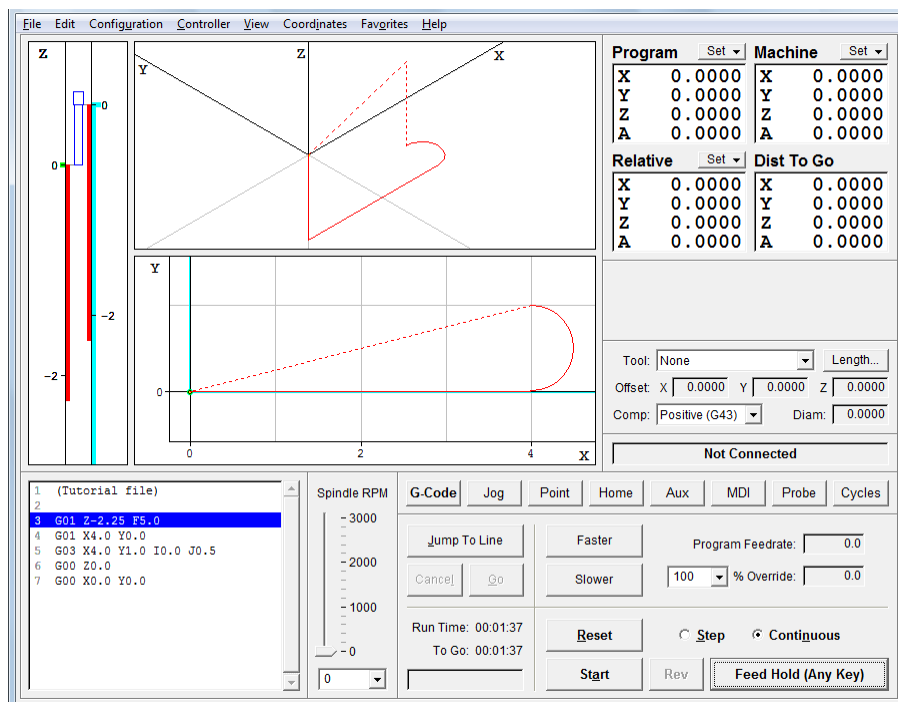


Notice how the G-Code listing appeared in the Program Listing Box and a red outline of the toolpath appeared in the Viewport Box.

