NCPlot™ v2.33
User Manual

Table Of Contents

Welcome

Welcome to NCPlot!

NCPlot™ v2.33
Copyright© 2005-2016
NCPlot Software LLC

For the latest release information, news, or if you can’t find what you need in this help file, please check the online forums at:

www.ncplot.com

We can also be contacted directly at:

Email: scottmartinez@ncplot.com

Registering NCPlot

When NCPlot is first installed, you are given a 15-day trial period. During this time the software is fully functional, allowing you to evaluate it's suitability for your needs.

After the trial period has expired, NCPlot will no longer run without a registration key. These may be purchased from the online ordering page at http://www.ncplot.com/ The web site contains current pricing and ordering information.

To register NCPlot, enter your registration name and product key into the splash window. This window is displayed every time NCPlot starts, but may also be accessed from the menu ***Help / About***. Note that the splash window is not displayed at startup after the software has been successfully registered.

3Dconnexion™ Devices

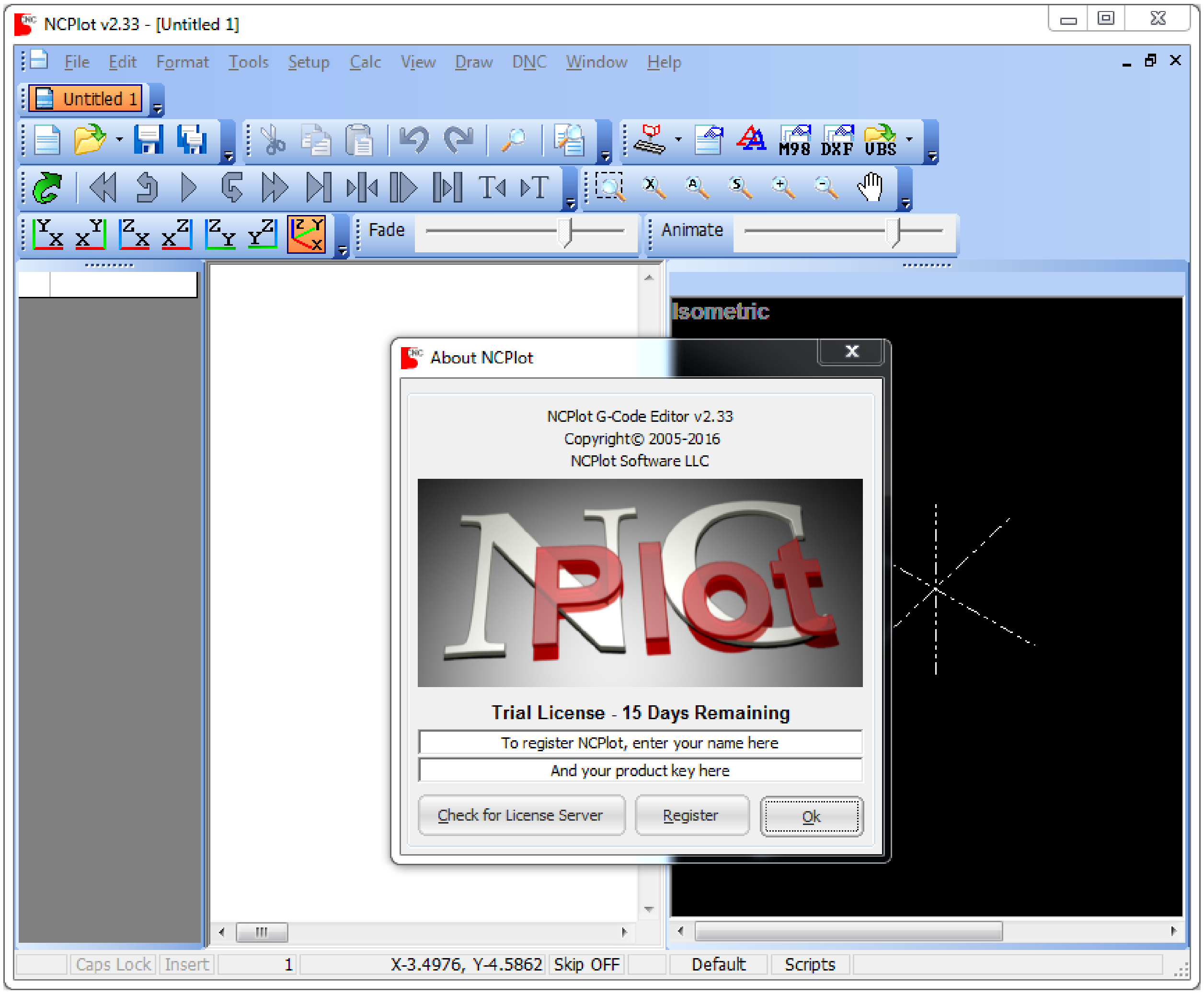
If you have a 3Dconnexion™ device such as a SpaceNavigator™, SpaceExplorer™ or SpacePilot™, NCPlot will recognize and allow you to use this device to pan, rotate and zoom the viewport.

NCPlot provides a configuration dialog that allows you to customize your 3D device, see the menu ***Setup / 3Dconnexion™ Setup***.

Getting Started

Getting Started

On startup, you are greeted with the following screen. If you have not registered the software, a window displays your remaining trial period time. You may click Ok to continue unregistered, or you may enter your name and key information.



NCPlot G-Code Editor v2.33
Copyright© 2005-2016
NCPlot Software LLC

Trial License - 15 Days Remaining

To register NCPlot, enter your name here

And your product key here

The NCPlot window consists of:

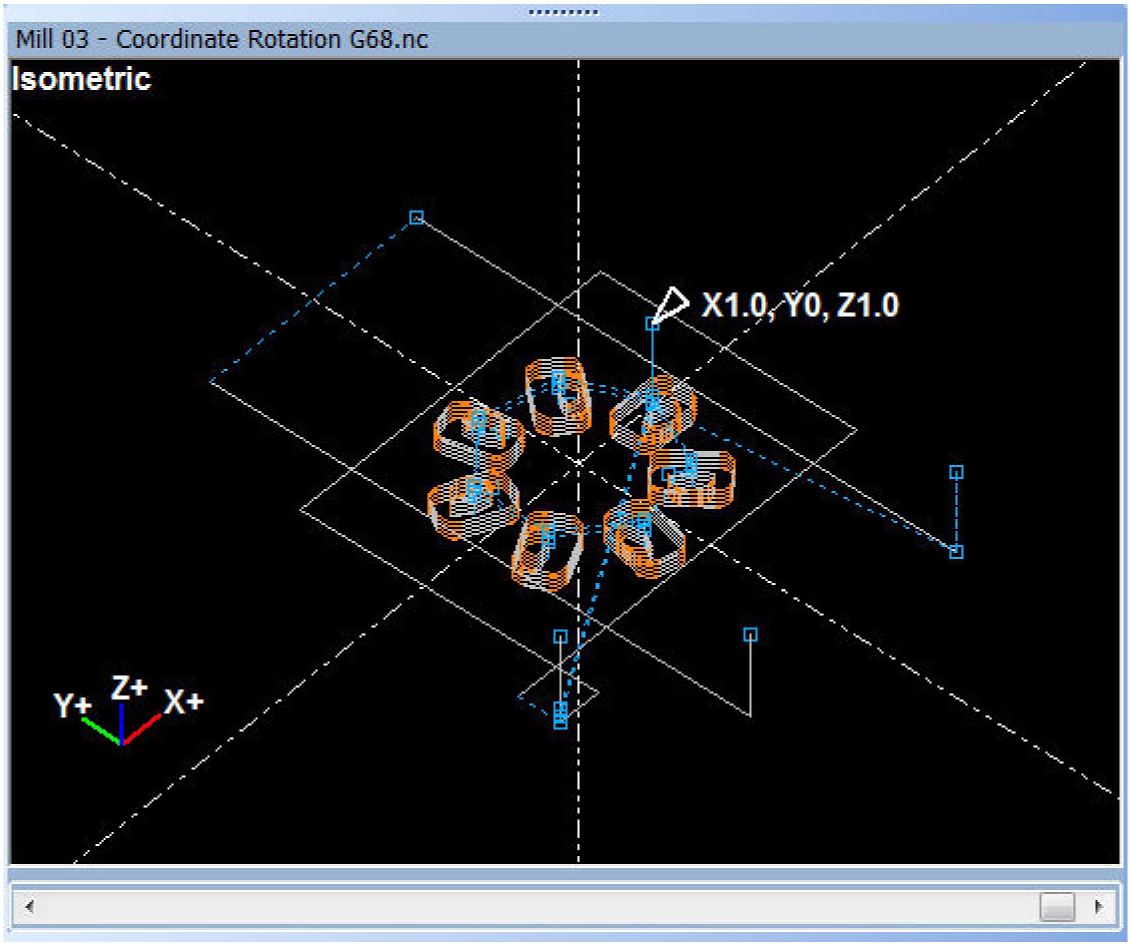
- The document workspace, where loaded G-Code programs are shown.
- The graphics viewport, where the graphic backplot appears.

- The menu bar, which is used to access most of the program features.
- The toolbars contain shortcut buttons to the most commonly used functions.
- A status bar which displays information about the current state of NCPlot.

The Viewport

The graphics viewport in NCPlot displays the graphical representation of the programmed G-Code tool path. The viewport not only shows you if your program will work as expected, but it also provides help when the results are not what you expected. Using the dynamic pan, zoom and rotate you can quickly spot parts of the tool path that are not correct. The plot controls such as animate, step forward and step backward can then help you locate the trouble spots in the program.

The viewport is a dockable toolbar that may be docked on either the left or right side of the application window. It can also be undocked from the application window and moved anywhere on your desktop. As a toolbar it may also be closed altogether.



In addition to the tool path graphics, there are several other useful items on the viewport:

- **File Name** - the top of the viewport displays the name of the file the backplot corresponds to.
- **View Name** - appears in the top left corner of the viewport and describes the active view orientation. This gives a reminder of which way the view is looking at the part. The color of the view name indicates whether the viewport is currently active, gray meaning it is not active. While active the view name will be either white or black depending on the background color of the viewport.
- **Orientation Icon** - like the View Name, this icon serves to show you which way the part is oriented on the viewport. The icon appears in the lower left corner of the viewport and indicates the positive direction for each displayed axis.
- **Axis Lines** - are drawn to indicate where the active zero point is. The zero point can represent the machine zero or any of six programmable work zero locations.
- **Marker Icon** - this icon is an arrow shaped pointer that appears on the viewport and shows the location of the active block endpoint. In addition to marking the location on the viewport, the marker also displays the coordinates of the active

point and a selectable group of modal address values. See Machine Configuration / Viewport Settings.

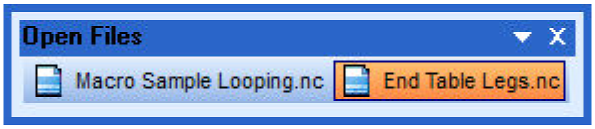
- **Viewport Slider Control** - the slider can be dragged with the mouse to quickly advance or rewind the plot to any point in the program. Likewise, when stepping or animating the graphics the slider moves to show the current progress.

Toolbars

The NCPlot toolbars give you quick access to the most common functions by grouping them together as buttons. The toolbars may docked into the application window on the top, left, bottom or right side. They may also be undocked and placed anywhere on your desktop, or they may be closed altogether. The toolbars may be rearranged to your liking by clicking and dragging the control handle on the left side of each toolbar.

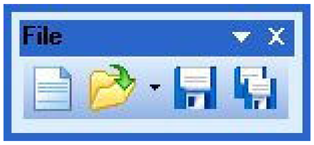
Most of the toolbars may be customized by adding buttons or hiding buttons or even creating your own custom toolbars. See Customizing the Toolbars.

Open Files toolbar



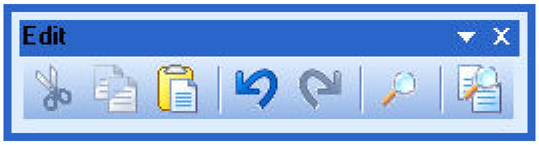
The ***Open Files toolbar*** displays a list of the currently open files. Clicking file names in the list will switch the active edit file to the clicked file.

File toolbar



The ***File toolbar*** contains buttons for commonly used file functions. The ***File Open*** button also contains the list of recently opened files. This list may be cleared with the menu item ***File / Clear Recent Files***.

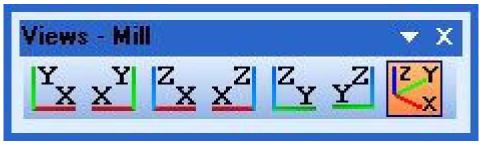
Edit toolbar



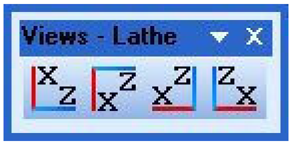
The ***Edit toolbar*** contains buttons for the clipboard operations cut, copy and paste as well as undo, redo, find and compare.

View toolbar

Mill Views

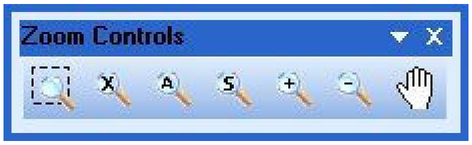


Lathe Views



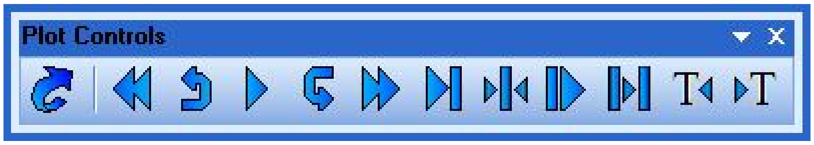
The ***View toolbar*** contains buttons for changing the selected viewport orientation. The displayed toolbar is dependant on the currently configured machine type. There are seven view buttons for the Mill configuration and four for Lathe.

Zoom toolbar



The ***Zoom toolbar*** contains buttons that change the graphic display size and location. There are buttons for ***zoom window, zoom extents, zoom all, zoom selected, zoom in, zoom out*** and ***pan***.

Plot toolbar

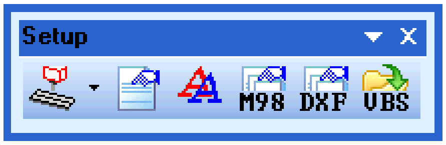


The ***Plot toolbar*** contains buttons that allow you to control the plotting of your program. The buttons from left to right are:

- ***refresh plot***
- ***rewind plot to beginning***
- ***step backward***
- ***animate***
- ***step forward***
- ***fast forward plot to end***
- ***plot to cursor***
- ***start at cursor***
- ***plot from cursor***
- ***plot selected blocks only***
- ***plot backward to previous tool***
- ***plot forward to next tool***

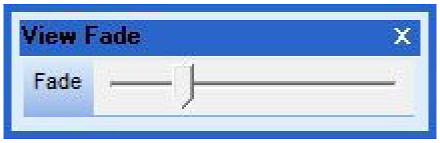
When editing your program, the refresh plot button will change from blue to green to indicate that the graphics should be refreshed. You do not need to save the program before refreshing.

Setup toolbar



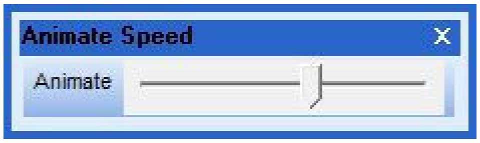
The ***Setup toolbar*** provides shortcuts to most of the NCPlot configuration settings including ***Machine Configuration, Preferences, Address color settings, Subprogram settings***, and ***DXF conversion settings***. It also provides quick access to your ***VBS scripts***.

View fade toolbar



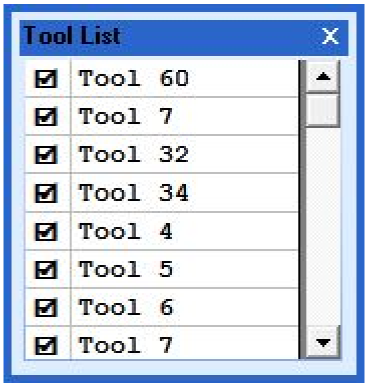
The NCPlot viewport provides View Fading, which allows you to dim the parts of the backplot that are not part of the current tool. The ***View Fade toolbar*** provides a slider that controls the brightness of the faded entities.

Animate toolbar



The ***Animate toolbar*** provides a slider that controls the speed of the backplot animation.

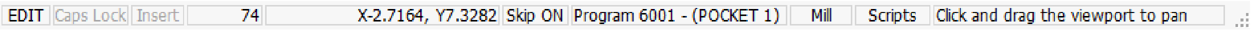
Tool list toolbar



☑ Tool 60
☑ Tool 7
☑ Tool 32
☑ Tool 34
☑ Tool 4
☑ Tool 5
☑ Tool 6
☑ Tool 7

The ***Tool List toolbar*** provides a list of the tools used by the active program. For more info please see the Tool List Toolbar topic.

Status Bar



The status bar is made up of panels. Some panels contain information about the current state of NCPlot, while some allow quick access to NCPlot features. The panels from left to right are:

- **Edit Status** indicates that changes have been made to the loaded file.
- **Caps Lock Status** indicates when the keyboard caps lock is active.
- **Insert Status** indicates when the keyboard insert is active.
- **Current Line Number** indicates the line number that the cursor is on.
- **Position** panel indicates the location of the mouse pointer in machine coordinates. This location will only update while the mouse pointer is within the viewport.
- **Block Skip** Status indicates the current status of the block delete toggle. Clicking this panel toggles between the Skip ON and Skip OFF state. This toggle is equivalent to the Block Delete switch you would find on your machine control panel. When ON, this toggle causes NCPlot to ignore (skip) program blocks that begin with the block delete character “/”.
- **Active Program** indicates the name of the currently selected program. This panel is used when there is more than one program in your edit file. Clicking this panel will display a list of all programs in the file and allows you to select the program you wish to backplot.
- **Current Configuration** indicates the name of the currently loaded machine configuration settings. Clicking this panel will display a list of the available configurations. Click one of the items on this pop up list and the selected configuration is then loaded.
- **Scripts** panel is a shortcut to the available script files. Clicking this panel will display a list of the available scripts. Click one of the items on this pop up list and the selected script is executed. The available scripts are stored in the folder "\Program files\NCPlot\Scripts"
- **Messages** panel displays messages related to the active operation.

Tool List Toolbar

The tool list toolbar displays a list of the tool numbers used by the program and is updated whenever the graphics view is refreshed.

A check mark appears next to each tool number in list. Clicking the check mark will toggle its checked state. By unchecking a tool its corresponding viewport graphics may be hidden. The plotting tools will skip over any hidden tool paths and hidden tool paths will not be included when printing the viewport or exporting a backplot as DXF.

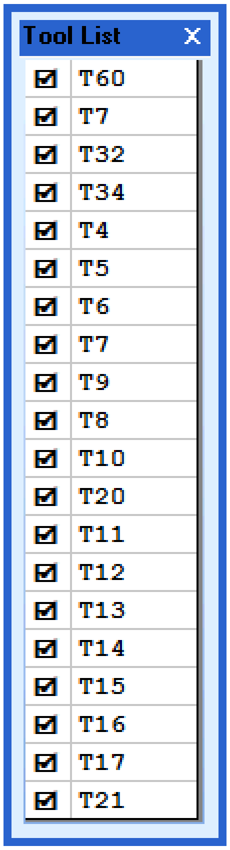
Right clicking on the tool list brings up a menu with these options:

- **Show All** - Checks all of the tools in the list.
- **Show Selected** - Checks the selected tools in the list.
- **Show Only This Tool** - Checks the tool number at the mouse pointer and unchecks all others.
- **Select This Tool** - Selects the region of the program that corresponds to the tool number being pointed to.
- **Highlight This Tool** - Selects the region of the backplot graphic that corresponds to the tool number being pointed to.
- **Hide All** - Unchecks all of the tools in the list.
- **Hide Selected** - Unchecks the selected tools in the list.

Multiple tools in the list may selected by clicking and dragging the mouse pointer on the tool list. When the mouse button is released a menu appears which gives you the choice of either hiding or showing the selected items.

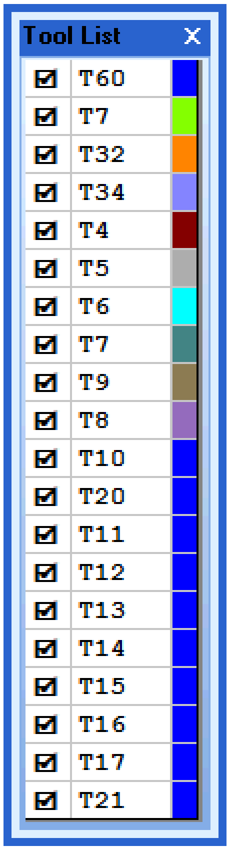
The tool list also acts as bookmarks, clicking on a tool number will highlight the block in the program where the tool was commanded. You can also plot the program up to a selected tool by holding the 'Ctrl' key and clicking a tool number.

Example tool list when 'Color by G-Code' machine configuration setting is selected:



- ☑ T60
- ☑ T7
- ☑ T32
- ☑ T34
- ☑ T4
- ☑ T5
- ☑ T6
- ☑ T7
- ☑ T9
- ☑ T8
- ☑ T10
- ☑ T20
- ☑ T11
- ☑ T12
- ☑ T13
- ☑ T14
- ☑ T15
- ☑ T16
- ☑ T17
- ☑ T21

Example tool list when 'Color by Tool' machine configuration setting is selected:



Tool List

☑	T60	🟦
☑	T7	🟩
☑	T32	🟧
☑	T34	🟪
☑	T4	🟥
☑	T5	⬜
☑	T6	🟦
☑	T7	🟩
☑	T9	🟫
☑	T8	🟪
☑	T10	🟦
☑	T20	🟦
☑	T11	🟦
☑	T12	🟦
☑	T13	🟦
☑	T14	🟦
☑	T15	🟦
☑	T16	🟦
☑	T17	🟦
☑	T21	🟦

Viewport Pan, Rotate and Zoom

The viewport can easily be manipulated using just the mouse or keyboard, no buttons or commands are required to activate these functions. The controls vary slightly between the 2D and 3D views. The 3D view refers to the Isometric view. This view mode is only available for Mill machine configurations. All other view modes are 2D views.