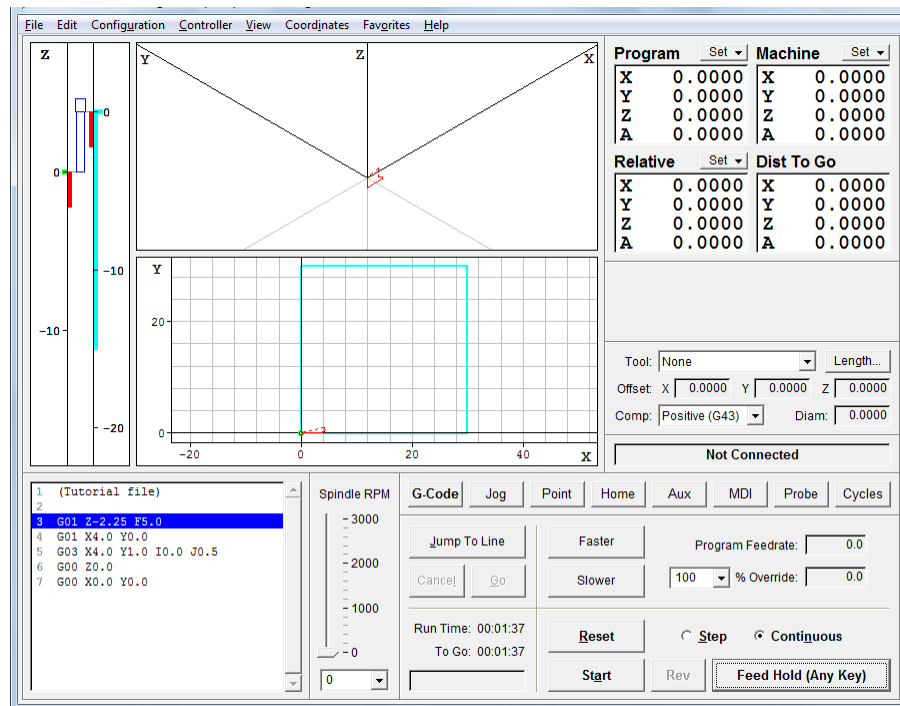
Viewing the Toolpath

Before examining the toolpath display let's zero the machine coordinates. To do this, choose Zero All from the Set button to the right of the Machine label in the DRO Box. Choose Yes in the confirmation dialog that follows.

The machine coordinates, previously shown as N/A, will now be zeroed and a light blue box will outline the tool envelope in the Viewport Box. Because Auto Zoom Toolpath Extents is checked in the View menu, you'll only see a portion of the tool envelope boundary, as shown below.



To see the entire machine tool envelope, choose Auto Zoom Machine Extents from the View menu. The screen will then appear like this:



Choose Auto Zoom Toolpath Extents from the View menu again, to see the largest possible image of the toolpath.

Now let's get familiar with the Viewport Box. Here are some important features:

Red Lines – Represent the toolpath for the G-Code program.

Green Dot – Represents Program Zero, the origin of any G-Code program.

Light Blue Dot – Represents Machine Zero, also called Home.

Yellow Dot (not shown here) – Represents the current XY position of the machine tool during the cutting operation.

Thick Blue Lines (not shown here) – Represent the portion of the toolpath already cut.

Dotted Lines – Represent a rapid move.

Solid Lines – Represent a feedrate move.

Tool Icon – Represents the current Z axis position relative to the workpiece (bottom of icon) and relative to the machine (near top of icon).

Light Blue Lines – Represent the borders of the machine tool envelope.

Animating the G-Code File

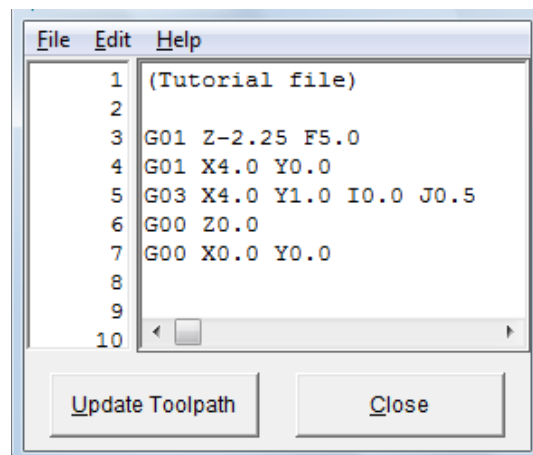
Now we're ready to animate and verify the toolpath on the screen.

1. On the Controller menu, make sure the Simulate When Not Connected command is checked.
2. Choose Zero All from the Set button next to the Program label, and then choose OK in the confirmation dialog box that follows. Techno sets all program coordinates to zero. This simulates the tool being in the correct position before the G-Code program begins.
3. Choose the G-Code button in the Control Selection Box to make sure the G-Code Control Panel is displayed.
4. Select the Step radio button so Techno will execute the G-Code program one line at a time.
5. Choose the Start button and watch the tool icon move down the Z axis viewport. Notice that Techno has highlighted the next line in the Program Listing Box, indicating it has fully executed the first line.
6. Choose the Start button again. Notice the yellow dot, which represents the current position of the tool, and the solid blue line, which represents the cutting move just executed.
7. Select the Continuous radio button, and then choose the Start button again. The yellow dot moves along the toolpath to completion, leaving a blue trace behind.

Editing a G-Code File

Now let's get familiar with using the Techno editor.