If you have a 3Dconnexion device such as a SpaceNavigator™, SpaceExplorer™ or SpacePilot™, NCPlot will recognize and allow you to use this device to pan, rotate and zoom the viewport.

Pan and Rotate for Isometric (3D) view

Pan - Press and hold the "Shift" key and the right mouse button while moving the mouse.
Pan - Press and hold both the left and right mouse buttons while moving the mouse.

Pan - Press and hold the mouse wheel button (if your mouse has one) while moving the mouse.
Rotate - Press and hold the right mouse button while moving the mouse.

The view will rotate in two directions. Moving the mouse left or right rotates around the viewport vertical axis, and moving the mouse up or down rotates around the viewport horizontal axis. The up and down rotation can be locked by checking the menu item ***View / Lock Vertical Rotation***. When this is checked you can temporarily unlock it by holding the Ctrl key and rotating the view.

The view will rotate about a point at the X, Y center of the viewport and at the negative Z extent. This rotation point can be moved with the tool ***View / Set ISO View Rotation Center***. This allows a point on the backplot to be picked as the new rotation center.

Panning for all other views

Press and hold the right mouse button while moving the mouse.

Zooming

In all view modes rolling the mouse wheel will zoom in or out depending on the direction the wheel is rolled. The zoom direction can be reversed on the Preferences window by checking the setting ***Reverse mouse wheel zoom direction***.

If your mouse has a middle button you can double click it to zoom to extents.

Viewport Keyboard Shortcuts

The viewport allows the use of keyboard shortcut keys to activate all of its most commonly used functions. The shortcut keys require that the viewport be active. The color of the viewport view name indicates whether the viewport is active, gray meaning it is not active. The viewport can be made active by:

- Clicking on the viewport
- Clicking the menu ***Window / 0***
- The menu shortcut keys **Alt-W / 0**
- The viewport shortcut key **Alt-V**

While the viewport is active the following keyboard shortcuts are available:

- **Zoom Window** - "Z" Key - Allows dragging a box around an area to fit into the viewport.

- **Zoom Extents** - "X" Key - Fits the part drawing into the viewport, including rapid motions.
- **Zoom All** - "A" Key - Fits the part drawing into the viewport, disregarding the rapid motions.
- **Zoom Selected** - "F" Key - Fits only the selected entities into the viewport.
- **Zoom In** - "C" Key - Increases the zoom magnification, making the part appear larger.
- **Zoom Out** - "V" Key - Decreases the zoom magnification, making the part appear smaller.
- **Pan** – “P” Key – Activates the viewport pan tool.
- **Step Forward** – “S” Key – Draws the next motion block in the program.
- **Step To Next Tool** – “T” Key – Draws up to the next tool change in the program.
- **Step Backward** – “B” Key – Un-draws the previous motion block.
- **Step To Previous Tool** – “R” Key – Un-draws back to the previous tool change in the program.
- **Measure** – “M” Key – Activates the measure tool.
- **Set ISO View Rotation Center** - "Q" Key - Allows selecting a point to be the isometric view rotation center.

Selecting Entities on the Viewport

The ability to select the entities that make up the backplot of your part is an extremely useful feature of NCPlot. When an entity is selected its color changes to the selection color, and a small square is drawn around its endpoint. The marked endpoint then gives a visual indication of the direction that the entity will machine in. There are three selection methods:

- **Clicking** - Left clicking any of the entities on the viewport will do three things.
    1. It will select the entity.
    2. The block in the program that created it will be highlighted.
    3. The entity info window will show the properties of the selected entity.

  Multiple entities may be selected by holding the Shift key while left clicking additional entities. Entities may also be deselected by holding the Ctrl key while left clicking selected entities.

- **Window selecting** - Groups of entities can be selected by simply dragging a box around them. Click and hold the left mouse button at one corner of a box and drag the mouse and release the button at the opposite corner. A box will be drawn as the mouse is moved to indicate the area containing the entities to be selected. The box will have either a solid line border or a dashed line border depending on which direction you drag the box. A solid line border appears when you drag to the right and will select everything that is completely inside the box when the mouse button is released. A dashed line border appears when you drag to the left and will select everything that is inside of or touching the borders of the box when the mouse button is released. Holding the Shift key while window selecting will add the selected entities to the current selection set.

- **Calc tools** - The "Calc" menu contains additional tools that make it easier to select groups of entities. For example, the "Select Chain" tool is useful for selecting an entire 2D profile. Just select one entity that is part of the profile, then select this tool. All entities that are connected to the selected one and at the same Z depth are selected automatically. The "Select by Z" tool will go through the entire backplot and select all entities that are at the same Z depth as the currently selected entity. This makes it easy to select all profiles that are at a given depth. The "Select Arc by Radius" tool will select only arc entities that are the same radius as the currently selected arc. You could then create points at the center of each selected arc and use these points to create a drill program.

After a group of entities has been selected, there are a few things that you can do with them:

- **Export as DXF file** - The File menu has an option called Export Selected as DXF File that will enable you save a DXF file that contains only the entities that you have selected. This can save a lot of work deleting unnecessary geometry from a drawing that contains the entire backplot.

- **Calc Tools** - The Calc tools such as offset and blend radius require that one or more entities be selected. These tools are applied only to the selected entities.

- **Convert to G-Code** - This tool will use the selected entities to create new G-Code snippets. This might not seem very useful at first, after all the selected entities were created from G-Code in the first place, right? Not necessarily, the Calc tools can be used to create new geometry which you can then turn into new G-Code with this tool. You could also take a backplot from a simple 2D profile and use it to create multiple Z passes.

- **Delete** – Pressing the DEL or Delete key on your keyboard will remove the selected entities from the viewport. This is useful for removing clutter when trying to isolate a particular area of the backplot. This does not remove them from your program and refreshing the viewport will restore the deleted entities.

Plot Settings

There are many settings that can affect the way your program will be plotted. Most settings can be found on the Machine Configuration dialog. This is found on the menu ***Setup / Machine Configuration***.

These options are found on the ***Draw*** menu. Each of these options will display a check mark next to them on the menu to indicate that they are enabled.

Axis Lines

Displays two or three (depending on the view) intersecting lines that indicate where 0,0,0 is on the viewport. The axis lines display may indicate the machine zero or any of the six work offsets.

Show Rapid Moves

Displays or hides rapid motion lines on the viewport.

Show Ticks

When enabled, a small square is drawn at the endpoint of rapid moves. This setting has no effect when Show Rapid Moves is not enabled.

Show Plunge Moves

When disabled, this option will hide Z axis movements in the negative direction. This only affects movements that are Z axis only.

Absolute Arc Centers

When checked, this option specifies how I / J / K specified arc centers are drawn. This option also affects the results of the arc conversion tools. This setting may also be changed on the Machine Configuration dialog under the Control Options tab.

Preferences

The preferences dialog contains some general settings that allow customizing the way NCPlot behaves. Please see the menu ***Setup / Preferences***.

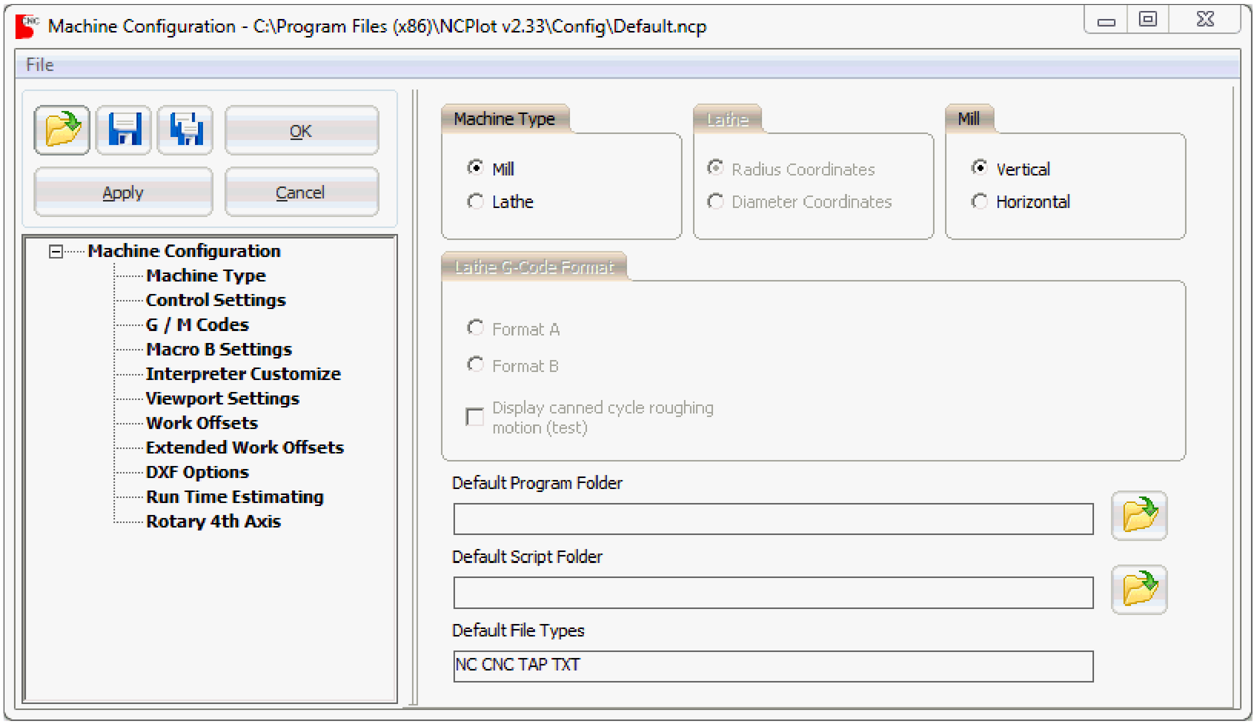
Machine Configuration

About the Machine Configuration

In order for the graphics viewport to properly display your G-Code program, it must first know a few things about the machine you intend to run it on. Since there are many different types of machines and CNC controls, NCPlot has options that allow it to mimic the way your particular CNC control reads G-Code. NCPlot doesn't recognize every G or M Code that your control does, but it should still be able to give you a good representation of your programs toolpath. Besides backplotting, the machine configuration is important for another reason. Some of the conversion tools require that the backplotter be properly configured in order to give the desired results. For example, if the arcs in your program do not look correct when plotted, the arc conversion tools will not work correctly. In general, if the plot looks correct the conversion tools will work the way they're supposed to.

NCPlot comes with a handful of predefined machine configurations. These configurations represent the most common settings for a CNC control and should be good enough to get you started. Even so, you should check that these settings match the way your control works.

To open the machine configuration dialog, click the menu ***Setup***, then click ***Machine Configuration***. This dialog is made up of several pages, the first page you see is labeled ***Machine Type***. This page has settings that define the basic setup of your machine.



Machine Configuration - C:\Program Files (x86)\NCPlot v2.33\Config\Default.ncp

Machine Type

● Mill
○ Lathe

Lathe

● Radius Coordinates
○ Diameter Coordinates

Mill

● Vertical
○ Horizontal

Lathe G-Code Format

○ Format A
○ Format B
□ Display canned cycle roughing motion (test)

Default Program Folder

Default Script Folder

Default File Types: NC CNC TAP TXT

In addition to the configuration settings, there are a number of buttons that are for managing your configurations:

Open allows you to browse for and open an existing configuration.

Save will apply the current changes and save them to the active configuration.

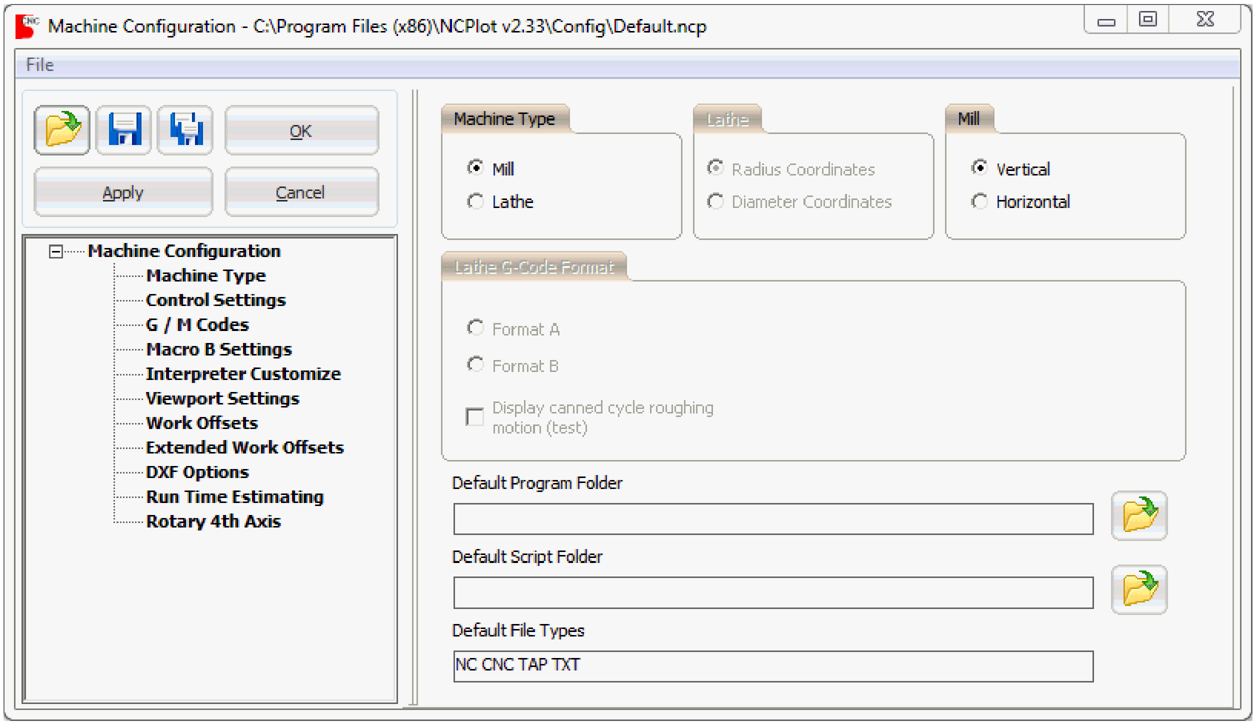
Save As will allow you to specify a new configuration file and save your settings. Note that when a new configuration is created, its canned cycles are copied from the currently active configuration.

OK accepts the current configuration changes and closes the configuration dialog.

Apply will apply the current configuration changes without closing the configuration dialog.

Cancel will close the configuration dialog discarding any changes.

Machine Type



Machine Configuration - C:\Program Files (x86)\NCPlot v2.33\Config\Default.ncp

Machine Type
● Mill
○ Lathe

Lathe
● Radius Coordinates
○ Diameter Coordinates

Mill
● Vertical
○ Horizontal

Lathe G-Code Format
○ Format A
○ Format B
□ Display canned cycle roughing motion (test)

Default Program Folder

Default Script Folder

Default File Types: NC CNC TAP TXT

The most basic configuration setting is the Machine Type, you should first select between Mill and Lathe before configuring the remaining settings.. Choosing one or the other will change or enable/disable other settings on the dialog. If you selected Mill, you now have the option to select between Vertical spindle and Horizontal spindle. If you selected Lathe, you now have the option to select between ***Radius Coordinate*** values and ***Diameter Coordinate*** values. This setting determines how NCPlot interprets the X/U axis command values. The Lathe type also has a check box that allows the direction of G2/G3 arc commands to be reversed.

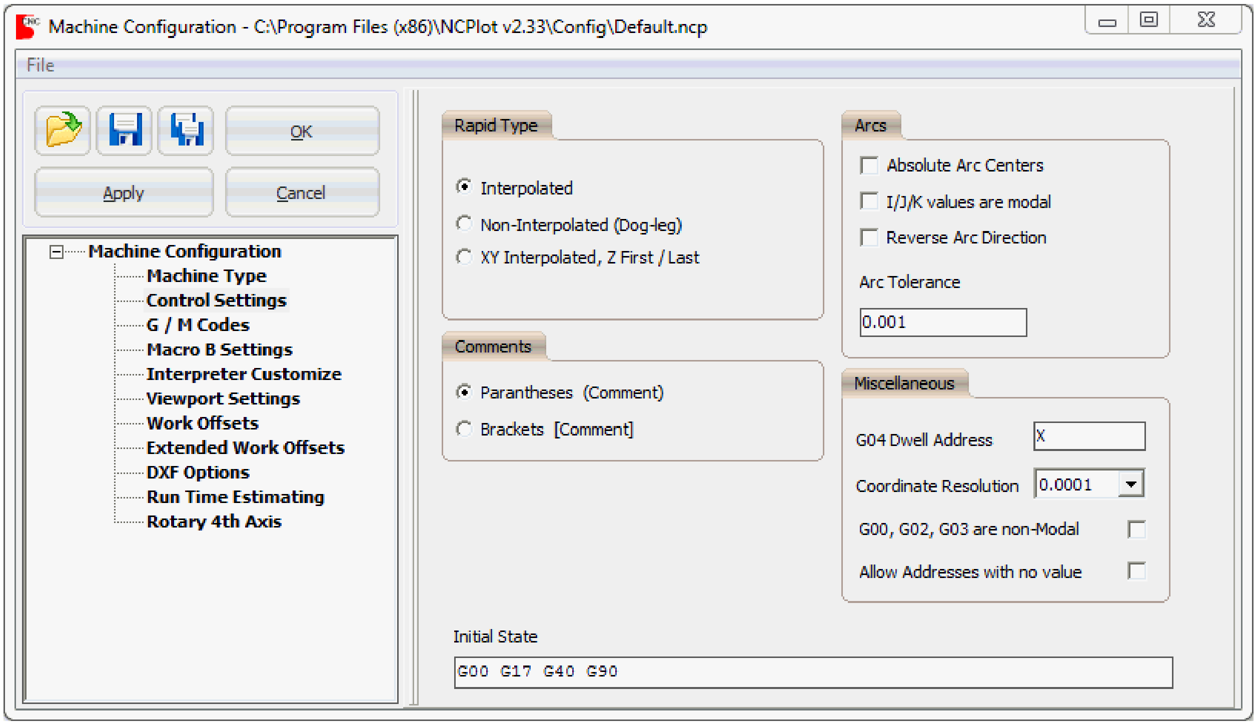
Also on this page is a setting called ***Default Program Folder***. This setting can be set to point to a folder where the G-Code programs for this particular machine configuration are stored. Say for example you have a configuration for a Makino vertical machining center. All the programs for it are stored at "C:\Jobs\MakinoVMC". Simply set the default program folder to this folder, then any time you want to open a file, the File Open dialog will open right to this folder. Since this setting is part of the machine configuration, you can specify a different folder for each configuration.

The ***Default File Types*** setting is a list of file extensions that you want to associate with your programs. This determines which file types are listed whenever you browse for a file to open or save. Wildcard characters may be used as part of the extensions.

The ***Default Script Folder*** setting allows you to specify a folder location where you store scripts associated with the configuration. When clicking the scripts panel on the status bar or the scripts toolbar button, the script list is populated with scripts only from the specified folder. If no folder is specified the default script folder at "\Program Files\NCPlot v2.20\Scripts" is used.

For lathe configurations you may choose between G-Code ***Format A*** and G-Code ***Format B***. These two formats differ between some of the G-Codes. When you select one or the other, a list of the G-Codes and their function are displayed. Select the format that most closely matches your control.

Control Settings



Machine Configuration - C:\Program Files (x86)\NCPlot v2.33\Config\Default.ncp

Rapid Type
- ● Interpolated
- ○ Non-Interpolated (Dog-leg)
- ○ XY Interpolated, Z First / Last

Comments
- ● Parantheses (Comment)
- ○ Brackets [Comment]

Arcs
- □ Absolute Arc Centers
- □ I/J/K values are modal
- □ Reverse Arc Direction

Arc Tolerance: 0.001

Miscellaneous

G04 Dwell Address: X
Coordinate Resolution: 0.0001
G00, G02, G03 are non-Modal □
Allow Addresses with no value □

Initial State: G00 G17 G40 G90

This page contains some of the most important settings for determining how your G-Code programs are interpreted. First off is the ***Rapid Type*** setting. This setting should be set to match how your machine responds to a multiple axis simultaneous rapid move. Select ***Interpolated*** if all your machine axes arrive at their endpoints at the same time. If the axes reach the endpoints one at a time, this would be ***Non-Interpolated***, sometimes called "Dog-Leg". Some controls use a third method, which is generally safer than the other two. This method is called ***Z First/Last*** and will always move the Z axis by itself, either before or after the X & Y axes depending on which direction the Z is going.

The ***Comments*** setting allows you to change the characters that NCPlot recognizes as comments. Fanuc and compatible controls will normally use the parentheses “( )” for comments, but the square brackets “[ ]” may also be used.

The arc settings determine how G02 and G03 arc commands are interpreted. If your control uses absolute arc centers, then check the setting ***Absolute Arc Centers***. When checked, the I, J and K values in a G02 or G03 command represent the location of the center of the arc in the current work coordinates. When unchecked, the I, J and K values represent the distance from the start point of the arc to the center point of the arc.

Your control may treat arc center locations as modal. If this is the case, the control remembers the last center point you programmed and you don't have to include an I, J or K value in every arc command. If you have a control that behaves this way, check the setting ***I/J/K values are modal***.

The ***Reverse Arc Direction*** setting may be used if your machine is not configured in the standard Cartesian axis configuration. This setting will reverse the drawing direction of all G02 and G03 arc commands.

When commanding an arc using IJK arc center designation, it's not uncommon for there to be a small difference between the arc's start radius and end radius. That is, the difference between the distances from the start point to the center and the distance from the end point to the center. Most controls will handle this without a problem up until the difference reaches a certain amount. Whether this amount is fixed in the control, or is parameter settable, you can enter this amount into the ***Arc Tolerance*** setting. When NCPlot encounters an arc where the start and end radius is different by more than this amount, an error message will be displayed.

The ***G04 Dwell Address*** setting allows you to define which letter address your control uses as the dwell time. Common settings are X, P, F and T.

The ***Coordinate Resolution*** setting determines how many decimal places to assume when a command value is given without a decimal point. For example, if you have a program that has commands like "Z-152500", then you would want to set the coordinate resolution to "0.0001" so that this would be properly interpreted as "Z-15.2500". Here are some more examples:

Command value	Coordinate Resolution	Interpreted value
X25	0.001	X0.025
X1	1.0	X1.0
Y1250	0.0001	Y0.1250
Y1.250	n/a	Y1.25

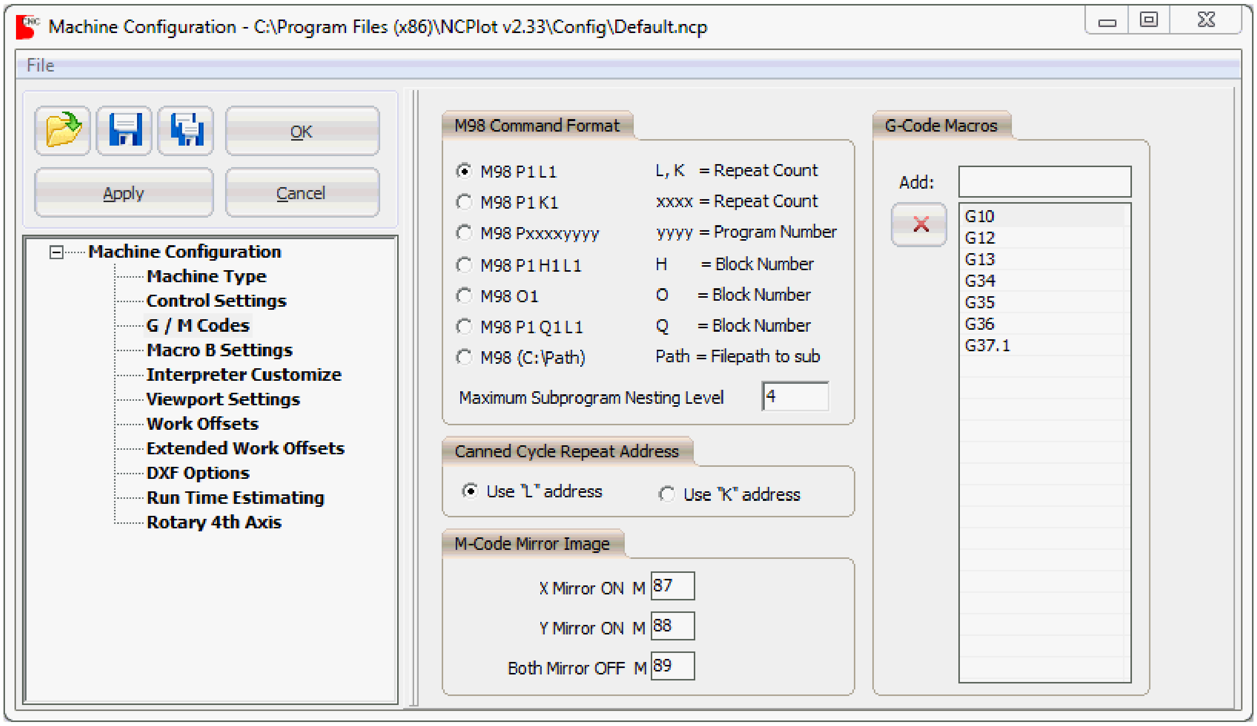
Since a decimal point was specified in the last value, the resolution setting is disregarded.

The setting ***G00, G02, G03 are non-Modal*** causes NCPlot to revert back to G01 after each block. This means that a G00, G02 or G03 command must be issued for every block that requires one of these commands.

Some controls do not require all addresses to include a value. In this case, a command such as “G00G91G28XYZ” would be interpreted as “G00 G91 G28 X0 Y0 Z0”. However, this reduces the error checking ability of NCPlot, so an option called ***Allow Addresses with no value*** can be used to set it the way you like.

When NCPlot begins to backplot a program, it starts from a fixed G-Code state. That is, certain G-Codes are active by default such as G00, G90, G54 etc. While this is acceptable for most controls, you may have a machine that defaults to some other active state, like G91. The ***Initial State*** setting is used to define the default state of your control. For example, if your control defaults to G91 you simple add "G91" to the Initial state setting.

G/M Codes



M98 Command Format

● M98 P1 L1	L, K	= Repeat Count
○ M98 P1 K1	xxxx	= Repeat Count
○ M98 Pxxxxyyyy	yyyy	= Program Number
○ M98 P1 H1 L1	H	= Block Number
○ M98 O1	O	= Block Number
○ M98 P1 Q1 L1	Q	= Block Number
○ M98 (C:\Path)	Path	= Filepath to sub

Maximum Subprogram Nesting Level: 4

Canned Cycle Repeat Address

● Use "L" address ○ Use "K" address

M-Code Mirror Image

X Mirror ON M 87
Y Mirror ON M 88
Both Mirror OFF M 89

G-Code Macros

Add:

G10
G12
G13
G34
G35
G36
G37.1

If you plan to backplot programs that use M98 for subprograms, then it's very important that you set the M98 command format to match your control. There are six different settings, so if you're not sure which one to use, you should consult your control's programming manual. For details about this setting, see the topic Plotting Subprograms.

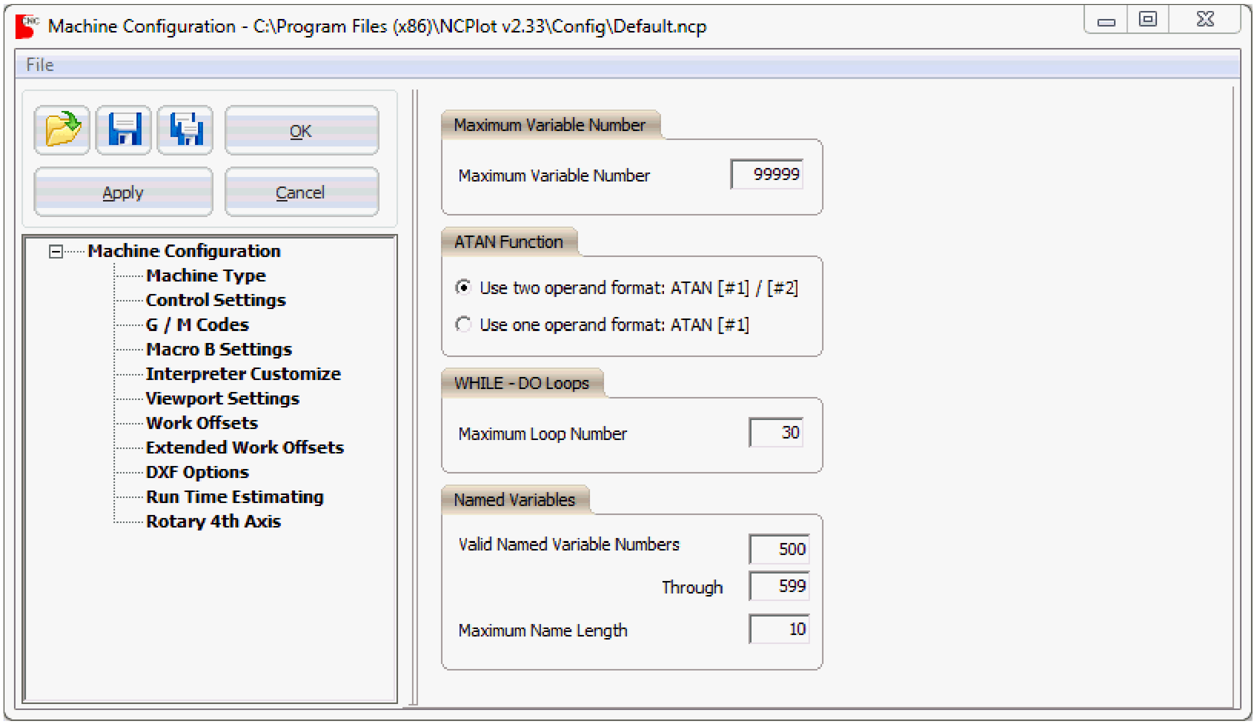
The maximum subprogram nesting level determines how many subprograms deep your programs are allowed to go before an error message is generated.

The canned cycle repeat address allows you to set the address used for specifying the repeat count, either "L" or "K" can be selected.

If your control supports M-Code activated mirror image, then use this page to set the M-Codes that are used to activate this function.

The ***G-Code Macros*** setting is a list of G-Codes that NCPlot will call as subprograms when they are encountered in a program. When encountered, all other address values are written to local variables and a specially named subprogram is loaded. The name of the subprogram that is loaded is in the format "Gxxx.PRG", where "xxx" is the G-Code value times 10. For example, if you have G12 in the G-Code macro list and NCPlot encounters the block "G12 X0 Y0 I0.5", a subprogram named "G120.PRG" must be in the configuration folder. The values for X, Y and I are saved to local variables and can be used by the subprogram to simulate the motion for a G12 command. This method allows you to simulate G-Codes that are not handled internally by NCPlot.

Macro B Settings



Maximum Variable Number

Maximum Variable Number: 99999

ATAN Function

● Use two operand format: ATAN [#1] / [#2]

○ Use one operand format: ATAN [#1]

WHILE - DO Loops

Maximum Loop Number: 30

Named Variables

Valid Named Variable Numbers: 500 Through 599

Maximum Name Length: 10

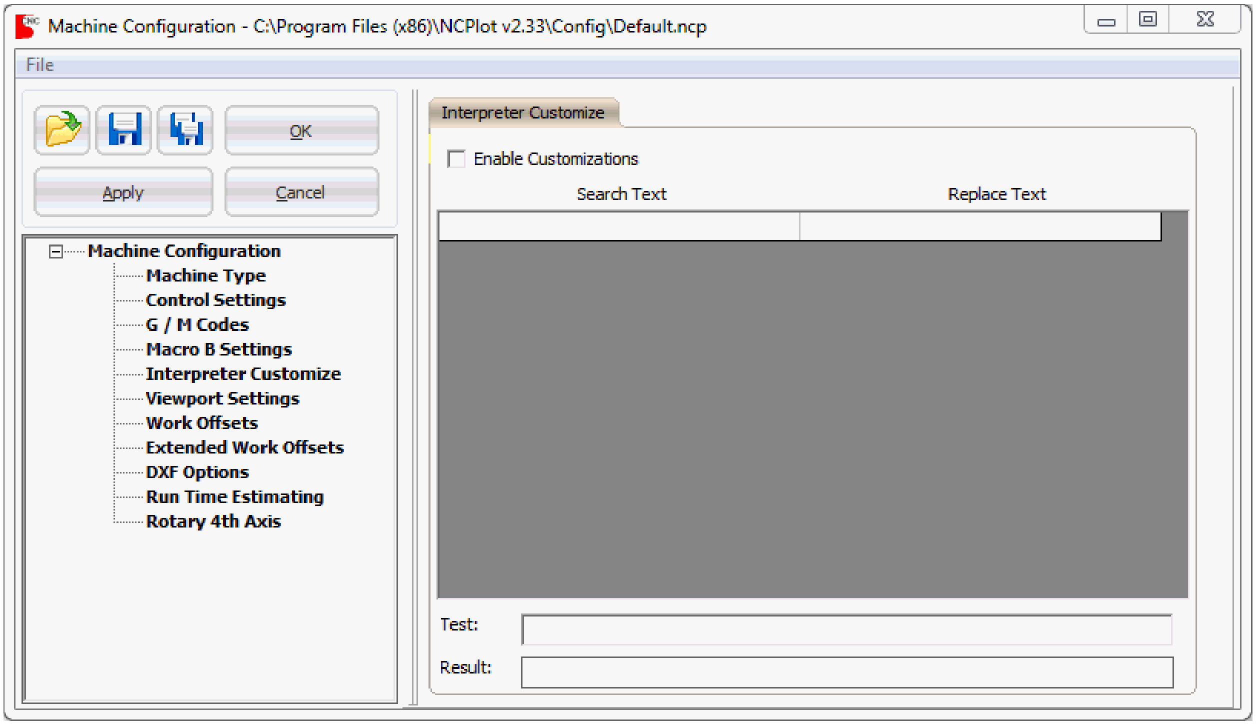
The ***Maximum Variable Number*** setting determines the highest allowed variable number. If your macro program tries to assign a value to a variable number that is higher than this setting then you will get an error message.

NCPlot supports two different formats of the macro ATAN function. This setting determines the format that is expected when an ATAN function is encountered in the program. In general, Fanuc controls expect the two operand format, while Mitsubishi controls expect the single operand format. For others, check your control documentation to determine the correct setting.

The ***Maximum Loop Number*** setting determines the maximum number that can be used for a DO - END loop. For many controls this setting should be "3", the maximum allowed setting is "30".

If your control supports named variables then set the valid variable number range and maximum length for a variable name. The variable number range determines which variables are able to be assigned a name, so attempting to set a name to a variable outside of this range will result in an error message.

Interpreter Customize



Machine Configuration - C:\Program Files (x86)\NCPlot v2.33\Config\Default.ncp

□ Enable Customizations

Search Text	Replace Text

Test:

Result:

The interpreter customize page allows greater flexibility in setting up NCPlot to backplot programs that use non-standard program or G-Code formats. It does this by finding and replacing text in the program blocks before they reach the interpreter. This process does not change the program being edited, it only changes how the program is read by the backplotter.

When the "Enable Customizations" box is checked the interpreter will find and replace each item in the list for each block in the program as it is read. As an example suppose you want the interpreter to read G70 & G71 as the inch / metric commands G20 & G21 that are recognized by NCPlot:

Search Text	Replace Text
G70	G20
G71	G21

In this example any instance of "G70" in the program will be read as "G20" and any instance of "G71" will be read as "G21".

If your control uses keywords that are not recognized by NCPlot you can use this feature to ignore them by leaving the replace field blank:

Search Text	Replace Text
WORKSHIFTS	

G200	
END:	

By replacing keywords with subprogram call commands you can handle them with an external macro.

Search Text	Replace Text
WORKSHIFTS	M98P9000
SETMS	M98P9001

This same method can also be used to create M code macros for simulating things like tool change motion or even an external 4th axis indexer.

Search Text	Replace Text
M70	M98P9170
M06	M98P9106

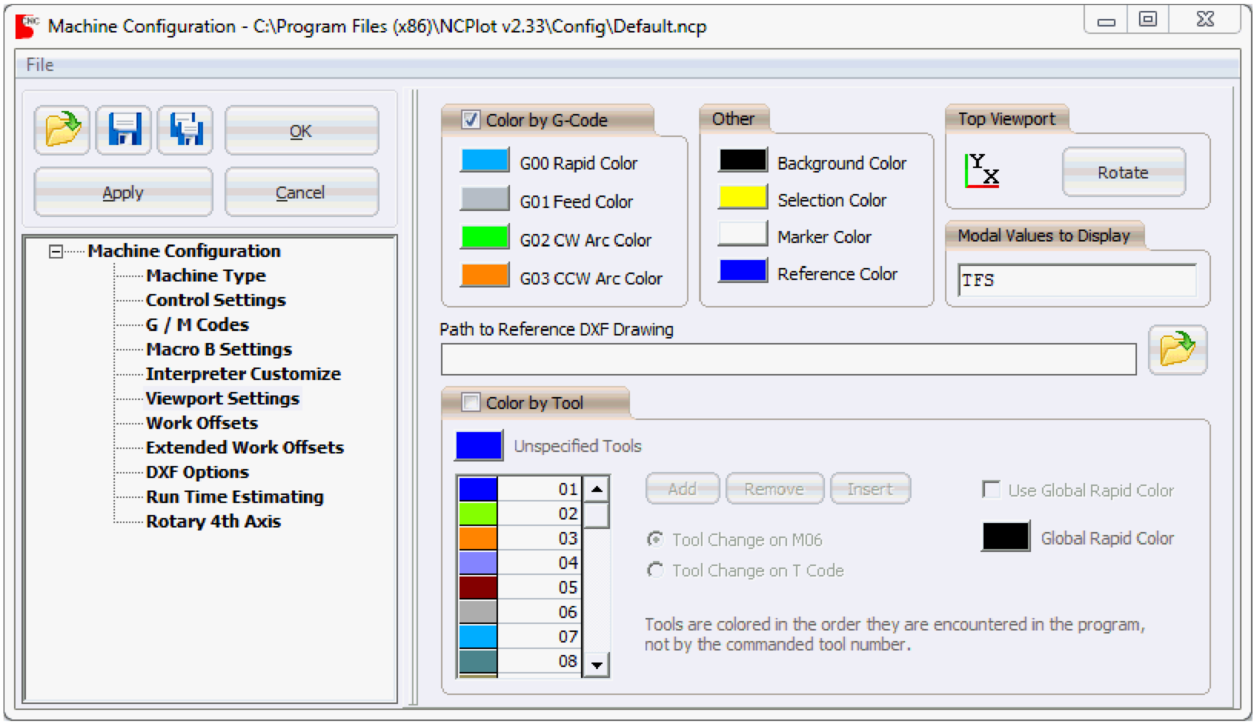
You can even redefine the standard G-Codes that are handled by NCPlot:

Search Text	Replace Text
G15	G65P9015
G16	G65P9016

Note that there will not be any spaces in the block at the time that the customizations are applied to it. Likewise, the replace text should not contain spaces.

The ***Test*** text field allows you to enter text so that you can see the effect that the customization list has on it. The ***Result*** field is what would be passed to the backplotter.

Viewport Settings



Machine Configuration - C:\Program Files (x86)\NCPlot v2.33\Config\Default.ncp

- Machine Configuration
  - Machine Type
  - Control Settings
  - G / M Codes
  - Macro B Settings
  - Interpreter Customize
  - Viewport Settings
  - Work Offsets
  - Extended Work Offsets
  - DXF Options
  - Run Time Estimating
  - Rotary 4th Axis

☑ Color by G-Code

- G00 Rapid Color
- G01 Feed Color
- G02 CW Arc Color
- G03 CCW Arc Color

Other

- Background Color
- Selection Color
- Marker Color
- Reference Color

Top Viewport

Modal Values to Display: TFS

Path to Reference DXF Drawing

□ Color by Tool

Unspecified Tools

01, 02, 03, 04, 05, 06, 07, 08

□ Use Global Rapid Color

● Tool Change on M06

○ Tool Change on T Code

Global Rapid Color

Tools are colored in the order they are encountered in the program, not by the commanded tool number.

This page contains settings that define the colors used to draw the backplot. You first must decide if you want to color by G-Code, or color by tool. To select one, check the box next to the header describing the method you want to use. When ***Color by G-Code*** is selected, the entities on the graphics viewport be will colored according to the type of motion it represents. There are 4 basic types of motion: G00 Rapid move, G01 Feed move, G02 Clockwise arc and G03 Counterclockwise arc. Each of these types of motion may be assigned a different color.

The ***Color by Tool*** option draws the backplot with different colors representing the range of motion for each tool used in the program. The ***Unspecified Tools*** color is used when the program commands motion before the first tool change or when there are more tools used in the program than have been defined. The color list contains the colors to use for each tool. The first color in the list is used after the first tool change, the second color after the second tool change, etc. If there are not enough colors in the list for all of the tool changes in the program, the ***Unspecified Tools*** color will be used for any remaining tool changes. You may also specify the type of command that is considered a tool change, either the M06 command or a T-Code. The ***Use Global Rapid Color*** option allows the G00 rapid moves to be displayed as the specified color regardless of tool number.

The viewport can display a reference drawing in DXF format. When loaded, this drawing is a persistent part of the viewport and is useful for displaying things like machine travels, part fixtures, interference areas, reference grids, etc. To display a drawing on the viewport, set the ***Path to Reference DXF Drawing*** setting to the path of the drawing you want to display.

In addition to the entity colors, you can also specify the background color of the graphics viewport, the color of entities that are selected, the color of the marker arrow and the color of the reference drawing.

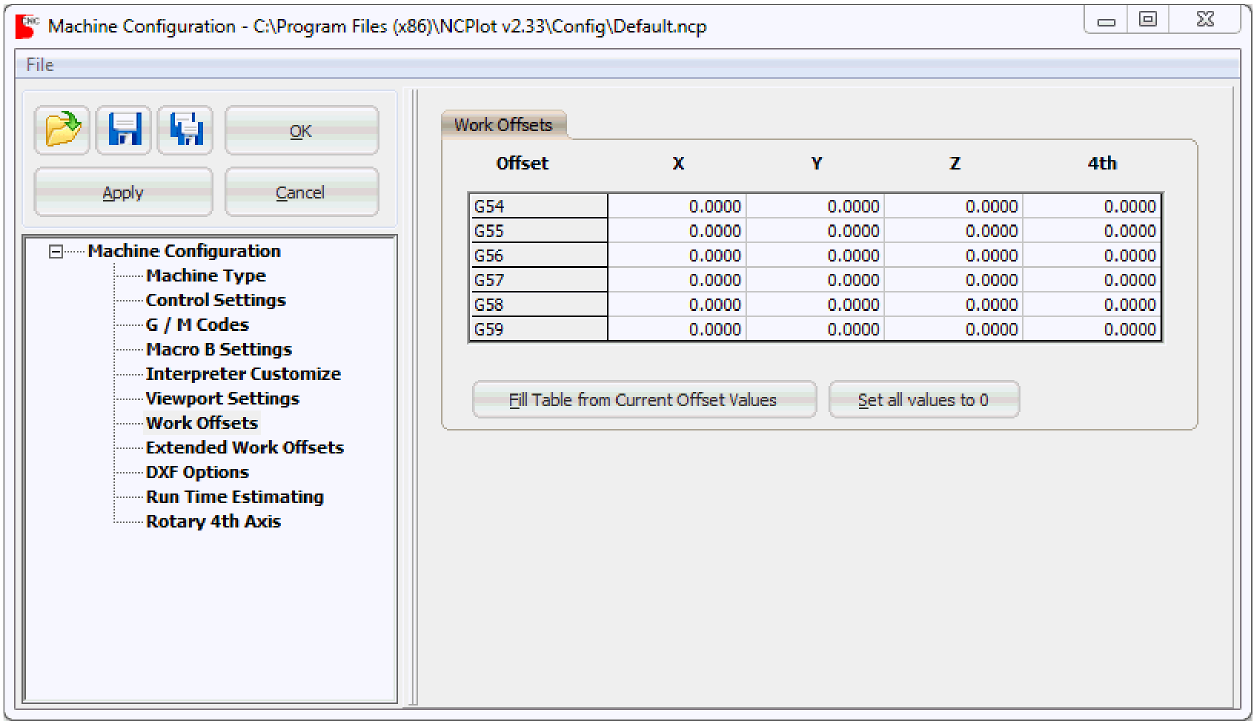
The ***Top Viewport*** rotation setting allows you to re-orient the graphics display to match the way the part appears from the operator side of the machine. This is simply a convenience setting that only affects the graphics view.