1. Choose Editor from the File menu. The editor dialog box will appear.

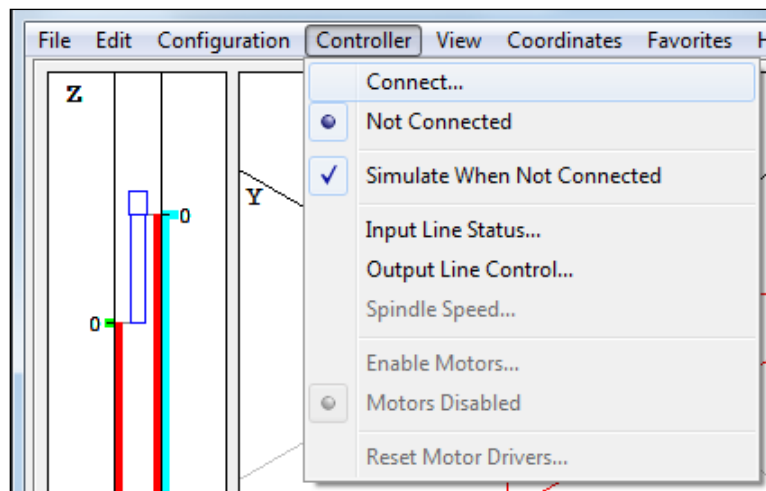


2. First, let's reduce the cutting depth from 2.25" to 0.25". With your mouse or your arrow keys, move the cursor to the first line of G-Code that reads: G01 Z-2.25 F5.0. Change it to read: G01 Z-0.25 F5.0. Then choose Update Toolpath at the bottom of the editor dialog box. Notice how the red line in the Z viewport, which shows the total length of Z travel, shrinks down to 0.25" long.
3. Now, let's change the diameter of the arc we're cutting. Move the cursor to the line of G-Code that reads: G03 X4.0 Y1.0 I0.0 J0.5. Change it to read: G03 X4.0 Y2.0 I0.0 J1.0. Choose Update Toolpath to see the effect of the change.
4. To save your changes:
 - Choose Save G-Code As from the editor's File menu.
 - Type "TUTOR2.FGC" in the File name box.
 - Choose OK.
5. To close the editor, choose Close.

Connecting with the Signal Generator

Now you're ready to have Techno establish communications with the Signal Generator. Note that when the Signal Generator is connected, all moves will be performed by the machine tool. If you're running the Demo version of Techno or do not have the means to connect to the Signal Generator at this time, ignore this section and continue with "Using the Jog Controls" below.

1. Make sure that Signal Generator is securely connected via USB cable to your PC, and you've completed the USB driver installation.
2. Turn on the Signal Generator.
3. Choose Connect from the Controller menu.



4. Techno will display the Safety First message. Please read the message carefully. If you agree with the safety guidelines, click Agree.

Using the Jog Controls

The jog controls let you manually position the tool anywhere within the machine tool envelope.

1. Choose the Jog button in the Control Selection Box. Techno displays the Jog Control Panel.
2. Select the Slow and Contin (Continuous) radio buttons. The machine tool will move continuously at the slow jog rate (defined on the Jogging panel of the Configuration dialog box.)
3. Check the machine tool to make sure there's enough room to move the tool in the Z+ direction. If there is enough room, click down, hold and then release the Z+ jog button. Notice how the tool moved up and the Z program and relative coordinates changed. Also notice how the machine moved until you released the button. If the tool went down instead of up, change the Motor Direction for the Z axis on the Motor Signals panel of the Configuration dialog box. Note that when machine coordinates are being used, Techno will not let you move beyond the machine tool envelope. Also note that you can jog the machine using the keyboard. The controls are mapped as follows:

X+	Ctrl + Right Arrow Key
X-	Ctrl + Left Arrow Key
Y+	Ctrl + Up Arrow Key
Y-	Ctrl + Down Arrow Key
Z+	Ctrl + Page Up Key
Z-	Ctrl + Page Down Key
A+	Ctrl + Plus (+) Key
A-	Ctrl + Minus (-) Key

4. Try jogging in all directions on all axes, making sure you have enough room in the direction of travel before you choose each axis jog button.
5. Now select the Fast radio button and repeat the same exercise. The tool will move at the fast jog rate defined on the Jogging panel of the Configuration dialog box.
6. Position the tool so there is at least 1" of room in the positive Z direction.
7. Now let's set the current position as program coordinates X=0, Y=0, Z=1. First, choose Enter from the Set button next to the Program label in the DRO Box. Then choose the Zero X and Zero Y buttons. Finally, enter 1.0000 in the Z text box. Choose OK to exit the dialog.