The ***Modal Values to Display*** setting determines which address values are displayed along with the location as part of the viewport marker. These values are displayed just below the endpoint location. Active G-Codes may be displayed here and are entered as "G" followed by a numeric value which indicates the group number to display. More than one "G" code may be entered. The group numbers are different between the machine types, these links will take you to a listing of the G-Codes and their group numbers: Mill, Lathe Format A or Lathe Format B.

Here are some examples:

TFS	T01 F12.4 S1600
TG12G2G8	T01 G54 G17 G43
G8HG7DG12S	G43 H01 G40 D00 S1600
G2	G17 (Displays the active G-Code from group 2)
G12	G54 (Displays the active G-Code from group 12)
G8H	G43 H01 (Displays the group 8 G-Code plus the active H value)

Work Offsets



Machine Configuration - C:\Program Files (x86)\NCPlot v2.33\Config\Default.ncp

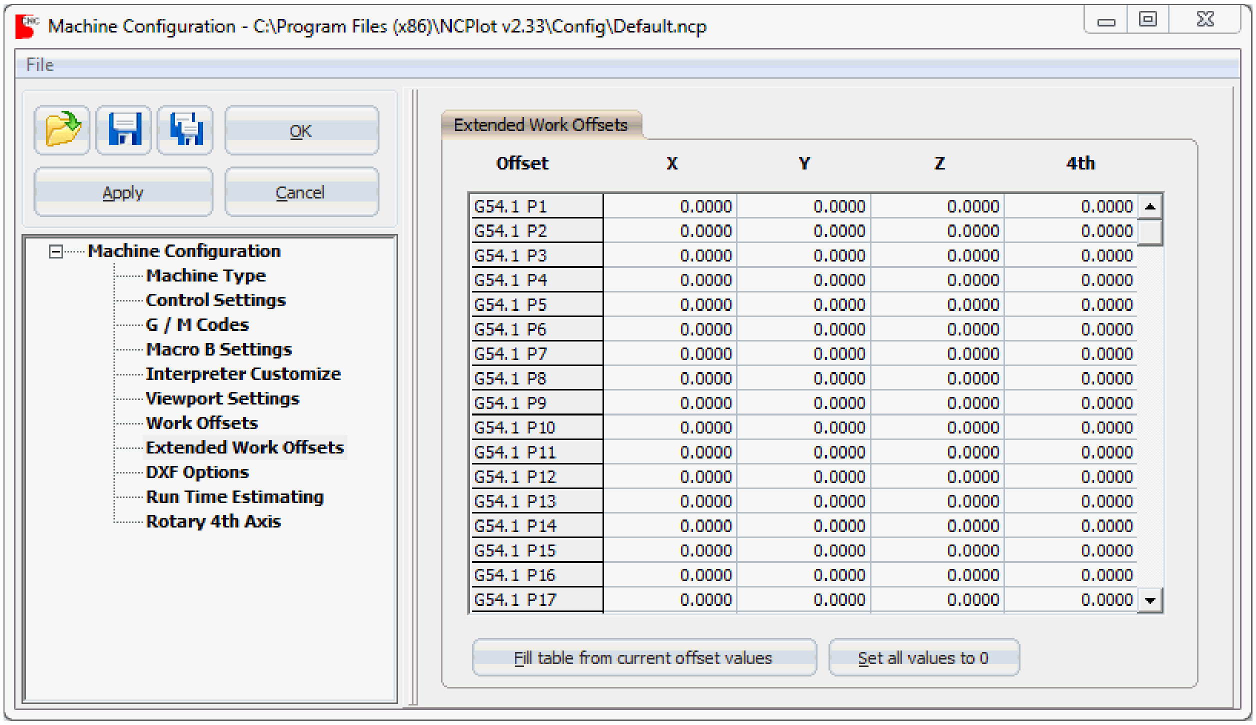
Offset	X	Y	Z	4th
G54	0.0000	0.0000	0.0000	0.0000
G55	0.0000	0.0000	0.0000	0.0000
G56	0.0000	0.0000	0.0000	0.0000
G57	0.0000	0.0000	0.0000	0.0000
G58	0.0000	0.0000	0.0000	0.0000
G59	0.0000	0.0000	0.0000	0.0000

Just like your machine can accommodate multiple work offset coordinates, NCPlot can also be configured to recognize multiple work locations. This gives a backplot that accurately represents a multiple fixture setup.

The **Fill table from current offset values** button will copy the current work offset system variables to the work offset table. This is useful for saving work offset values that have been set by your programs.

The **Set all values to 0** button will clear all work offset values to 0.

Extended Work Offsets



Machine Configuration - C:\Program Files (x86)\NCPlot v2.33\Config\Default.ncp

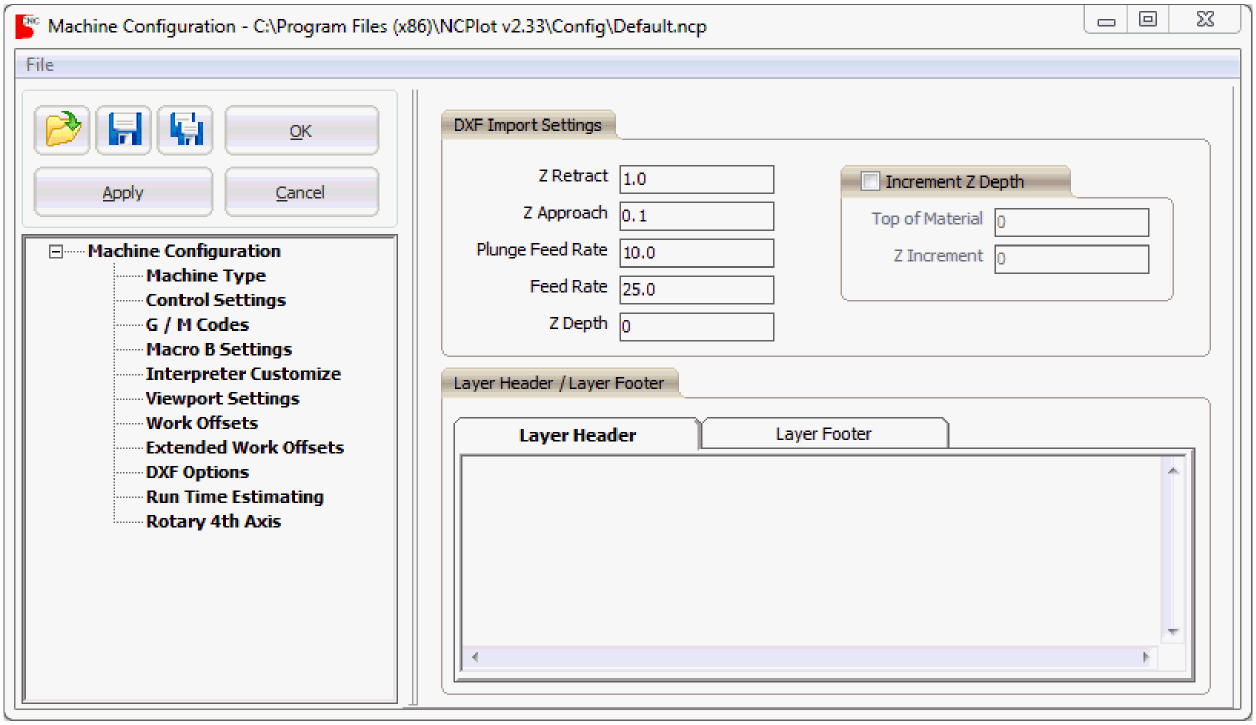
Offset	X	Y	Z	4th
G54.1 P1	0.0000	0.0000	0.0000	0.0000
G54.1 P2	0.0000	0.0000	0.0000	0.0000
G54.1 P3	0.0000	0.0000	0.0000	0.0000
G54.1 P4	0.0000	0.0000	0.0000	0.0000
G54.1 P5	0.0000	0.0000	0.0000	0.0000
G54.1 P6	0.0000	0.0000	0.0000	0.0000
G54.1 P7	0.0000	0.0000	0.0000	0.0000
G54.1 P8	0.0000	0.0000	0.0000	0.0000
G54.1 P9	0.0000	0.0000	0.0000	0.0000
G54.1 P10	0.0000	0.0000	0.0000	0.0000
G54.1 P11	0.0000	0.0000	0.0000	0.0000
G54.1 P12	0.0000	0.0000	0.0000	0.0000
G54.1 P13	0.0000	0.0000	0.0000	0.0000
G54.1 P14	0.0000	0.0000	0.0000	0.0000
G54.1 P15	0.0000	0.0000	0.0000	0.0000
G54.1 P16	0.0000	0.0000	0.0000	0.0000
G54.1 P17	0.0000	0.0000	0.0000	0.0000

In addition to the standard work offsets G54 through G59, NCPlot also supports the use of extended work offsets. This configuration page allows the setting of your extended work offsets.

The **Fill table from current offset values** button will copy the current extended work offset system variables to the extended work offset table. This is useful for saving extended work offset values that have been set by your programs.

The **Set all values to 0** button will clear all extended work offset values to 0.

DXF Options



Machine Configuration - C:\Program Files (x86)\NCPlot v2.33\Config\Default.ncp

DXF Import Settings

Z Retract	1.0
Z Approach	0.1
Plunge Feed Rate	10.0
Feed Rate	25.0
Z Depth	0

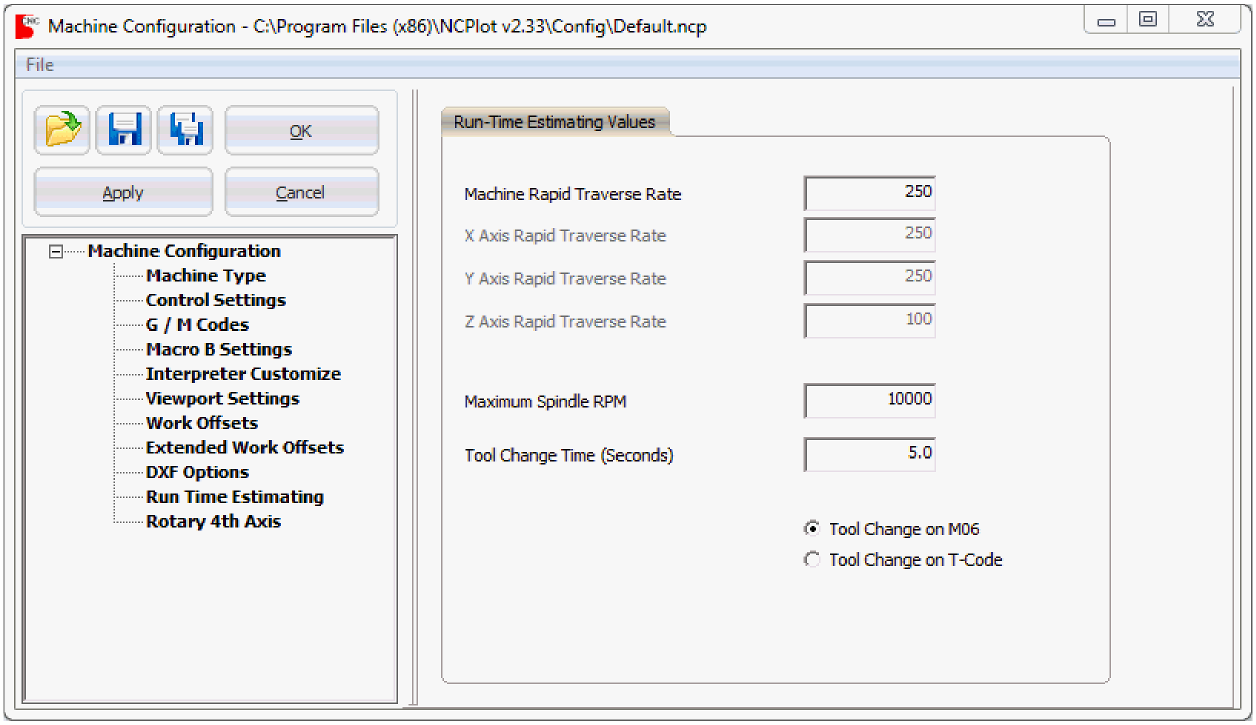
☐ Increment Z Depth

Top of Material	0
Z Increment	0

Layer Header / Layer Footer

Use this page to enter default DXF drawing conversion settings. When a DXF file is loaded, any layer that is not in your saved layer list will be given these settings. For more information see the section Converting DXF Drawing Files.

Run Time Estimating



Machine Configuration - C:\Program Files (x86)\NCPlot v2.33\Config\Default.ncp

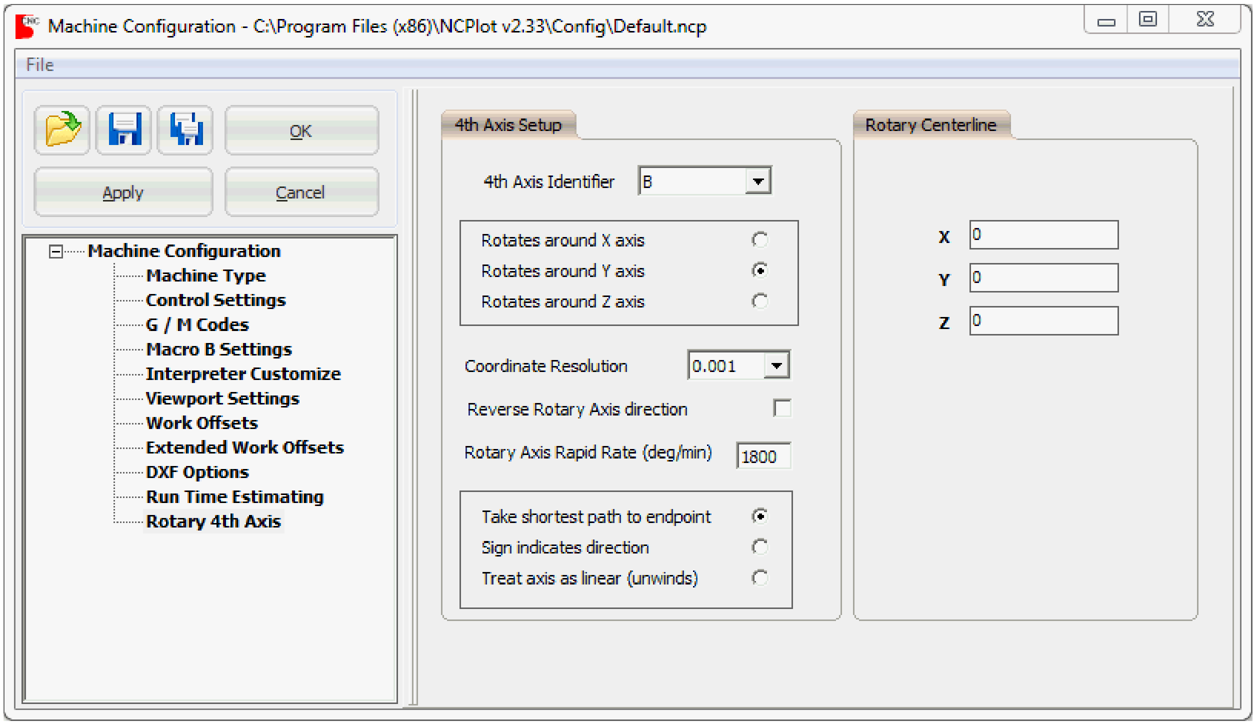
Machine Rapid Traverse Rate 250
X Axis Rapid Traverse Rate 250
Y Axis Rapid Traverse Rate 250
Z Axis Rapid Traverse Rate 100
Maximum Spindle RPM 10000
Tool Change Time (Seconds) 5.0
● Tool Change on M06
○ Tool Change on T-Code

The settings on this page are used for calculating the estimated machining time of your programs. The ***Machine Rapid Traverse Rate*** and ***Tool Change Time*** settings are used for all machine types, while the ***Maximum Spindle RPM*** setting is used only during lathe CSS calculations.

If you have the ***Non-Interpolated (Dog-leg)*** rapid type selected on the ***Control Settings*** page then you can set the individual axis rapid rates here. Setting these values to your actual machine rapid rates will provide a more accurate plot of the rapid motion in your program.

Also on this page is a selection for specifying the type of command that is considered a tool change, either the M06 command or a T-Code. This is the same setting as on the ***Viewport Settings*** page, duplicated here to allow changing it when ***Color By Tool*** is not enabled.

Rotary 4th Axis



4th Axis Setup

4th Axis Identifier: B

○ Rotates around X axis
● Rotates around Y axis
○ Rotates around Z axis

Coordinate Resolution: 0.001

□ Reverse Rotary Axis direction

Rotary Axis Rapid Rate (deg/min): 1800

● Take shortest path to endpoint
○ Sign indicates direction
○ Treat axis as linear (unwinds)

Rotary Centerline

X: 0
Y: 0
Z: 0

If your machine has a rotary 4th axis, use this page to define the settings for it. First, set the ***4th Axis Identifier*** to specify the letter address that commands the 4th axis. The most common settings are an "A" or "B" axis. Next, set the orientation of the rotary axis by specifying which axis it rotates around. By definition an "A" axis rotates around the "X" axis, a "B" axis rotates around the "Y" axis and a ”C” axis rotates around the ”Z” axis. You must also set the ***Coordinate Resolution*** setting for the 4th axis command values. This works the same way as the setting on the ***Control Settings*** page.

The ***Reverse Rotary Axis direction*** setting allows you change the positive rotation direction of the rotary axis.

One of three interpolation methods may be selected, select the setting that matches the way your rotary axis behaves:

1. **Take shortest path to endpoint** - the rotary axis will move in the direction that results in less than 180 degrees of motion.
2. **Sign indicates direction** - the rotary axis will move to the designated endpoint in the direction indicated by the sign of the endpoint value.
3. **Treat axis as linear (unwinds)** - the rotary axis endpoints are handled as though the axis was linear. If the axis rotates past 360 degrees it must move back the same amount to get back to zero. This is usually referred to as "unwinding" the axis.

The ***Rotary Centerline*** settings can be used to specify where on your machine the rotary axis is located. This tells NCPlot where the center of rotation is located on the machine.

Plotting Subprograms

NCPlot provides you with the ability to backplot G-Code programs that make use of sub programming. This includes support for M98, G65 and M97 subprogram call codes. But, like the machine configuration, there are some settings that need to be configured to match the way your particular control works.

Select a subprogram call format

Of the three subprogram commands, M98 is the only one that varies between different controls. So, the ***Machine Configuration*** dialog contains an option that let's you tell NCPlot which command format your control uses. This is on the **G/M Codes** page and there are six possible settings:

M98 P1 L1

This is the most common setting and works for most Fanuc controls. The "P" address is the program number to call as a subprogram, and the "L" address is the number of times to repeat the sub.

M98 P1 K1

Some older Fanuc controls use the "K" address as the repeat count instead of the "L" address.

M98 P010002

Some Fanuc controls combine the subprogram number and the repeat count into a single 6 digit number. The first two digits are the repeat count and the last four digits are the subprogram number. So, in this example the repeat count is 01 and the program number is 0002.

M98 P1 H1 L1

Mitsubishi controls add the use of the "H" address which represents the target block number in the subprogram being called. So, besides the program number and the repeat count, you can also specify a starting block number for the subprogram.

M98 O1

For some other types of controls, M98 is a local subprogram call that uses the "O" address to specify a target block number. This type of subprogram call cannot use other programs as subs, the subprogram must be within the current program.

M98 P1 Q1 L1

This format is a variation in which the "Q" address represents the target block number in the subprogram being called.

The other two subprogram call commands use fixed formats, which are:

M97 P1

This code is used by the HAAS controls and is a local subprogram call that uses the "P" address to specify the target block number. This type of subprogram call cannot use other programs as subs, the subprogram must be within the current program.

G65 P1

This code is commonly used for macro programming because any address values that appear after the G65 are copied to local variables before the subprogram is called. The exception is the "P" address, which represents the program number being called. This makes it very useful for creating custom cycles that work much like the control's built-in cycles.

Where to put your subprograms

Once you've got NCPlot set up to read the subprogram calls correctly, you just need to be able to tell it where the subprograms are on your computer. On the CNC control, this is not an issue because the control knows where all the programs are, but on your PC this may not be so simple. If you only use the local subprogram call commands M97 or M98 O, no setup is required because the subprogram must be within the main program.

The simplest way to handle subprograms is to include them in the file you are editing. However, this is not always practical and so NCPlot will search for subprograms in several locations. The order in which the search takes place is:

- The currently loaded file is searched first.

  Programs in the edit file are identified by a line that begins with the colon character “:” or the letter “O” followed by a number and optionally a comment. For example:
  O1234(MILLING SUB)

- The subprogram association list is searched next.

  The subprogram association list is on the subprogram setup dialog and can be opened by clicking the menu ***Setup / Subprograms***. The subprogram association list allows you to assign a specific file to a specific program number. To create a new association, click the "Add" button on the subprogram setup window. You will be prompted for the program number to assign. Enter the program number and you can then browse for the file that contains the G-Code program for the specified program number. There is no limit to the number of associations you can assign.

- The subprogram default folder is searched next.

  This is also a setting on the subprogram setup window and defines a folder to be searched for subprograms. Subprogram files in this folder must be named in a specific format in order for NCPlot to locate them. The file names should be the program number optionally preceded by the letter “O”. File extensions are ignored. These are examples of valid file names:

  O1234.NC
  1234.NC
  O12345678.TXT

- The folder of the active edit program file is searched last.

  Similar to the default folder, the folder where the active edit file is located will be searched last. The filename requirements are the same as for the default folder.

The subprogram setup window also contains settings that determine what action will be taken if the requested subprogram cannot be found. The possible actions can be set separately for M98 and G65 commands and are:

- **Error if sub not found** - if the commanded subprogram could not be located, NCPlot stops plotting and displays an error message.
- **Warn if sub not found** - if the commanded subprogram could not be located, NCPlot pauses to display a message then continues.
- **Ignore if sub not found** - if the commanded subprogram is found it will be plotted, otherwise NCPlot will skip it.
- **Ignore all** - all subprogram commands will be ignored.

Canned Cycles

The backplotting of canned cycles in NCPlot is not hard coded. These cycles are external macro programs that may be customized to match your particular machine. The canned cycle macros are located at:

\Program Files\NCPlot\Config\*config name*

The Config folder contains the machine configuration files along with a corresponding sub folder for each. These sub folders contain the canned cycle macro programs. Every machine configuration uses it's own set of canned cycle macros, which means that you can have different canned cycles for each machine configuration.

The G-Codes that are handled as external macros by default are:

G70 – G79, G81 – G89	Standard canned cycles

G12	Clockwise circle cutting
G13	Counterclockwise circle cutting

The G34 through G37.1 cycles are included in the default mill configurations.

G34	Bolt circle cycle
G35	Holes on line at angle cycle
G36	Holes on arc cycle
G37.1	Grid pattern cycle

The G70 through G72 bolt pattern cycles are included in the HAAS mill configuration.

G70	Bolt circle cycle
G71	Holes on arc cycle
G72	Holes on line at angle cycle

Additional cycles may be added to simulate G-Codes not supported by NCPlot. Open the ***Machine Configuration*** dialog to the ***G/M Codes*** page and enter additional G-Codes in the ***G-Code Macros*** list. The active configuration folder must contain a corresponding program file to handle the simulation of the additional codes.

The macro program file name is the corresponding G-Code value times 10. For example, the file name for G81 is G810.PRG. This is to accommodate decimal G-Codes such as G37.1. You can add cycles for G-Codes that fall within these ranges by simply adding an appropriately named G-Code program file to the desired configuration folder.

Menus

File Menu

New

Opens a new blank document.

Open File

This selection allows you to browse for an existing file to be loaded into the edit window. If any unsaved changes have been made to the current program, you are prompted to save it before loading a new file. The file browse window contains a drop down list that allows you to select the types of files to be displayed, either plain text files of various extensions, .DXF drawing files or all files. If a .DXF drawing file is selected, the .DXF conversion options window will open instead of the file being loaded into the edit window.

After a file is loaded, it will automatically be plotted and zoomed to the size of the viewport. Either plain text or rich text (RTF) files may be loaded, the format will be determined automatically.

Open Recent File

Displays a list of up to 20 of the most recently opened files.

Insert File

This selection allows you to browse for an existing file to be inserted into the edit window. The selected file will be inserted at the current cursor position. If any unsaved changes have been made to the current program, you are prompted to save it before loading a new file. The file browse window contains a drop down list that allows you to select the types of files to be displayed, either plain text files of various extensions, .DXF drawing files or all files. If a .DXF drawing file is selected, the .DXF conversion options window will open instead of the file being loaded into the edit window. Either plain text or rich text (RTF) files may be inserted, the format will be determined automatically.

Merge Files

Allows you to select multiple files to merge into one document.

Close

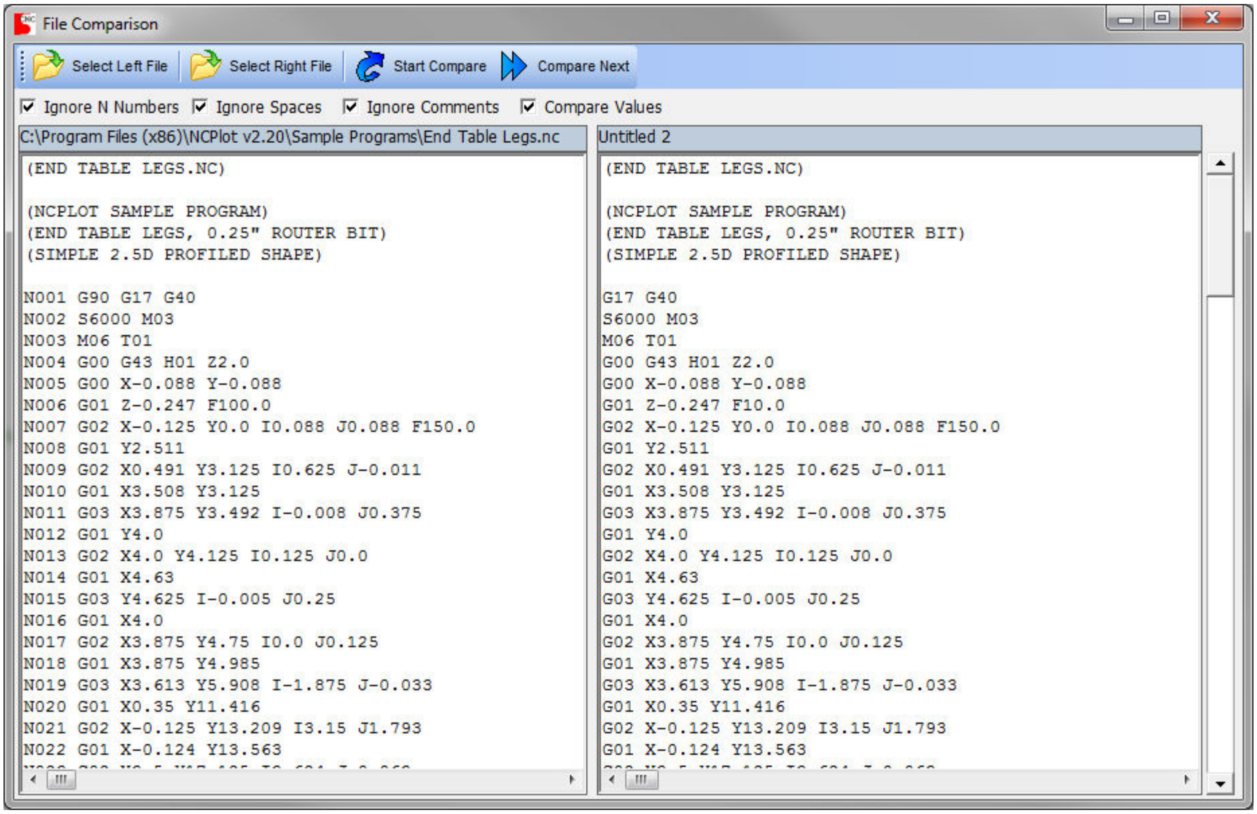
This selection clears the current program and the viewport. If any unsaved changes have been made to the current program, you are prompted to save it before clearing the program.

Close All

Closes all open files.

Compare Files

Opens the file comparison tool.



☑ Ignore N Numbers ☑ Ignore Spaces ☑ Ignore Comments ☑ Compare Values

C:\Program Files (x86)\NCPlot v2.20\Sample Programs\End Table Legs.nc

```
(END TABLE LEGS.NC)

(NCPLOT SAMPLE PROGRAM)
(END TABLE LEGS, 0.25" ROUTER BIT)
(SIMPLE 2.5D PROFILED SHAPE)

N001 G90 G17 G40
N002 S6000 M03
N003 M06 T01
N004 G00 G43 H01 Z2.0
N005 G00 X-0.088 Y-0.088
N006 G01 Z-0.247 F100.0
N007 G02 X-0.125 Y0.0 I0.088 J0.088 F150.0
N008 G01 Y2.511
N009 G02 X0.491 Y3.125 I0.625 J-0.011
N010 G01 X3.508 Y3.125
N011 G03 X3.875 Y3.492 I-0.008 J0.375
N012 G01 Y4.0
N013 G02 X4.0 Y4.125 I0.125 J0.0
N014 G01 X4.63
N015 G03 Y4.625 I-0.005 J0.25
N016 G01 X4.0
N017 G02 X3.875 Y4.75 I0.0 J0.125
N018 G01 X3.875 Y4.985
N019 G03 X3.613 Y5.908 I-1.875 J-0.033
N020 G01 X0.35 Y11.416
N021 G02 X-0.125 Y13.209 I3.15 J1.793
N022 G01 X-0.124 Y13.563
```

Untitled 2

```
(END TABLE LEGS.NC)

(NCPLOT SAMPLE PROGRAM)
(END TABLE LEGS, 0.25" ROUTER BIT)
(SIMPLE 2.5D PROFILED SHAPE)

G17 G40
S6000 M03
M06 T01
G00 G43 H01 Z2.0
G00 X-0.088 Y-0.088
G01 Z-0.247 F10.0
G02 X-0.125 Y0.0 I0.088 J0.088 F150.0
G01 Y2.511
G02 X0.491 Y3.125 I0.625 J-0.011
G01 X3.508 Y3.125
G03 X3.875 Y3.492 I-0.008 J0.375
G01 Y4.0
G02 X4.0 Y4.125 I0.125 J0.0
G01 X4.63
G03 Y4.625 I-0.005 J0.25
G01 X4.0
G02 X3.875 Y4.75 I0.0 J0.125
G01 X3.875 Y4.985
G03 X3.613 Y5.908 I-1.875 J-0.033
G01 X0.35 Y11.416
G02 X-0.125 Y13.209 I3.15 J1.793
G01 X-0.124 Y13.563
```

This tool allows comparing two files and highlighting any differences. It consists of a button bar, option check boxes and the two files to compare. The button bar functions are:

- **Select Left File** - This button pops up a list of the open documents. Clicking an item in the list will copy it to the left side document.
- **Select Right File** - This button pops up a list of the open documents. Clicking an item in the list will copy it to the right side document.
- **Start Compare** - Reloads the selected files and begins comparison. If any edits have been made to the files in the main NCPlot window then these changes are reflected in the comparison tool before comparison begins.
- **Compare Next** - Searches for the next mismatch between the two files.

The comparison allows ignoring formatting differences between the two files such as line numbering and spaces. These options are enabled by the corresponding check boxes:

- **Ignore N Numbers** - The comparison disregards any differences in line numbering.
- **Ignore Spaces** - The comparison disregards any spaces.
- **Ignore Comments** - The comparison disregards any comments in the two files.
- **Compare Values** - When checked the comparison disregards any leading or trailing zeroes in numeric values.

Import DXF File

This feature allows reading files that are in the Drawing eXchange Format. You are first prompted to select a file for importing. The selected file is then loaded and displayed and you are presented with the DXF conversion options dialog.

For help with converting DXF files to G-Code please see the topic Converting DXF Drawing Files to G-Code.

Export DXF File

This option will save the current backplot display as a DXF drawing file. The drawing file will include all axis motion except G00 rapid moves. When the machine type is configured for lathe, the exported motion is translated from the ZX plane to the XY plane. The saved DXF file will have a layer for each tool used in the program. Each layer will be named for the tool number that created it.

If any tool paths have been hidden by unchecking them on the tool list toolbar you will receive a message when attempting to export. Hidden tool paths will not be exported, you can either choose to export anyway or cancel the export operation.

Export Selected as DXF File

This tool is similar to the Export DXF File tool, except that only currently selected viewport entities will be saved.

Save File

Saves your current edits to the loaded file. If the current file is untitled, you will be prompted for a filename to save it as.

If you want to save document formatting with your file (colors, fonts, etc.), you can specify rich text format (RTF) as the file type. Note that formatted documents cannot be run by most machine controls, so this feature is primarily for documentation purposes.

Save File As

Saves your program under a new name. If you want to save document formatting with your file (colors, fonts, etc.), you can specify rich text format (RTF) as the file type. Note that formatted documents cannot be run by most machine controls, so this feature is primarily for documentation purposes.

Save As Separate

Saves all programs in the loaded file as individual files. You will be prompted for a folder to save to, and the files are saved with the program names used as the file names. For example, if you had the following file the saved files would be O100.txt, O1000.txt and O1001.txt.

```
%
O100(PROGRAM_100)
G0X0Y0Z1.
M98 P1000
M98 P1001
M2

O1000(SUB_1)
G91G1X5.F100.
```

```
M99

O1001(SUB_2)
G91G1X-5.F100.
M99
%
```

NCPlot recognizes the start of a new program when it encounters an “O” word or a colon character “:” at the beginning of a line followed by a program number. For a report of all the programs in the file, see the “Show Programs in File” tool.

Send To

This menu provides an easy way to send a file being edited in NCPlot to another application. The list of applications can be customized via the file "SendTo.ini" which can be found in the NCPlot application folder.

Example SendTo.ini file:

```
[Notepad]
notepad.exe %1
[Wordpad]
c:\Program Files\Windows NT\Accessories\wordpad.exe %1
```

The file should contain the name or description of the application enclosed in square brackets. This is what appears in the 'Send To' menu. This line should be followed by the executable application name on a separate line, including the path if necessary.

The symbols "%1" in the example above will be replaced with the active edit file path.

Show Programs in File

This tool reports a list of all programs it finds in the loaded file. Each program number is reported along with its approximate size and program comment. NCPlot recognizes the start of a program when it encounters an “O” word or the colon character “:” at the beginning of a line.

Here’s an example report:

```
   Program    Size  Description
================================
      6000    4141  LINK MAIN-PALLET A
      6001    1669  LINK MAIN-PALLET B
      8000     198  PROG FOR PART RESTART-GENERIC
      8001     172  TEST TO QUALIFY TOOL-
```

Execute Script File

Allows you to browse for and execute a script file. NCPlot provides a folder for commonly used scripts in the “Scripts” subfolder located in the application install folder. The default location for this folder is “\Program Files\NCPlot\Scripts”. Script files that are stored in this folder can be quickly accessed through the scripts toolbar button or the scripts status bar panel.

Print Program

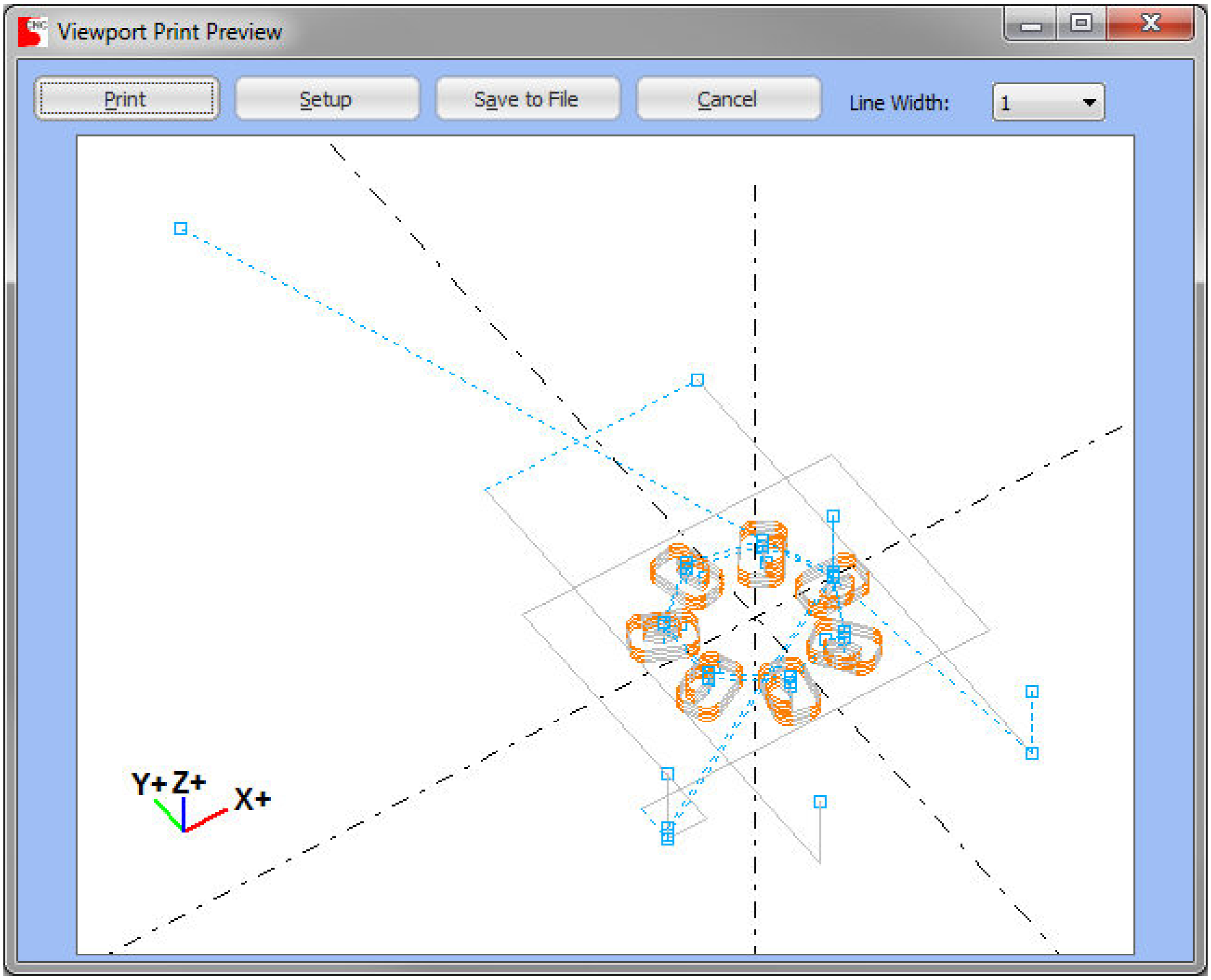
Sends the current program contents to the printer.

Print Selected Text

Sends the currently selected text to the printer.

Print Viewport

Opens the print preview window so that print settings can be configured before sending to the printer.



- **Print** - Sends the plot to the selected printer.
- **Setup** - Opens the printer setup dialog where you can select which printer to use, the desired print orientation, the number of copies and black & white or color printing.
- **Save to File** - Allows saving the print preview graphic as a Bitmap file. When clicked you will be prompted for the desired image width in pixels.
- **Cancel** - Closes the print preview window without printing.
- **Line Width** - Sets the drawing line width for printing, can be set from 1 to 20.

Clear Recent Files

Clears the list of recently opened files. The last 20 files that were opened are stored in the recent files list for easy re-opening. The recent files list is located on the toolbar next to the 'Open' icon. Clicking on the arrow opens the list, and clicking on a file in the list will open the file. If there are unsaved changes, you are prompted to save before opening the selected file. If the file no longer exists, you receive an error message and the file is removed from the list. When a file is opened from the recent files list, that file is moved to the top of list.

Reset Toolbars

This will load the default toolbar layout, clearing any changes that have been made to the layout including custom toolbars.

Exit

Exits the application. If any unsaved changes have been made to any of the loaded files you will be prompted to save them before the application closes.

Edit Menu

Undo

Undoes the last edit to the program.

Redo

Redo the last undone change.

Cut

Cuts the selected text from the program and places it on the clipboard.

Copy

Copies the selected text from the program to the clipboard.

Paste

Pastes the current clipboard contents into the program at the current cursor location.

Select From

Use this tool to mark the current cursor location in the file. Then move the cursor to the end of the area you want to select and use the “Select To” tool to select the desired region.

Select To

Selects from the marked cursor location to the current cursor location. See Select From.

Select All

Selects the entire contents of the edit window.

Find

Find text in the program. If a single word or line of text is selected, it will be entered into the ***Find What*** field of the find dialog. If multiple lines are selected then the ***Find What*** field will be blank and the find or replace operations will take place within the selected region of the file.

Find Next

Repeats the last Find operation. If the end of the program is encountered, you are asked if you want to repeat the search from the beginning.

Replace

Replaces text in the program. The selected program text is replaced by the text in the ***Replace With*** field.

Jump To Top

Moves the cursor to the beginning of the active file.

Jump To Line Number

Prompts for, then moves the cursor to the entered file line number.

Jump To Bottom

Moves the cursor to the end of the active file.

Highlight Selected

This tool is used to locate the selected portion of your program on the viewport. To use this tool, select the portion of the program you want to locate and then click the Highlight Selected menu item. The entities on the viewport that correspond to the selected program will then be highlighted.

Highlight and Zoom to Selected

This tool is used to locate the selected portion of your program on the viewport. To use this tool, select the portion of the program you want to locate and then click the Highlight and Zoom to Selected menu item. The entities on the viewport that correspond to the selected program will be highlighted and the viewport will zoom to fit these entities.

Font

Opens the font dialog. The selected font and color settings may be applied to the entire program or just the selected text. If any part of the program text is selected, only the selected text will be affected. Otherwise, the settings will be applied to the entire program.

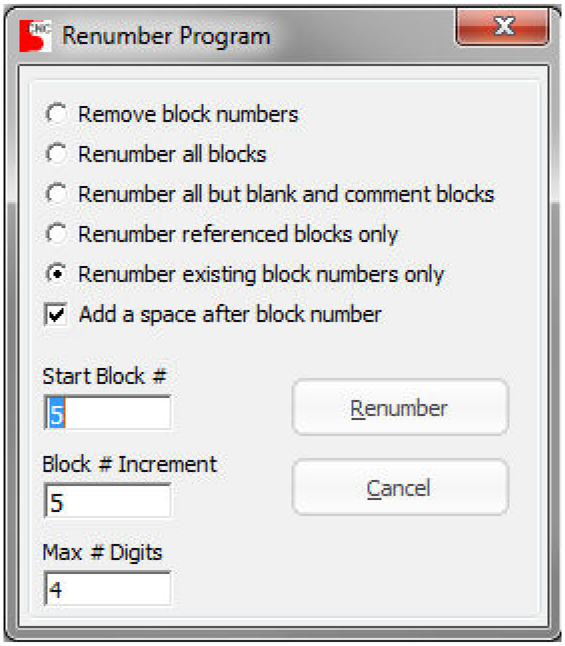
Colorize

Applies the address color settings to the current file. The color settings can be changed on the address color settings dialog found under the menu ***Setup / Address Colors***.

Format Menu

Renumber Blocks

Opens the Renumber Program dialog.



○ Remove block numbers
○ Renumber all blocks
○ Renumber all but blank and comment blocks
○ Renumber referenced blocks only
● Renumber existing block numbers only
☑ Add a space after block number

Start Block # 5

Block # Increment 5

Max # Digits 4

This dialog allows you to set up how you would like your program blocks to be numbered. The ***Start Block #***, ***Block # Increment*** and ***Max # Digits*** settings define how the program will be numbered. If the block number exceeds the maximum digits setting then it will rollover to the starting block number.