8. Now let's use the jog buttons to move the tool up exactly 1.0000". It's convenient to use three jog modes to do this: Fast Continuous, Slow Continuous and Slow Step.
9. While the jog rate is still set to Fast, move the tool up until the Z axis program coordinate is close to 2".
10. Now change the jog rate to Slow and do the same to get even closer to 2".
11. Finally, change the jog mode to Step. Techno will move the tool exactly one step each time you choose one of the jog buttons. Repeatedly choose either the +Z or -Z Axis Jog button until the tool is exactly at program coordinate 2.0000 on the Z axis. Depending on the tool positioning resolution of your machine tool, you might not be able to reach 2.0000 exactly.
12. Now use the same process to move the tool exactly 0.5000 inches in the +X direction and 0.2500 inches in the +Y direction. The program coordinates should now read:

X: 0.5000

Y: 0.2500

Z: 2.0000

Setting Machine Zero

In general you can set Machine Zero two ways: manually by jogging the tool to the corner of the machine tool envelope and choosing Zero All from the machine coordinates Set button, or by using the Seek Home button on the Home Control Panel. In this tutorial we will jog to a point and manually define it as Machine Zero. (To use the Seek Home button, your machine must have homing switches installed.)

Follow these steps to set Machine Zero manually.

1. Choose Clear from the Set button next to the machine coordinate label in the DRO Box. This will clear the machine coordinates and remove the light blue machine tool envelope boundary.
2. Choose the Jog button in the Control Selection Box.
3. Jog the tool to 1/10" from the top of the Z axis.
4. Jog the table in the X- direction to about 1/10" from the end of travel.
5. Jog the table in the Y- direction to about 1/10" from the end of travel.
6. Choose Zero All from the Set button to the right of the machine coordinate label. Then choose OK in the confirmation dialog box that follows.

Using the Point Move

The Point Control Panel provides an easier way to move the tool to an exact position.

1. Choose the Point button in the Control Selection Box.
2. Select Any Point in the Point List.
3. Select Machine from the coordinate system pull-down menu.
4. Fill in the X, Y and Z text boxes to read 1, 2, and -1.
5. Enter “4” in the Linear Feedrate text box. Now everything is ready to move to machine coordinates 1, 2, -1 at a feedrate of 4 inches/minute.

Control Selection	Machine	Feedrate
G-Code	Machine	Feedrate
Jog	X 1.000000 in	4.00 in/min
Point	Y 2.000000 in	360.0 °/min
Home	Z -1.000000 in	Move
MDI	A 0.000000 °	Feed Hold
Probe		
Cycles		

6. Choose the Move button. Notice how the machine first performed 2-Axis linear interpolation for the X and Y axis, then moved the Z axis down. This sequence of movement, unique to the Point mode and G28/G29/G30 commands, helps to avoid tool crashes.
7. Select Machine Zero in the Point List.
8. Select Rapid from the Rate Mode pull-down menu. Everything is set to move the tool to machine coordinate 0, 0, 0 as quickly as possible.

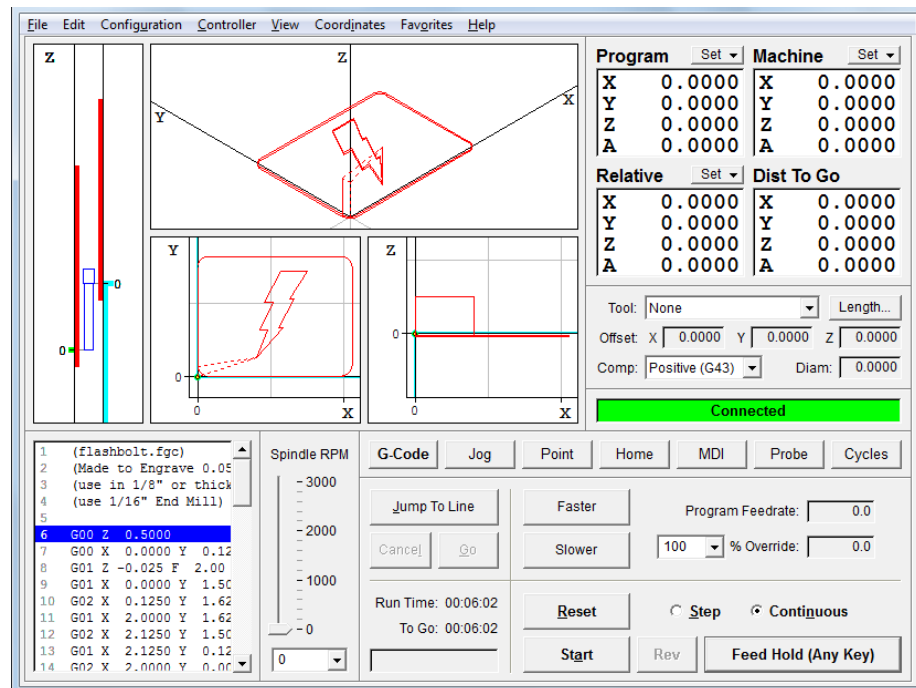
Control Selection	Machine	Rate Mode
G-Code	Machine	Rapid
Jog	X 0.000000 in	4.00 in/min
Point	Y 0.000000 in	360.0 °/min
Home	Z 0.000000 in	Move
MDI	A 0.000000 °	Feed Hold
Probe		
Cycles		

- Choose the Move button. Notice how the machine first moved the Z axis up and then performed 2-axis linear interpolation on the X and Y axes.

Setting Program Zero on the Machine Tool

Program Zero is the origin to which all program coordinates in the G-Code file are referenced. Before we cut a part, Program Zero must be set correctly relative to the workpiece. For this tutorial we will cut a file called FLASHBOLT.FGC.

- Choose Open G-Code from the File menu and select FLASHBOLT.FGC, and then choose OK. The screen appears like this:



- Make sure there is enough room on all axes of the machine to run this G-Code file from the Program Zero point. By observing the toolpath displayed in the Viewport Box, you'll see the FLASHBOLT.FGC program needs +2.125 inches on the X axis, +1.625 inches on the Y axis, +0.5 and -0.05 inches on the Z axis.
- Fixture a sheet of 1/8" or thicker plastic or aluminum at least 2" wide and 4" long onto the XY table of the machine tool. Be sure that all clamps and fixtures are well out of the way of the tool during all parts of the program. This program works best with a 1/16" end mill.
- Jog the tool to the X-, Y- corner of the workpiece using the Jog Control Panel. Then carefully jog the tool down in the Z axis until it just barely touches the top of the workpiece.

5. To define this point as Program Zero, choose Zero All from the Set button next to the Program label. Then choose OK in the dialog that immediately follows.
6. Using either the Jog or Point Control Panel, raise the Z axis exactly 1.0" and define this point as Program Zero. Note that this point is exactly 1" above where it needs to be to actually cut the workpiece. The machine is now ready to do a dry run without cutting the workpiece.

Testing the Program on the Machine Tool

It is always a good idea to do a dry run of the G-Code file, both connected and not connected to the Signal Generator, before cutting a part. This lets you check for G-Code programming errors, spot interferences with fixtures, and so on.

1. Choose the G-Code button in the Control Selection Box.
2. Choose the Reset button to make sure the G-Code program is reset. Choose OK in the dialog that follows.
3. Disconnect from the Signal Generator by choosing Disconnect from the Controller menu.
4. Go into step mode by choosing the Step radio button.
5. Repeatedly choose the Start button, watching the screen to make sure the tool behaves properly.
6. Once you are satisfied the program will behave properly, re-establish communications with the Signal Generator by choosing Connect from the Controller menu. A dialog will ask you if you want to revert to the coordinates used before disconnecting. Choose Yes.
7. Choose the Step radio button.
8. Choose the Start button. The machine should move the X and Y axes, and then stop.
9. Step through the entire program by choosing the Start button for every line of G-Code. If at any time you need to stop the tool, choose the Feed Hold button or hit any key on your keyboard. If you stop the tool in the middle of the program, you can start exactly where you left off by choosing the Start button. You may want to try this for practice.