- **Remove block numbers** - This option will remove block numbers from the program with the exception of block numbers that are being referenced by other blocks. This includes the macro keyword "GOTO", local subprogram calls and canned cycles that reference block numbers.

- **Renumber all blocks** - This option will renumber all blocks including blank and comment lines.

- **Renumber all but blank and comment blocks** - This option will renumber all blocks except blank and comment lines.

- **Renumber referenced blocks only** - This option will renumber only those blocks that are being referenced by commands in other blocks. This includes the macro keyword “GOTO”, local subprogram calls and canned cycles that reference block numbers. This option removes all other block numbers.

- **Renumber existing block numbers only** - This option will renumber only those blocks that already have a block number in them.

Remove Block Numbers

This tool will remove all block numbers from the program except for block numbers that are referenced by a “GOTO” macro statement, a subprogram call or other program blocks.

Add Spaces

This tool inserts spaces between letter addresses and macro keywords to improve readability of the program. This tool does not affect text inside of comments..

```
(Testing)
G0X0Y0Z1
IF[#1GT[SQR[#2+#3]]]GOTO50
```

This becomes:

```
(Testing)
G0 X0 Y0 Z1
IF [#1GTSQR[#2+#3]] GOTO 50
```

Remove Spaces

This tool removes all spaces from the program except for text inside of comments.

Remove Leading Spaces

This tool removes any spaces from the beginning of the block.

Remove Trailing Spaces

This tool removes any spaces from the end of the block.

Remove Blank Lines

This tool removes blank lines from the program.

Remove Comments

This tool removes comments from the program. Comments are anything that is enclosed in parentheses ( ). Comments may also be enclosed in square brackets [ ], this is selectable on the ***Machine Configuration***.

Remove Garbage

This tool fixes non-standard end of line characters. Any combination of carriage return and line feed characters are converted to standard CR/LF format. This tool also converts TAB characters (ASCII 09) to spaces and removes all other control characters. It also converts any extended characters (ASCII 128 thru 255) to standard characters.

Add Comment Chars

This tool adds comment characters ( ) or [ ] to the beginning and end of each selected line in the program. If nothing is selected, then no changes are made. The comment character is selectable on the ***Machine Configuration***.

Remove Comment Chars

This tool removes comment characters ( ) or [ ] from each selected line of the program. The comment character is selectable on the ***Machine Configuration***.

Add Block Skip Chars

This tool adds the block skip character "/" to the beginning of each selected line in the program. If nothing is selected, then no changes are made.

Remove Block Skip Chars

This tool removes the block skip character "/" from each selected line of the program.

Convert to All Caps

This tool converts the program to all caps, including comments.

Remove Redundant Endpoints

Many controllers only require axis endpoints that are changed from previous blocks. This tool reduces program size by removing axis endpoints that are specified but not changed from the previous block.

Tools Menu

Display Precision

Selects the number of decimal places for coordinate display and G-Code editing tools. The display precision may be set to between 3 and 6 decimal places.

Macro Translator

This tool will execute a variable macro and translate it into standard G-Code blocks. Any blocks that contain variable commands will be output with the variables replaced with their current values. This will expand the macro program into an equivalent longhand G-Code program. This process will also expand any macro program loops and subprograms. In fact, the executed program doesn't need to be a variable macro, this tool can be used to expand a main program and sub programs into one continuous program file.

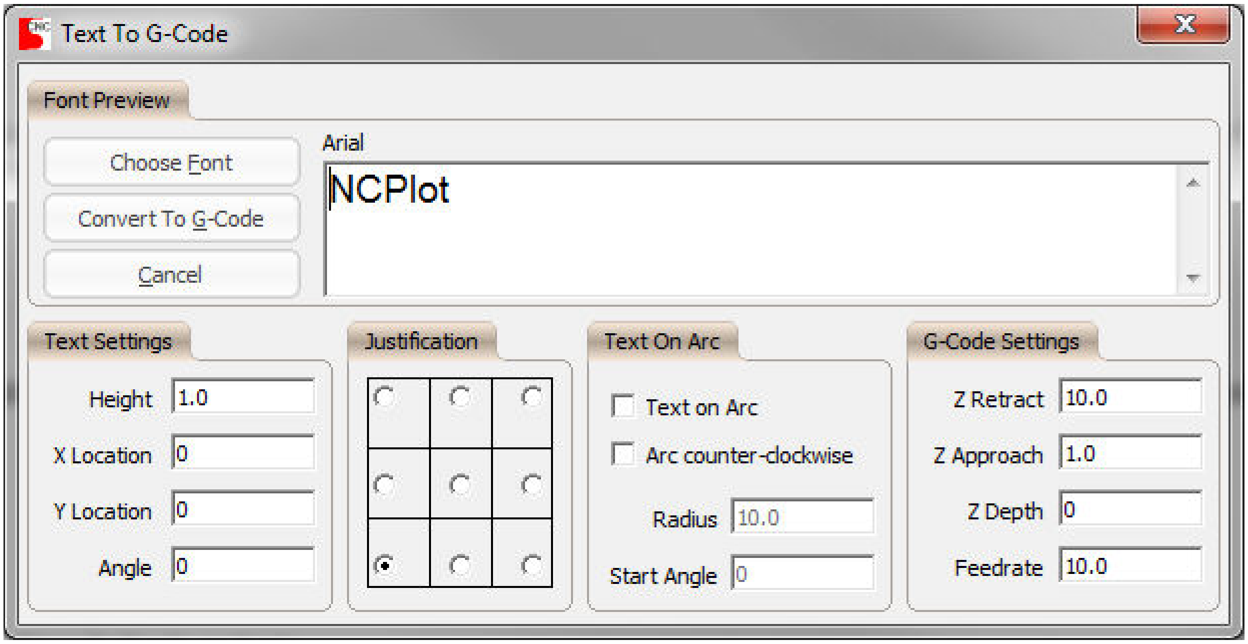
See also Macro Translator Help

Program Expand

This tool executes the currently selected edit file and converts it to a longhand G-Code program in a new document. Similar to the ***Macro Translator*** tool it will convert macro programs as well as expand macro loops and subprograms.

Text to G-Code

This tool generates G-Code that follows the outline of the entered lettering. The lettering may be in any font that you have installed on your computer.



Arial

NCPlot

Height 1.0

X Location 0

Y Location 0

Angle 0

☐ Text on Arc

☐ Arc counter-clockwise

Radius 10.0

Start Angle 0

Z Retract 10.0

Z Approach 1.0

Z Depth 0

Feedrate 10.0

Font Preview

Use the Choose Font button to select the desired font style and enter the text to convert into the font preview window. When all other settings have been made, use the Convert To G-Code button to convert the displayed text to G-Code. When entering text to convert, you may create multiple lines by pressing CTRL-ENTER to start a new line. This dialog does not close after the conversion process is complete, but the graphics view is refreshed so that you can immediately see the results of the conversion. If the resulting code is not what you want, simply undo the added code and make your changes.

Text Settings

These settings define the resulting size, location and orientation of the converted text. When the Text on Arc option is checked, the X and Y location settings become the X and Y arc center location and the Angle setting is disabled.

Justification

This setting allows you to define which location on the text will correspond to the X and Y location settings. This makes it much easier to center or align the output text. This setting is disabled when the Text on Arc option is checked.

Text On Arc

When checked, this option will create the output text on an arc. You just set the desired X and Y center of the arc under the Text Settings and then enter the radius and start angle of the arc.

G-Code Settings

These settings are used for the resulting G-Code output.

Convert Selected to G-Code

This tool makes it possible to create G-Code from any entity on the viewport. Since the viewport is made up of entities created from G-Code, this applies mainly to entities created by the Calc tools. To use this tool, first select all the entities that you want to convert to G-Code. When this tool is picked, a dialog appears that allows you to define the G-Code settings that will be used for the conversion. The selected entities are then chained together before being converted to G-Code.

Repost

The repost tool will generate G-Code from the active viewport graphic. This tool differs from the ***Convert Selected to G-Code*** tool in that this tool will output 3D code for the graphics exactly as-is.

Mirror

This tool changes the program endpoints to create a ‘Mirror image’ of the original program. The program can be mirrored in either the X axis, Y axis or both. You can also set the program coordinate that will act as the center of the mirror axis.

Rotate

This tool rotates program endpoints in any of the three planes. Simply select the desired plane, enter the rotation center coordinates and rotation angle.

Please note that if your program contains arc commands that are not in the rotation plane the resulting program will probably not function correctly. By definition, arc commands must lie in one of the three planes and rotating them will create illegal arcs. One possible solution to this is to apply the tool Break Arcs into Lines which replaces the arc commands with a series of line segments which can be rotated.

Shift

This tool adds a specified shift amount to each of the axes.

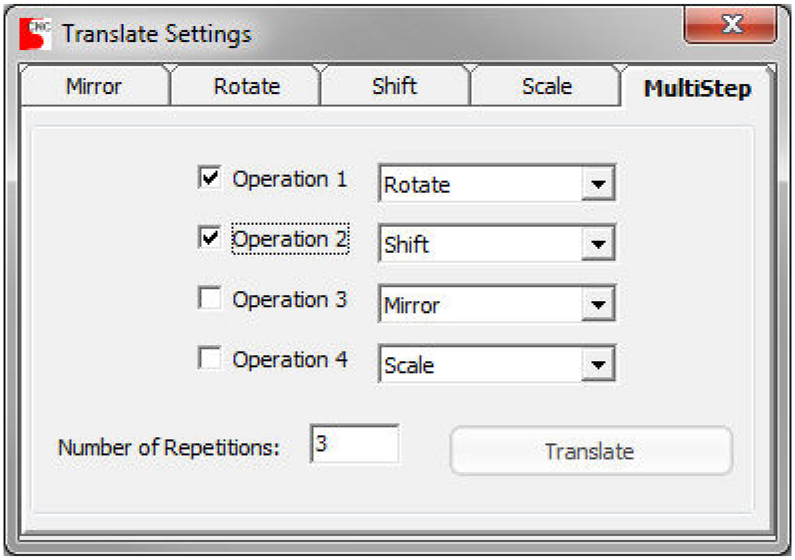
Scale

This tool applies the specified scale factor to the program.

MultiStep Translate

The MultiStep translate tool allows you to make translated copies of all or part of your program. This would be useful for making left/right hand parts, making equally spaced copies of a feature, making rotated copies of a feature or lots of other uses.

Use the MultiStep tool to enable up to four operations for each repetition. Each operation can be any of the four translations ***mirror, rotate, shift*** or ***scale***. The four tabs on the Translate Settings dialog corresponding to the four translations are used to define the settings for each operation.



☑ Operation 1 Rotate

☑ Operation 2 Shift

□ Operation 3 Mirror

□ Operation 4 Scale

Number of Repetitions: 3

The ***Number of Repetitions*** setting determines how many copies to make. The translations are applied in the order of operation, so the translation specified by

Operation 1 is performed first then ***Operation 2***, etc. The results of the enabled operations are copied to your program and then become the input for the next repetition. This means that the translations are incremental, as each repetition uses the results from the previous one.

After all settings have been configured, press the ***Translate*** button to apply the translations. If only a portion of the program has been selected for translation then the results are inserted immediately after the end of the selection. If nothing is selected then the translations are applied to the entire program and the results are appended to the end of the program. In this case remember to remove the end of program codes such as M02 or M30 or the backplot may not show the newly inserted code.

Convert Coordinates To ABS

This tool converts endpoint coordinates from G91 incremental to G90 absolute values.

Convert Coordinates To INC

This tool converts endpoint coordinates from G90 absolute to G91 incremental values.

Address Adjustments

This tool allows you to apply math operations to specific program addresses. You can add, subtract, multiply or divide the program values by a given adjustment amount. The Value Format string determines the format of the resulting values. If no adjustment is applied (by specifying * 1 or + 0 as the operation) the specified addresses will simply be reformatted. This makes it possible to change the value format of any address in the program.

Address Replace

This tool allows you to replace any program address with another. This is useful for changing the address letter for a 4th axis or when converting from one program format to another. This tool does not affect comments so it’s smarter than a simple find / replace operation.

Address Remove

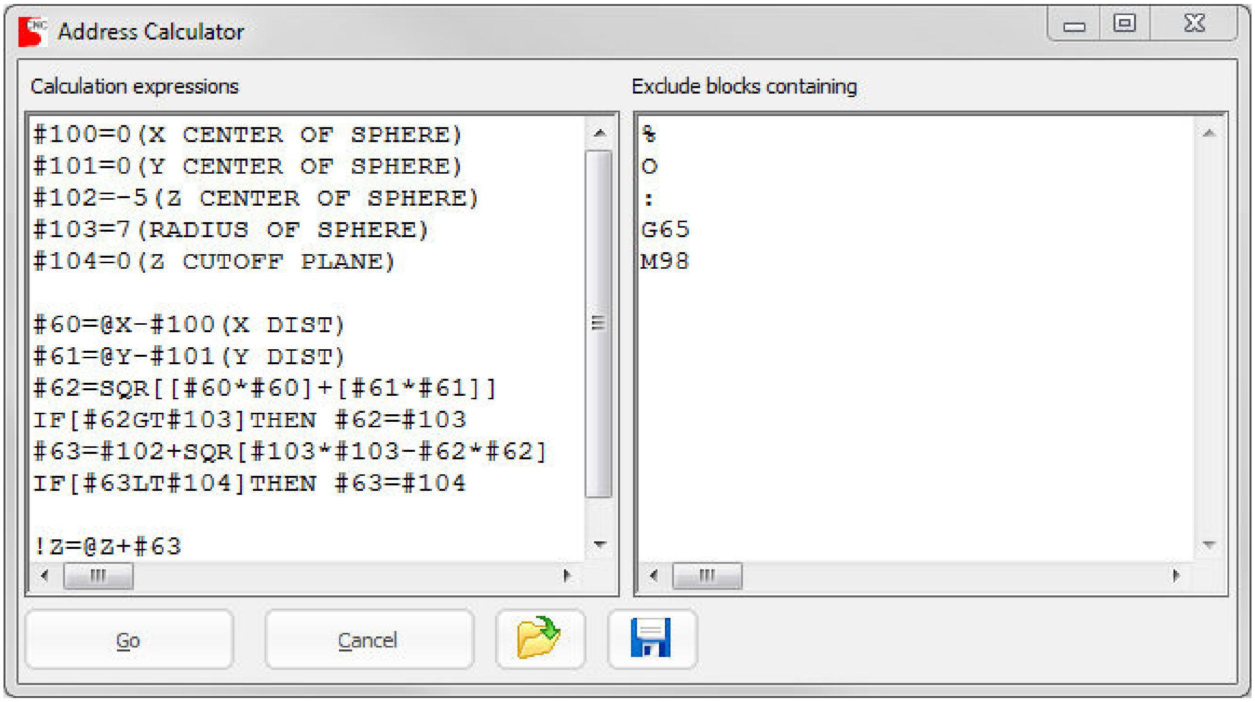
This tool removes the selected addresses and their values from the program. This also does not affect comments.

Address Swap

This tool allows you to swap any two program addresses. This tool does not affect comments so it’s smarter than a simple find / replace operation.

Address Calculator

The address calculator is a tool that allows you to create custom program translations.



Calculation expressions

```
#100=0(X CENTER OF SPHERE)
#101=0(Y CENTER OF SPHERE)
#102=-5(Z CENTER OF SPHERE)
#103=7(RADIUS OF SPHERE)
#104=0(Z CUTOFF PLANE)

#60=@X-#100(X DIST)
#61=@Y-#101(Y DIST)
#62=SQR[[#60*#60]+[#61*#61]]
IF[#62GT#103]THEN #62=#103
#63=#102+SQR[#103*#103-#62*#62]
IF[#63LT#104]THEN #63=#104

!Z=@Z+#63
```

Exclude blocks containing

```
%
O
:
G65
M98
```

For each block in the program the address values are loaded into local variables and one or more calculation expressions can be performed. The address values can be updated based on the calculation results or even removed from the program. The expressions are given in the Macro B format and can include the "IF - THEN" macro keywords.

Expression lists may be saved or loaded via the load or save buttons. By default they are saved as text (.txt) files.

This tool works similarly to the other translation tools such as ***Mirror***, ***Rotate***, ***Shift*** and ***Scale*** in that you can translate all or selected portions of the file. However, the translation to be performed is determined by the calculations you specify. Any number of calculation steps can be used as well as any local or common variables. Note that the local variables #1 through #26 are set to the values in the block being processed before the calculations are performed. When calculations are finished the addresses in the block are updated to the current state of their corresponding local variable. To simplify things the letter addresses may be used in expressions in place of their variable number.

There are many possible uses:

- Custom translations such as 3D plane rotation, twist, or wrap around a cylinder.
- Wrap text around a cylinder or sphere.
- Limit axis values to a specific range, or translate only values that fall within a specific range.
- Remove axis values when they fall in or outside of a specific range.
- Duplicate an address value to another address.
- And many others...

In place of the local variable numbers the letter addresses may be used in calculations. For example:

```
X = Z / Y
```

This statement will update block X values with the result of Z / Y. If no X address appears in the block then no change is made.

There are two modifiers that can be used with an axis address letter, these are the exclamation point "!" and the AT symbol "@". When assigning a new value to an axis address the exclamation point will add the address to a block even if it doesn't already exist. Likewise, if the assigned value is <empty> the address will be removed from the block. For example:

This statement will add the "Z" address to blocks that don't already have it.

```
!Z = SQR[[X*X] + [Y*Y]]
```

This statement will duplicate A address values to the U address:

```
!U = A
```

This statement will remove "Z" addresses from blocks that have them.

```
!Z = #0
```

The address letter shortcuts cannot be used next to a macro keyword:

```
IF[Z LE 0] THEN Z=0
```

In this case use the local variable instead:

```
IF[#26 LE 0] THEN #26=0
```

When the AT symbol is used in front of an axis address, the current absolute axis position is used in place of the value in the block. This is helpful when calculating values based on axis position. The following local variables may be used in place of the "@" shortcuts:

#96	X axis
#97	Y axis (Z axis for lathe)
#98	Z axis (invalid for lathe)
#99	4th axis (invalid for lathe)

Note that this tool does not process the program the same way that the backplotter does. System variables are not updated and it does not process subprograms, instead the file is processed from top to bottom in a line by line fashion.

Convert Arc Centers to ABS

This tool converts arc centers specified with incremental I / J / K to absolute. This tool also turns on the Absolute Arc Centers option of the Draw menu.

Convert Arc Centers to INC

This tool converts arc centers specified with absolute I / J / K to incremental distance from the arc start point. This tool turns off the Absolute Arc Centers option of the Draw menu.

Convert Arc R to I/J/K

This tool calculates the center point for R specified arcs and replaces ‘R’ with I / J / K values in either incremental or absolute. Incremental or absolute is selected via the Draw menu and is indicated by the check next to the Absolute Arc Centers menu item.

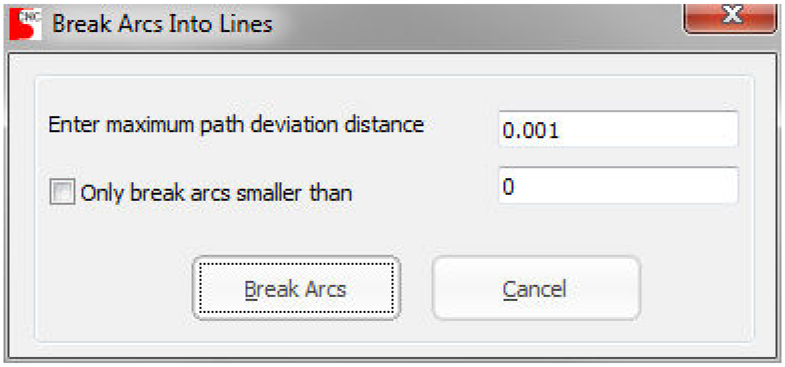
Convert Arc I/J/K to R

This tool calculates the radius of the arc and replaces I / J / K values with an R value. For arcs less than or equal to 180 degrees, the R value is positive and for arcs greater than 180 degrees, the R value is negative. This tool will not convert arcs that result in a full circle. If any arcs could not converted for this reason, a message will indicate how many arcs could not be converted.

Break Arcs Into Lines

This tool will break arcs into line segments. Any G02/G03 blocks are replaced with a sequence of G01 moves that approximate the arc. You will be prompted for a maximum deviation distance that determines how closely the line segments will follow the arc. This also determines how many line segments are required to approximate the arc. This tool will also break helical arcs up to 360 degrees.

The option ***Only break arcs smaller than*** allows breaking of only arcs that are less than the specified length.


Break Arcs Into Lines

Enter maximum path deviation distance 0.001

□ Only break arcs smaller than 0

Break Arcs At Quadrants

This tool will break arc commands that cross any of the 4 quadrant points. The 4 points are at 0, 90, 180 and 270 degrees. Up to 3 additional arc commands are created for each G02 / G03 command so that each command creates an arc that is no more than 90 degrees. This tool will not break helical arcs.