Cutting the Part

Assuming everything was fine in the previous step, we're ready to cut a part.

1. Check to make sure the program coordinates are at 0, 0, 0. If not, go through the "Setting Program Zero on the Machine Tool" section above.

2. Go into jog mode and **carefully** move the tool down in the Z- direction to the part surface. This should be at the program coordinate 0, 0, -1.
3. Set this point as Program Zero by choosing Zero All from the Set button next to the Program label and choosing Yes in the dialog that follows.
4. Now jog the tool 0.5” **UP** (in the Z+ direction). This places the tool in the correct starting position to begin cutting the workpiece.
5. Choose the G-Code button in the Control Selection Box.
6. Choose the Continuous radio button.
7. Turn the machine tool spindle on and make sure everything is ready on the machine tool.
8. Choose the Start button and the machine tool will begin to cut out your first part. **Always be on alert to choose the Feed Hold button or hit any key in case of emergency.**
9. Congratulations! You’ve successfully cut your first part using Techno.

Exiting the Program

To exit Techno, choose Exit from the File menu. This terminates communications with the Signal Generator and returns you back to Windows.

Turning off the Electronics

Always turn off the Signal Generator and Motor Driver when they are not in use.

6. Remote Control Interface

Getting Started

Techno CNC has the ability to be controlled by an external program. This can be accomplished by writing an application that uses the Techno Remote Control Interface (FCRCI).

The FCRCI is a set of functions that have been defined and packaged in such a way as to be accessible via any .NET language program. This interface is in the form of a managed DLL called Remote.DLL. The classes defined in Remote.DLL contain useful functions that allow the remote application to control various aspects of the Techno CNC program.

The steps for using an application to control Techno CNC are simple.

1. Obtain Remote.DLL and RemoteLoader.DLL from Techno. See support.technocnc.com or call technical support for more information. Place both of these files in the same folder as the Techno CNC application (usually something like C:\Program Files\Techno CNC 4).
2. Write a .NET application that exercises the part of the FCRCI that you desire. You can use any .NET language you wish.
3. Place your application in the same folder as the Techno CNC application.
4. Run Techno CNC.
5. Run your application.

When your application runs, it should attempt to connect to the Techno CNC application. If the attempt is successful, your application will be able to control Techno CNC.

Two example programs are included in the FCRCI package: one written in Visual Basic and the other in C#. These samples should give you the information you need to build your own remote control program.

The namespace that contains all classes in the FCRCI is called CNCRemote. In the CNCRemote namespace, the main class is API. The API class contains most of the functions and therefore is the object through which most activity will take place. When writing your application, you must first set up .NET remoting (examples of how to do this are in both sample programs). Once .NET remoting is set up, you can obtain an instance of the API class. Finally, after getting the API object, you can call the Connect() function on it. This establishes the connection to the Techno CNC program.

An additional step that was added is to call Ping() after the connection has been established. Testing has shown that the first function called after connecting via .NET remoting usually takes much longer to execute than all subsequent

functions. For this reason, a function call was added to “prime the pump” and get that time delay out of the way.

Functions

Once you have an API object, you can call any of the functions listed below on your API object:

void Connect(Responder^ responder)

Description: This function attempts to connect the remote application to the Techno CNC program.

Parameters: responder – A handle to the object that will receive response messages from Techno CNC.

void Disconnect(Responder^ responder)

Description: This function attempts to disconnect the remote application to the Techno CNC program.

Parameters: responder – A handle to the object that is receiving response messages from Techno CNC.

void LoadGCodeFile(System::String^ fileName)

Description: This function loads the G-code file specified by the supplied fileName. Any errors that occur while loading the file are reported in the Techno CNC program

Parameters: fileName – The full path to the file to be loaded.

void StartGCode()

Description: This function causes the G-code file loaded in Techno CNC to run.

void Feedhold()

Description: This function causes execution of a G-code file in Techno CNC to pause.

void Move(CommonTypes::Position^ position, CommonTypes::PointMoveType iMoveType, bool bIsRapid, double dFeedrate)

Description: This function causes Techno CNC to move to the specified point in the manner described by the parameters supplied.

Parameters: position – The position to which the system is to move.

iMoveType – The type of move (incremental, program, machine, or relative).

bIsRapid – An indication of whether a rapid or feedrate move is desired.

dFeedrate – The feedrate at which a feedrate move is to take place. Must be greater than 0, even if this is a rapid move.

CommonTypes::Position^ GetCurrentMachineCoordinates()

CommonTypes::Position^ GetCurrentProgramCoordinates()

CommonTypes::Position^ GetCurrentRelativeCoordinates()

Description: These functions return the current coordinates in the appropriate coordinate system.

void Ping(Responder^ responder)

Description: This function causes Techno CNC to send a message back to the remote application.

Parameters: responder – A handle to the object that will receive the ping response message from Techno CNC.

Messages

Messages are used to communicate from the Techno CNC system to the remote application. Each message contains a specific ID that indicates the purpose of the message. Useful message IDs that are sent to the remote application are listed below:

- 501 – A feed hold occurred (motion was stopped).
- 706 – The G-code program is loaded.
- 708 – Program execution is complete.
- 731 – The move to a specific point completed successfully.
- 746 – Ping response code.
- 1001 – An error occurred while loading a G-code file.
- 2000 – A limit switch was hit while motion was occurring. This is an error code. This code will not occur during homing.
- 2014 – A safety interlock was triggered. This will only occur if your system has interlocks configured.

- 99999 – An error occurred while loading a file. The system may be busy.

Additional messages that are sent, but are of limited use, are:

- 702 – Motion mode update (rapid vs. feedrate, line vs. arc)
- 716 – The feedrate has been updated.
- 717 – The feedrate override has been updated.
- 720 – Motion has stopped.
- 725 – Progress update (percentage completion).
- 738 – Controller status update (eventually will contain position data).
- 742 – Run time estimate update.
- 900 – Preview message (for displaying the toolpath).

By responding to the messages that are sent from Techno CNC, you can ensure that your remote application will operate in harmony with Techno CNC.

7. Glossary

- Backlash** – The amount of motor movement that occurs without table movement when changing directions on an axis. A typical source for backlash is play between the screw and nut in a leadscrew drive system.
- Backlash Compensation** – When an axis changes direction, a small turn of the motor to move the drive mechanism through any mechanical play, before resuming motion.
- Connected** – Mode in which Techno communicates with the Signal Generator. In this mode, all G-Code, Jog and Point moves are executed by the machine tool.
- Continuous Contouring** – An advanced feature that lets Techno run G-code files smoother and faster. Smooth transitions from one G-code line to the next allow high speed machining of complex 2D profiles and 3D surfaces. Also called “Look-Ahead” and “Constant Contouring.”
- DXF** – Document eXchange Format. Defined by AutoDesk Inc. as a way to exchange design data between CAD, CAM, CAE and CNC programs.
- DRO Box** – The area of the main screen that shows the current location of the machine tool. You can display one or all of the four coordinate systems in this box. ‘DRO’ is an acronym for Digital Readout.
- Feedrate** – The speed of the cutting tool relative to the workpiece. Defined in G-Code by the F parameter in inches/minute, millimeters/minute or degrees/minute.
- Full-Step** – Step mode where one step from the Signal Generator corresponds to one full step of a stepper motor.
- G-Code** – Standard programming language used to control CNC machines.
- G-Code Program** – Program used to control the movement of the machine tool. G-Code is an industry-standard machine tool programming language.
- Half-Step** – Step mode where two steps from the Signal Generator correspond to one full step of a stepper motor.
- Home** – Same as Machine Zero (see below).
- Included File** – G-Code file containing subroutines that can be called from a main G-Code file. The included file must be listed at the top of the main G-Code file.
- Jog** – Method of manually controlling each axis of the machine tool.
- Limit Switches** – Switches placed at the travel limits of each axis. When the machine tool travels too far in either direction of any axis, a limit switch is tripped. The control system then stops the machine to prevent damage.
- Machine Coordinates** – The position of the tool relative to Machine Zero.
- Machine Tool Envelope** – The three dimensional box (plus optional 4th and 5th axis travel) defined by the maximum travel in the X, Y, Z, A, B and C axes. Once the machine tool envelope is defined, the tool cannot move beyond it. You can disable the machine tool envelope by choosing the Clear button.