Break Lines Into Segments

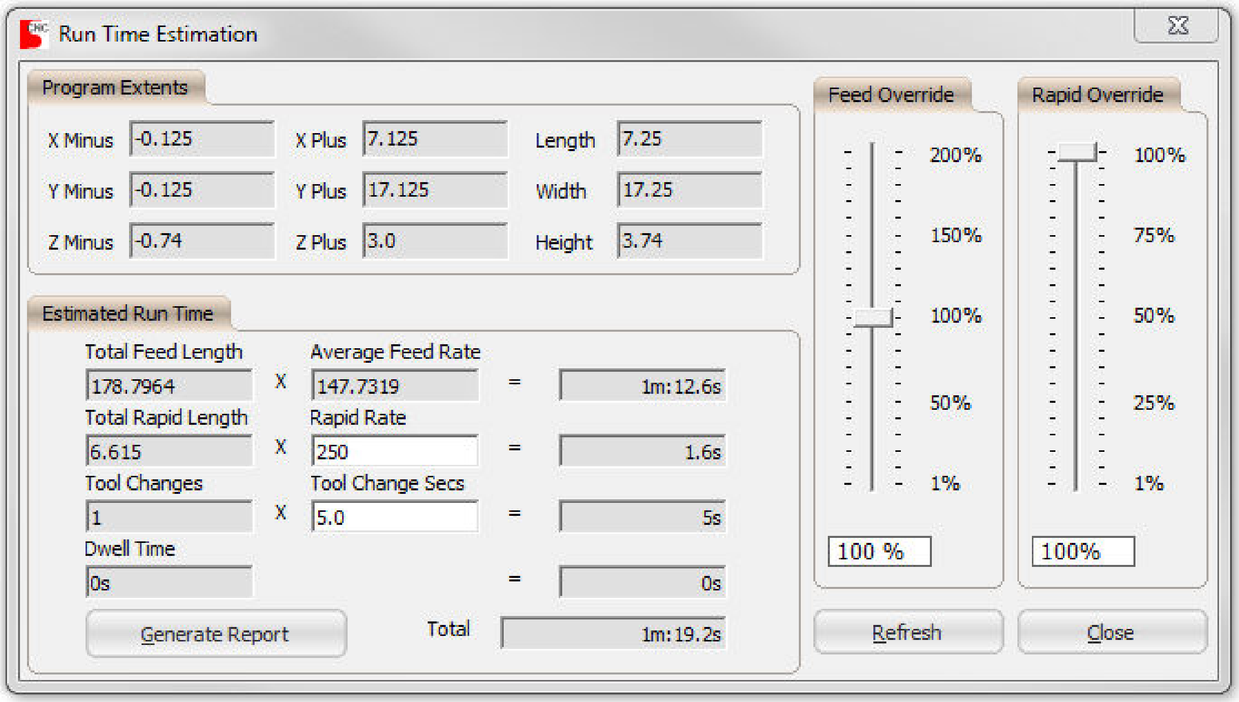
This tool will break linear move commands into multiple smaller move commands. This tool prompts you for a maximum move length which specifies how long to make each of the smaller moves. Movement commands that are equal to or smaller than the entered length will not be changed.

Run Time Estimation

This tool shows a window that displays the programmed extents for each axis and the estimated run time of the loaded file.

You can enter your machine’s rapid rate and approximate tool change time to more accurately estimate the required cycle time. These values may be stored in the machine configuration so that values specific to each machine may be set.

The feed and rapid override controls allow you to see the effects that these controls have on your machine run times. After changing either of these settings click the 'Refresh' button to recalculate the run time.



Run Time Estimation

Program Extents

X Minus	-0.125	X Plus	7.125	Length	7.25
Y Minus	-0.125	Y Plus	17.125	Width	17.25
Z Minus	-0.74	Z Plus	3.0	Height	3.74

Estimated Run Time

Total Feed Length 178.7964	X	Average Feed Rate 147.7319	=	1m:12.6s
Total Rapid Length 6.615	X	Rapid Rate 250	=	1.6s
Tool Changes 1	X	Tool Change Secs 5.0	=	5s
Dwell Time 0s			=	0s
		Total		1m:19.2s

Feed Override: 100 %

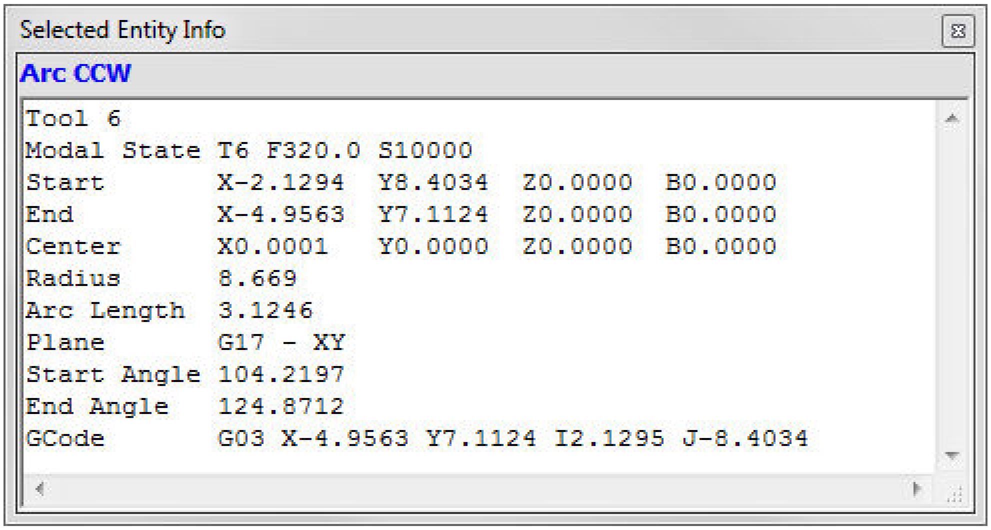
Rapid Override: 100%

Generate Report

This button will create a detailed time estimation report that gives you information about the run time for each tool in the program. This report automatically opens in the default text editor so that it may be saved or printed.

Show Entity Info

This tool shows a window that displays information about an entity on the viewport. This window pops up automatically whenever an entity is picked on the viewport. The auto pop-up can be disabled on the Preferences dialog via the ***Auto show entity info*** setting.



```
Tool 6
Modal State T6 F320.0 S10000
Start       X-2.1294  Y8.4034  Z0.0000  B0.0000
End         X-4.9563  Y7.1124  Z0.0000  B0.0000
Center      X0.0001   Y0.0000  Z0.0000  B0.0000
Radius      8.669
Arc Length  3.1246
Plane       G17 - XY
Start Angle 104.2197
End Angle   124.8712
GCode       G03 X-4.9563 Y7.1124 I2.1295 J-8.4034
```

Show Modal State

This tool reports the active modal state at the current cursor location. The graphics backplot is refreshed and a window appears that contains the active modal information at the current line. This information includes all active G-Codes, the active values of all other address values and any active canned cycle.

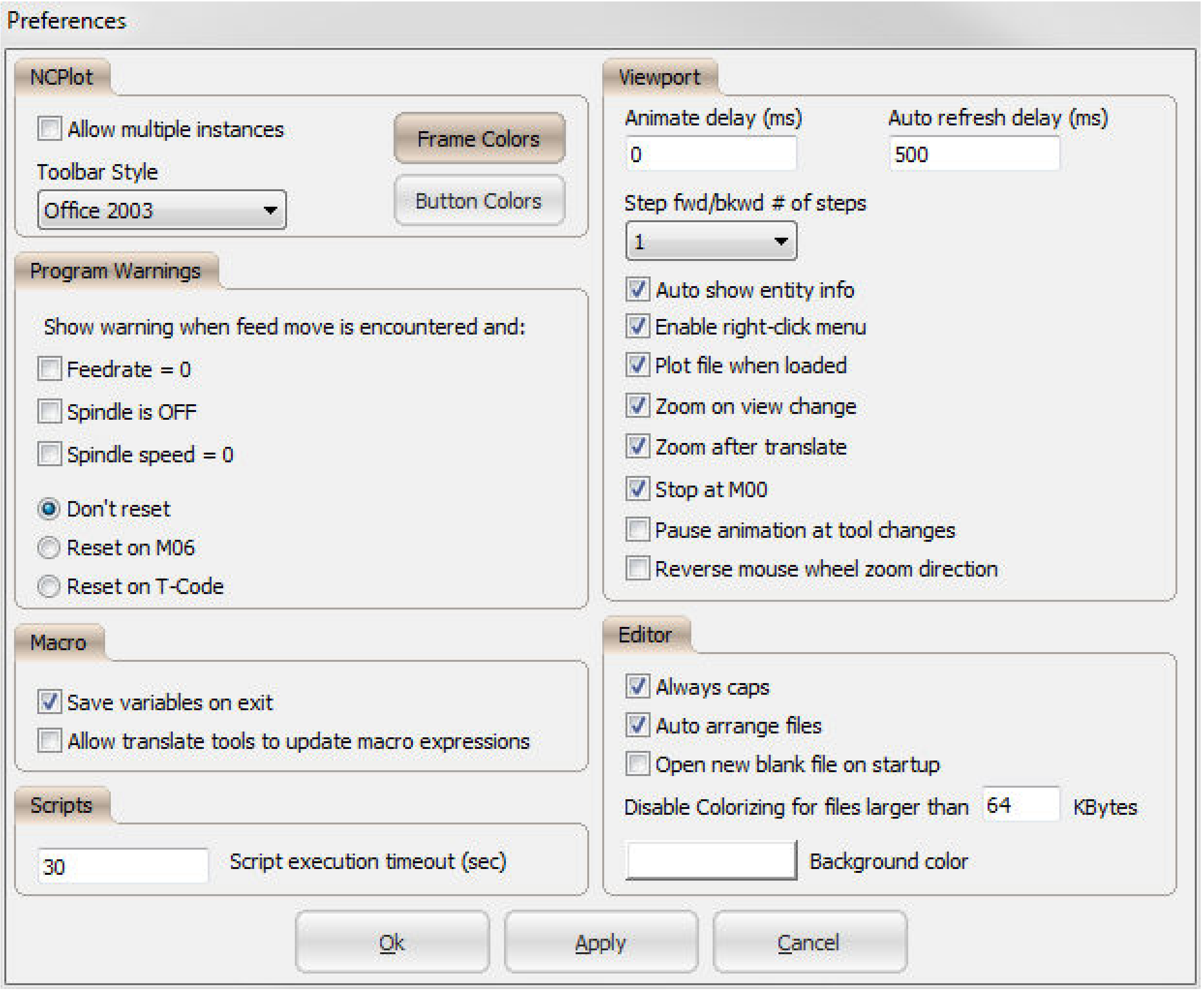
Setup Menu

Toolbars

This menu allows hiding or showing any of the toolbars by displaying a submenu containing a list of all the toolbars, including any custom toolbars. Each toolbar in the list has a checkbox next to it indicating if it is visible. To hide a toolbar, uncheck the box next to its name. To show a toolbar, check the box next to its name.

Preferences

Opens the Preferences dialog. This dialog contains default behavior settings for the editor and viewport.



Preferences

NCPlot

□ Allow multiple instances

Toolbar Style: Office 2003

Viewport

Animate delay (ms): 0

Auto refresh delay (ms): 500

Step fwd/bkwd # of steps: 1

☑ Auto show entity info

☑ Enable right-click menu

☑ Plot file when loaded

☑ Zoom on view change

☑ Zoom after translate

☑ Stop at M00

□ Pause animation at tool changes

□ Reverse mouse wheel zoom direction

Program Warnings

Show warning when feed move is encountered and:

□ Feedrate = 0

□ Spindle is OFF

□ Spindle speed = 0

● Don't reset

○ Reset on M06

○ Reset on T-Code

Macro

☑ Save variables on exit

□ Allow translate tools to update macro expressions

Editor

☑ Always caps

☑ Auto arrange files

□ Open new blank file on startup

Disable Colorizing for files larger than 64 KBytes

Background color

Scripts

30 Script execution timeout (sec)

NCPlot

- **Allow multiple instances** – When checked, you may have more than one copy of NCPlot open at a time.
- **Toolbar Style** - Allows selecting from one of four different toolbar styles. Choices are ***Office 2000***, ***Office XP***, ***Office 2003*** and ***Visual Studio 2005***.
- **Frame Colors** - Allows changing the color of the frame headers.
- **Button Colors** - Allows changing the color of the button faces.

Program Warnings

When checked, these settings will enable warning messages that indicate when a feedrate move is encountered in the program and no feedrate or spindle command has been programmed.

Allowing the warnings to reset at each tool change ensures that a feedrate and spindle command has been given for each tool in the program. You can select whether you want the warnings to reset on an M06 or a T-Code command.

Macro

- **Save variables on exit** – Check this setting if you want common variables to be saved on exit.
- **Allow translate tools to update macro expressions** - When enabled the translation tools will treat constant values in macro expressions as endpoint values and will update them as such.

Scripts

- **Script execution timeout** - After launching a script this is a delay time before you will be prompted to either kill the script execution or allow it to continue.

Viewport

- **Animate delay** – This number is a delay time in milliseconds. When the animate button is pressed the program is drawn one step at a time with this amount of delay time between steps. Note that this delay time changes when the ***Animate speed*** slider toolbar control is moved.
- **Auto refresh delay** – This setting determines how long the viewport will wait before refreshing after a change is made to the program. This setting is ignored if the ***Auto Refresh Viewport*** setting is not enabled.
- **Step fwd/bkwd # of steps** – This setting determines how many entities are drawn each time the program is stepped forward or backwards.
- **Auto show entity info** – When checked the entity info window is automatically opened when an entity on the viewport is clicked. Likewise, when all entities are unselected, the window will close.
- **Enable right-click menu** – When checked enables the right click popup menu.
- **Plot file when loaded** – When checked enables automatically backplotting a file when it is loaded into the editor.
- **Zoom on view change** – When checked the viewport will re-zoom to the part extents anytime a new view orientation is selected.
- **Zoom after translate** - When checked the viewport will re-zoom to the part extents after a translation tool has been applied to the program.
- **Stop at M00** – When checked the backplotter will pause and display a message each time it encounters an M00 in the program. When paused the backplot may be cancelled or resumed.
- **Pause animation at tool changes** - When checked program animation will stop at tool changes. Clicking the animate button again will continue with the animation.

- **Reverse mouse wheel zoom direction** - Reverses the zoom in/out direction of the mouse wheel.

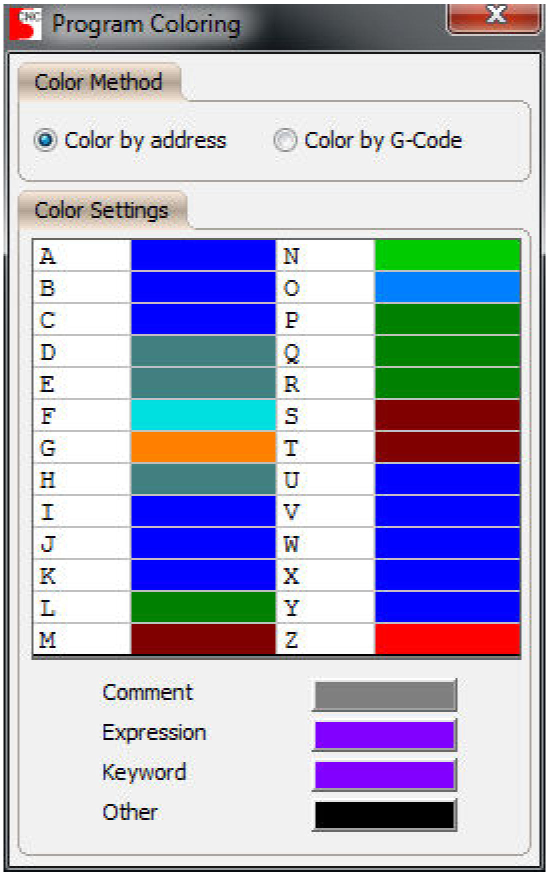
Editor

- **Always caps** – When checked, this setting forces all keystrokes in the edit window to be in upper case.
- **Auto arrange files** - This option enables the automatic tiling of the open documents.
- **Open new file on startup** - When enabled a new, blank document will be opened when NCPlot starts.
- **Background color** – This button allows you to change the background color of the edit window.
- **Disable Colorizing for files larger than** - Allows setting a size limit for colorizing the edit file. Files larger than this setting will automatically disable the program colorizing. Setting this to 0 will disable this feature.

Program Color Settings

Opens a dialog that allows you to setup how the program is colored when the ***Colorize*** tool is applied to it. There are two coloring methods to choose from: ***Color by address*** and ***Color by G-Code***.

The ***Color by address*** method applies color to each letter address and its corresponding value. To change the color settings, double-click the colored box next the address letter you want to change. A color picker dialog opens and you can select the color for that address. Color changes are automatically applied when colorizing is enabled via the menu ***Edit / Colorize***.



Color Method

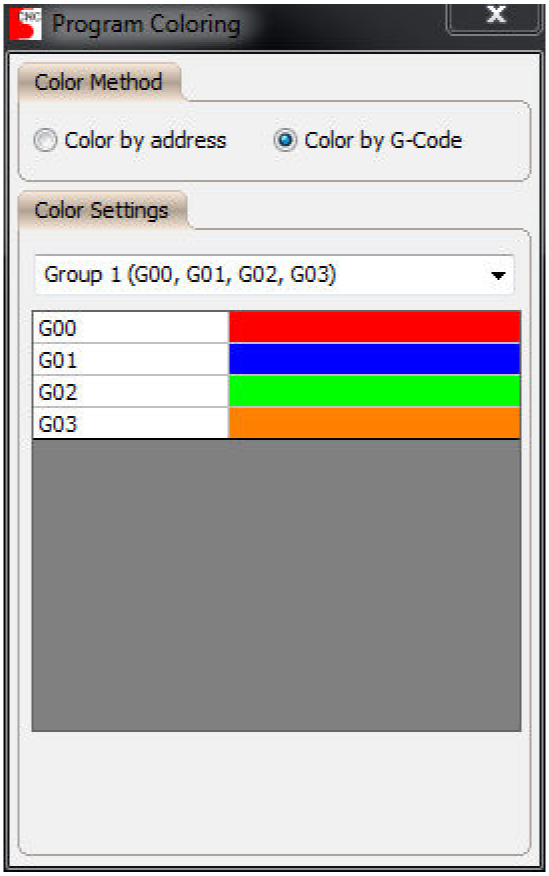
● Color by address ○ Color by G-Code

Color Settings

A	🟦	N	🟩
B	🟦	O	🟦
C	🟦	P	🟩
D	🟦	Q	🟩
E	🟦	R	🟩
F	🟦	S	🟥
G	🟧	T	🟥
H	🟦	U	🟦
I	🟦	V	🟦
J	🟦	W	🟦
K	🟦	X	🟦
L	🟩	Y	🟦
M	🟥	Z	🟥

Comment	▨
Expression	🟪
Keyword	🟪
Other	⬛

The ***Color by G-Code*** method applies color to each program line based on the active G-Codes. The G-Code group that is used to apply colors can be selected from the drop down list. To change the color settings, double-click the colored box next the address letter you want to change. A color picker dialog opens and you can select the color for that address. Color changes are automatically applied when colorizing is enabled via the menu ***Edit / Colorize***.



Program Coloring

Color Method

○ Color by address ● Color by G-Code

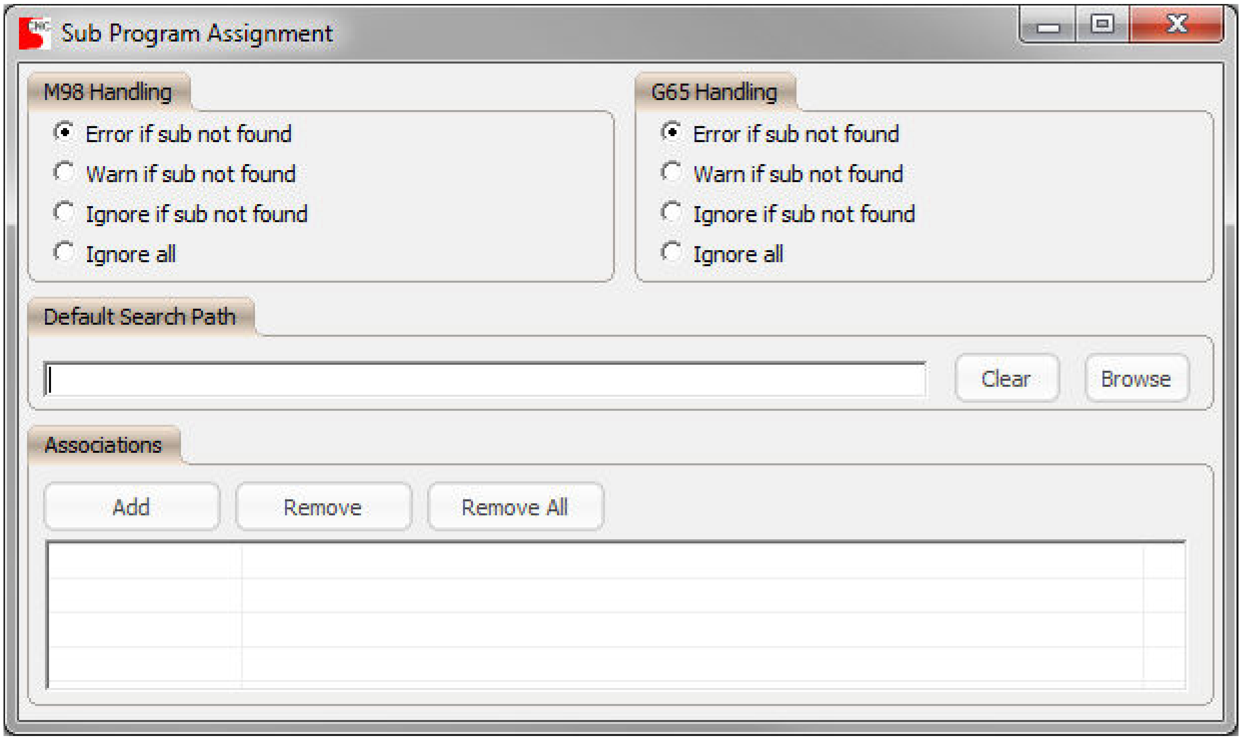
Color Settings

Group 1 (G00, G01, G02, G03)

G00	🟥
G01	🟦
G02	🟩
G03	🟧

Subprograms

Opens the sub program assignment window. This window is used to tell NCPlot where to find any sub programs that are called from your G-Code program using M98 or G65 codes.



Sub Program Assignment

M98 Handling

● Error if sub not found
○ Warn if sub not found
○ Ignore if sub not found
○ Ignore all

G65 Handling

● Error if sub not found
○ Warn if sub not found
○ Ignore if sub not found
○ Ignore all

Default Search Path

Associations

M98 and G65 Handling

These settings determine what action NCPlot will take when it encounters a subprogram command in the program:

- **Error if sub not found** – If the commanded subprogram cannot be found, NCPlot will display an error and program processing ends.
- **Warn if sub not found** - If the commanded subprogram cannot be found, NCPlot will display a message then continue.
- **Ignore if sub not found** – If the commanded subprogram is found it will be plotted, otherwise it will be ignored.
- **Ignore all** – All subprogram calls will be ignored.