- Machine Zero** – The origin of useful space within the machine tool envelope. Can either be defined manually or by using home switches.
- Maximum Feedrate** – The highest possible feedrate for a given axis, which generally requires ramping.
- Micro Steps** – Micro steps occur when a micro stepping driver subdivides each full step electronically into a number of smaller steps.
- Modal** – A type of command that stays in effect while Techno processes subsequent G-Code commands.
- Motor Driver** – The electronics that convert step and direction signals from the Signal Generator into a sequence of amplified signals to drive a motor.
- Motor Direction** – The association between the actual direction an axis moves and the direction Techno intends it to move. If an axis moves in the wrong direction, this setting should be changed from positive to negative or vice-versa.
- Motor Resolution** – The number of full motor steps for one revolution of the motor. For example, a 1.8° Stepper motor has 200 full steps per revolution; a 0.9° Stepper Motor has 400 full steps per revolution, and so on.
- Motor Step** – The amount of motor movement associated with one electrical pulse to the stepper motor driver.
- Not Connected** – Mode in which Techno does not communicate with the Signal Generator. Techno displays all G-Code, Jog, and Point moves, but the machine tool does not move.
- Open Loop** – A type of control system in which an actuator moves without sending motion information back to the controller. Most stepper motor systems are open loop due to their high reliability in performing step commands when used within their torque limits.
- Program Coordinates** – The position of the machine tool relative to Program Zero.
- Program Listing Box** – The area of the main screen that displays the G-Code program currently loaded into Techno.
- Program Zero** – The zero point, or origin, to which all absolute coordinates in the G-Code file are referenced. It is represented by a green dot in the viewports.
- Ramping** – Smooth acceleration that allows a machine tool to reach high feedrates. Ramping rates are measured in full steps/sec/sec.
- Relative Coordinates** – The position of the machine tool relative to the point at which the relative coordinates were zeroed. The relative coordinate system is general purpose and may be used for anything you choose (as long as it's legal).
- Resonant Speeds** – Rotational speeds at which a stepper motor vibrates excessively. Often the motor will stall if run at these speeds. Resonant speeds depend on the size of the motor, the amount of loading, and the power of the motor driver. Typically, increasing the load and reducing motor driver current will reduce resonance.
- Safety Interlock** – Logic to prevent pairs of digital output lines from reaching an incompatible on-off state.
- Serial Port** – A communications port on both the PC and the Signal Generator, used to exchange commands and other information.

Setup File – A file containing the CNC Setup Parameters for a machine tool. These files have an “STP” extension.

Signal Generator – The electronic box that converts computer commands into step and direction signals suitable for a stepper or digital servo motor driver. It also interprets input line signals and produces output signals to control various peripheral devices.

Start/Stop Feedrate – The maximum rate at which a motor can reliably start and stop without ramping.

Step Mode – The number of micro steps between each full motor step.

Stepper Motor – A motor that moves a precise amount when given an electrical pulse. Stepper motors typically have 200 full steps per revolution, or 1.8° per full step. Other popular stepper motors have 0.9° and 7.5° per step.

Tool Positioning Resolution – The amount of machine tool movement that results from one step pulse produced by the Signal Generator.

Tooling File – A file containing the geometry information for up to 100 tools. These files have a “TLG” extension.

Toolpath – The path that a machine tool follows as Techno executes a G-Code program.

Viewport Box – The area of the main screen that graphically displays the toolpath and updates the machine position in real time.