Default Search Path

If you have many sub programs, or sub programs that you use often, you can keep them together in the same folder and set the default search path to point to this folder. In order for NCPlot to find it, the file name must begin with the letter ‘O’ followed by the program number. Any extension may be used, for example: O1.txt is a valid file name.

Associations

You may also assign files at random to individual program numbers. To do this, click the ‘Add’ button. You will be prompted for the program number, enter the number that follows the ‘P’ address in the M98 or G65 block. You will then be presented with the file browse dialog, use this to select the file that contains the G-Code for the entered program number.

DXF Layer Setup

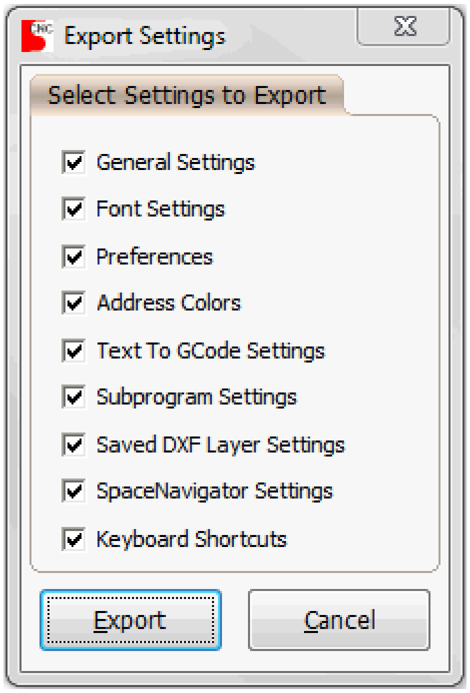
Opens the DXF Layer Setup dialog so that stored layer data may be modified without actually opening a drawing file.

Import Settings

This tool allows you to import NCPlot configuration settings from a previously exported settings file. These files are in the .INI format.

Export Settings

This tool allows you to export your NCPlot application settings to an .INI file for easy backup or transfer to another computer. These settings are:



☑ General Settings
☑ Font Settings
☑ Preferences
☑ Address Colors
☑ Text To GCode Settings
☑ Subprogram Settings
☑ Saved DXF Layer Settings
☑ SpaceNavigator Settings
☑ Keyboard Shortcuts

General Settings

These are the settings that are enabled or disabled via the menus:

- Auto Refresh
- Show Axis Lines
- Show Rapid
- Show Ticks
- Show Marker
- Show Plunge
- Lock Vertical Rotate
- Block Skip
- Editor Colorize

Font Settings

The editor font settings.

Preferences

The settings on the Preferences dialog. See also Preferences

Address Colors

The Address color settings from the Address Colors dialog. See also Address Colors

Text To G-Code Settings

The settings on the Text To G-Code dialog. See also Text To G-Code

Subprogram Settings

The settings on the Subprograms dialog. See also Subprograms

Saved DXF Layer Settings

The saved DXF layer settings. See also DXF Layer Setup

SpaceNavigator™ Settings

The current device configuration may be saved. See also SpaceNavigator™

Keyboard Shortcuts

The active menu shortcut keys. See also Menu Shortcut Keys

Machine Configuration

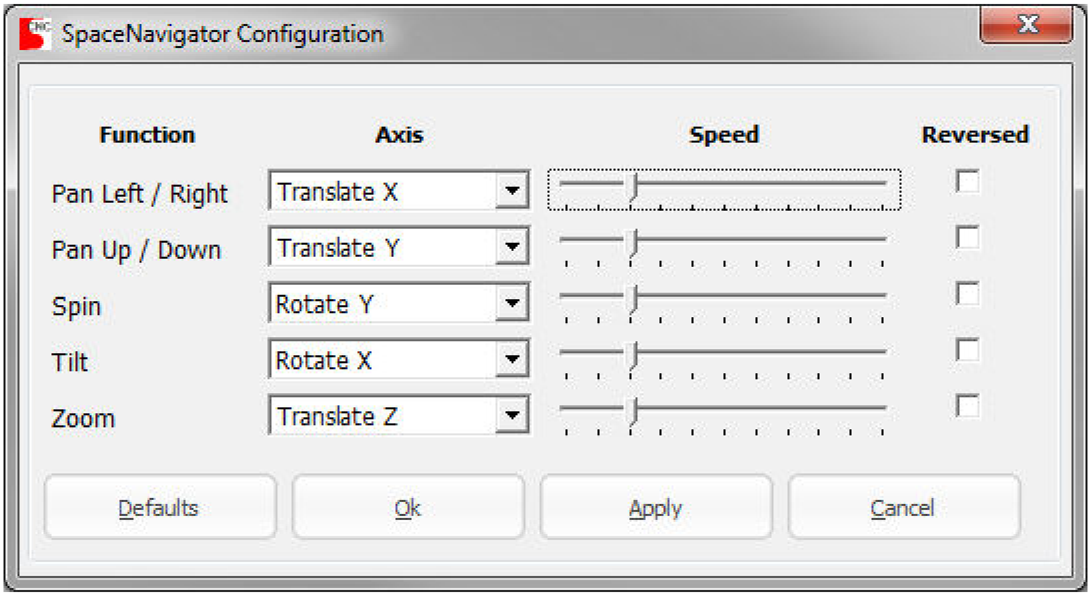
Opens the machine configuration dialog. This dialog is used to customize NCPlot to accurately simulate the way your control handles certain G-Code functions.

For more information, see About the Machine Configuration.

SpaceNavigator™

If you have a 3Dconnexion™ device such as a SpaceNavigator™, SpaceExplorer™ or SpacePilot™, NCPlot will recognize and allow you to use this device to pan, rotate and zoom the viewport.

The ***Setup / SpaceNavigator*** menu opens a configuration dialog that allows you to configure your device for use in NCPlot.



Function	Axis	Speed	Reversed
Pan Left / Right	Translate X		☐
Pan Up / Down	Translate Y		☐
Spin	Rotate Y		☐
Tilt	Rotate X		☐
Zoom	Translate Z		☐

Each Function may be controlled by any one of the six device axes. You can also adjust the speed and direction for each of the functions.

Defaults - This button resets all the device settings back to the NCPlot defaults shown here.

Ok - Accepts the current settings and closes the dialog.

Apply - Applies the current settings without closing the dialog.

Cancel - Discards any changes and closes the dialog.

Calc Menu

Expression Calculator

The Macro Calculator is a programming calculator that will evaluate expressions given in Macro B format. See also Expression Calculator.

Show Variables

The Show Variables menu enables the ***Variable Display*** window. This window is a tool designed to help you debug programs written in Fanuc Macro B format. For help on using this tool please see Macro Debugging.

Renumber Variables

The Renumber Variables menu opens the ***Renumber Macro Variables*** window. This window is a tool that will display the macro variables used by your program and allows you to renumber them. For more information please see the topic Macro Variable Renumber Tool.

Windows™ Calculator

This menu item launches the Calculator application.

Select Entities

These selection tools make it easy to select logical groups of entities on the viewport. This menu contains three submenus:

- **Select Chain** – This tool is useful for selecting 2D profiles. Just select one entity that is part of the profile, then select this tool. All entities that are connected to the selected one and at the same Z depth are selected automatically.
- **Select by Z** – This tool will go through the entire backplot and select all entities that are at the same Z depth as the currently selected entity. This makes it easy to select all profiles that are at a given depth.
- **Select Arcs by Radius** – This tool will select only arc entities that are the same radius as the currently selected arc. You could then create points at the center of each selected arc and use these points to create a drill program.

Offset

This tool will create new entities at the given offset distance from the selected entities. To use this tool, select the desired entities by either clicking them on the viewport or by selecting lines in the program and using the Highlight Selected tool on the edit menu. After the desired entities are selected, go to the menu Calc | Offset. You will be prompted for the desired offset distance. Entering a positive distance will offset to the left, while entering a negative distance will offset to the right. Once created, the new entities will stay on the viewport until it is refreshed with the plot button or a new file is loaded.

Intersect

This tool will find the intersection points between two entities. To use this tool, first select the two entities you want to find the intersection points for. Then select the Intersect tool. The point or points are calculated and displayed on the viewport. You also get a message on the status bar that tells you the coordinates of the points. This tool will solve for Line / Line, Line / Arc, and Arc / Arc intersections even if the selected entities do not visibly touch each other.

Blend Radius

This tool will create an arc of the specified radius that is tangent to two lines. To use this tool, first select two intersecting lines. After selecting this tool, you will be prompted for the desired blend radius. Enter the radius and an arc will be created on the viewport that is tangent to both lines at the given radius.

Point at Center

This tool will create points representing the center of each of the selected entities.

Measure

This tool will report the X, Y, Z and overall distance between the endpoints of the two selected endpoints.

View Menu

Top View

The Top view mode displays the axes with the X+ direction toward the right side of the screen and the Y+ toward the top. This view mode is only available when the Mill machine type is active.

Bottom View

The Bottom view mode displays the axes with the X+ direction toward the left side of the screen the Y+ toward the top. This view mode is only available when the Mill machine type is active.

Front View

The Front view mode displays the axes with the X+ direction toward the right side of the screen and the Z+ toward the top. This view mode is only available when the Mill machine type is active.

Back View

The Back view mode displays the axes with the X+ direction toward the left side of the screen and the Z+ toward the top. This view mode is only available when the Mill machine type is active.

Right View

The Right view mode displays the axes with the Y+ direction toward the right side of the screen and the Z+ toward the top. This view mode is only available when the Mill machine type is active.

Left View

The Left view mode displays the axes with the Y+ direction toward the left side of the screen and the Z+ toward the top. This view mode is only available when the Mill machine type is active.

Isometric View

The Isometric view mode displays the X, Y and Z axes in a 3D view. This view mode is different from all others because it can be rotated to show the part from any angle. This view mode is only available when the Mill machine type is active.

Front Turret View

The Front Turret view mode displays the axes with the X+ direction toward the bottom of the screen and the Z+ direction toward the right. This represents a machine where the tool approaches the work from the front of the machine. This view mode is only available when the Lathe machine type is active.

Back Turret View

The Back Turret view mode displays the axes with the X+ direction toward the top of the screen and the Z+ direction toward the right. This represents a machine where the tool approaches the work from the back of the machine. This view mode is only available when the Lathe machine type is active.

Vertical Left

The Vertical Left view mode displays the axes with the X+ direction toward the left side of the screen and the Z+ direction toward the top. This represents a vertical lathe where the tool approaches the work from the left side of the machine. This view mode is only available when the Lathe machine type is active.

Vertical Right

The Vertical Right view mode displays the axes with the X+ direction toward the right side of the screen and the Z+ direction toward the top. This represents a vertical lathe where the tool approaches the work from the right side of the machine. This view mode is only available when the Lathe machine type is active.

Set As Reference

Copies the current backplot to the background.

Clear Reference

Clears the background graphic.

Pan

After selecting this tool, use either mouse button to drag the viewport to the desired view center. This tool can also be activated by pressing the ‘P’ key when the viewport is active.

Zoom Extents

Sets the view center and zoom size to fit the entire program in the viewport. This tool is also activated by pressing the ‘X’ key when the viewport is active.

Zoom All

Sets the view center and zoom size to fit the entire program in the viewport. This tool is different from the Zoom Extents tool in that this tool will not include rapid motions when fitting the view. This tool is also activated by pressing the ‘A’ key when the viewport is active.

Zoom Selected

Sets the view center and zoom size to fit only the selected entities. This tool is also activated by pressing the ‘S’ key when the viewport is active.

Zoom In

Enlarges the view size while keeping the current view center. This tool is also activated by pressing the ‘C’ key when the viewport is active.

Zoom Out

Reduces the view size while keeping the current view center. This tool is also activated by pressing the ‘V’ key when the viewport is active.

Zoom Window

Using the left mouse button, drag a box around the desired view area. This tool is also activated by pressing the ‘Z’ key when the viewport is active.

Lock Vertical Rotation

When this item is checked, the isometric viewport will only rotate around the viewport vertical axis. This can be overridden by holding the Ctrl key while rotating the viewport. After rotating the view and releasing the Ctrl key, the viewport will maintain the new horizontal rotation.

Set ISO View Rotation Center

This tool allows you to select an endpoint to be the center of rotation for the isometric view. When a point is picked the view will pan so that the picked point is at the center of the viewport.

Draw Menu

Animate

Draws the loaded file while highlighting the corresponding blocks in the program window. There is a delay between steps so you can see the motion that is taking place. This delay can be set on the preferences dialog.

Pause

Pauses an animation in progress. Once paused, pressing Animate again will resume the program.

Rewind to Beginning

Clears the backplot graphics and sets the cursor back to the first motion block in the program.

Forward to End

Draws the backplot graphics from the current start position up to the end of the program and sets the cursor to the last motion block in the program.

Step Forward

Draws the next motion block in the program and highlights the corresponding block in the program window.

Step to Next Tool

Draws the program up to the next tool change.

Step to Previous Tool

Un-draws the program back to the previous tool change.

Refresh Plot

Refreshes the viewport with the current contents of the program window.

Plot To Cursor

Draws the file from the beginning to the current cursor location.

Start at Cursor

This item clears the viewport and sets the current program step point to the line in the program that the cursor is on. You can then animate, step forward or step backward from this point.

Plot From Cursor

Draws the file from the current cursor location to the end.

Plot Selected Blocks

This tool draws just the selected portion of the program.

Auto Refresh Viewport

This menu item is an on/off toggle setting that enables automatic refreshing of the viewport graphics after program edits are made. A check mark next to this menu item indicates that auto refreshing is enabled. There is a setting on the ***Preferences*** dialog called ***Animate Delay*** that determines how long NCPlot will wait between a change being made and refreshing the graphics.

Axis Lines

Displays two or three (depending on the view) intersecting lines that indicate where 0,0,0 is on the viewport. The actual location of the axis lines depends on the selected submenu item. A check mark indicates which item is selected.

Off – No axis lines are displayed
Machine Zero – The axis lines represent the Machine Zero location
G54-G59 Work Zero – The axis lines represent the selected Work Zero location

Show Rapid Moves

When checked, this option enables drawing of G00 rapid moves.

Show Ticks

When checked, this option enables drawing of tick marks at the endpoints of rapid moves.

Show Marker

When this menu item is checked, an arrow is drawn on the viewport which indicates the current plot endpoint. This marker also displays the coordinate values of the indicated endpoint.

Show Plunge Moves

When checked, this option enables drawing of G1 moves in the Z- direction.

Absolute Arc Centers

When checked, this option specifies how I / J / K specified arc centers are drawn. This option also affects the results of the arc conversion tools. This setting may also be changed on the Machine Configuration dialog under the Control Options tab.

DNC Menu

Send

Sends the entire contents of the edit window to the DNC tool. This tool buffers the data to be sent so that you can continue to work in NCPlot while the transfer is taking place.

When the DNC tool is open, click “Start” to begin the transmission, “Pause” to pause the transmission, or “Cancel” to abort. There are three status panels at the bottom of the window. The left panel displays the word “Waiting” whenever an XOFF character is received to indicate that the transmission has paused for a handshaking signal. The middle panel displays the elapsed transmission time, and the right panel displays the name of the program being sent.

Send Selected

Similar to the SendSend tool, but only the selected program text is sent to the DNC tool.

Send File

Similar to the SendSend tool, but allows you to browse for and send a file other than the current edit file.

Receive

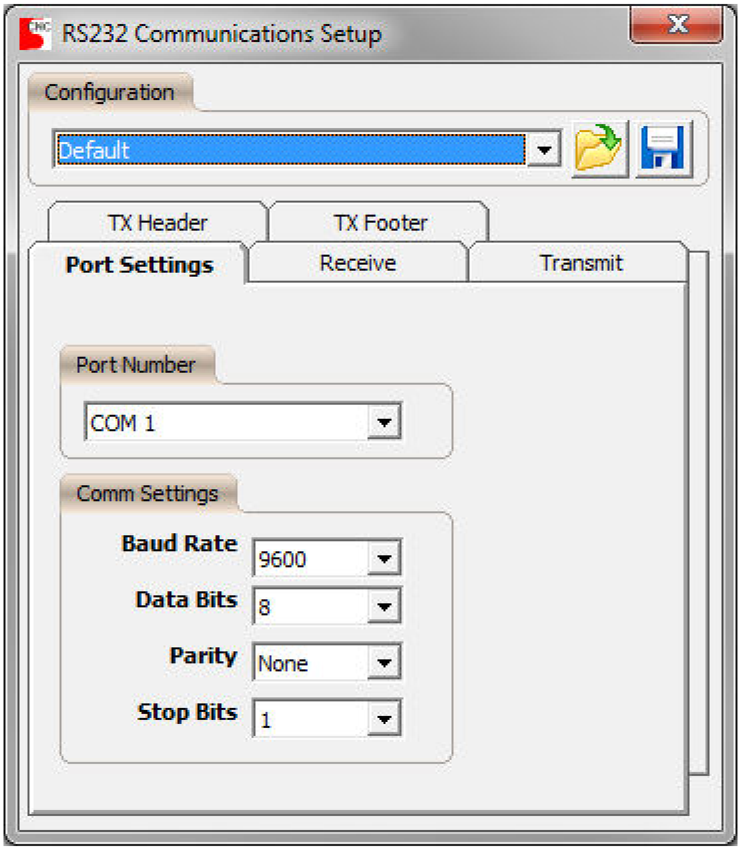
Opens the COM port for receiving. Once opened, any received data is inserted into the edit window at the current cursor location.

Receive as New

Similar to the ReceiveReceive tool, but closes the current edit file before opening the port for receiving.

Comm Setup

Opens the RS232 setup dialog.



Configuration: Default

Port Number: COM 1

Comm Settings

Baud Rate: 9600

Data Bits: 8

Parity: None

Stop Bits: 1

Configuration – The currently active configuration name.

The remaining settings are organized into three tabs: Port Settings, Receive and Transmit.

The Port Settings tab contains the basic port setup such as the COM port number, baud rate, data bits, etc.

- Port – Selects the active COM port number.
- Comm Settings – Selects the desired baud rate, number of data bits, parity and number of stop bits.

The Receive tab contains timeout settings for receiving data from the machine.

- Timeout before receive – After the COM port has been opened and nothing is received within this set amount of time, the COM port will close. If this setting is zero then NCPlot will wait indefinitely for data to be received.
- Timeout after receive – Once data has been received, the COM port will close after nothing more has been received for this set amount of time.

The Transmit tab contains settings for transmitting data to the machine.

- Handshaking – This setting is currently ignored, the XON/XOFF handshaking method is always used.
- Remove spaces while transmitting – When this setting is enabled any spaces in the program will not be transmitted to the machine.
- Wait for XON before transmitting – Enabling this setting will cause NCPlot to wait for an XON character to be received from the machine before beginning transmission. This allows you to start the data transmission from NCPlot and then go to the machine to begin receiving.
- Wait for XXX seconds before transmitting - Enabling this setting causes transmitting to begin after the specified time delay.

TX Header Tab allows you to set text to be transmitted before the main transmission begins.

TX Footer Tab allows you to set text to be transmitted after the main transmission ends.

Window Menu

Tile Vertically

This tool will automatically arrange the open documents side by side and size them to fill up the document workspace.

Tile Horizontally

This tool will automatically arrange the open documents top to bottom and size them to fill up the document workspace.

Cascade

This tool will automatically arrange the open documents so that they overlap with their window titles displayed.

Tile

This tool will automatically arrange the open documents side by side and top to bottom so that they are roughly the same size.

Close All

Closes all open files.

Viewport

This menu item provides a shortcut for showing or activating the viewport. When the viewport toolbar has been turned off this tool first turns it back on then makes it active. If the viewport is already visible then this tool simply makes it active.

See also Viewport Keyboard Shortcuts.

Help Menu

Quick Start Reference

This menu item will open the help file to the “Getting Started” topic.

Macro Programming Reference

This menu item will open the help file to the “Macro Programming Reference” topic.

Scripting Reference

This menu item will open the help file to the “Scripting Reference” topic.

Release Notes

Opens the release notes. This is a notepad document that details the most recent changes to NCPlot.

About NCPlot

Displays a window showing the NCPlot version number and your license status.

This window also allows you to de-register NCPlot on your computer. To do this you must enter the original product key that was used to register the software and then click the "De-Register" button.

If your NCPlot trial period has expired and you have a network license manager installed you can click "Check for License Server" to enable connecting to the license server.

NCPlot on the Web

This menu item will open your default web browser to the NCPlot home page www.ncplot.com.

Ordering NCPlot

This menu item will open your default web browser to the NCPlot online ordering page.

Enable Network License Server

Checking this menu item will enable communications to the license manager. This requires that the Network License Manager be running on a computer that is connected to the network.

Check Out License

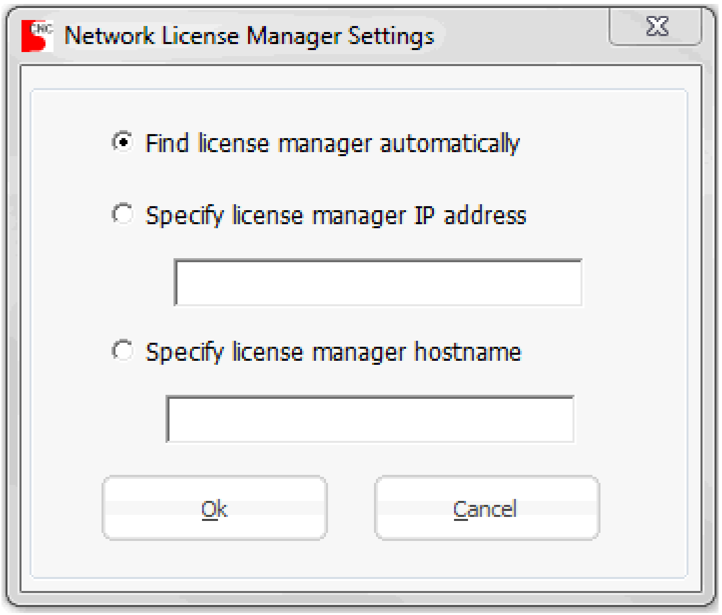
This will request a license transfer from the license manager. If successful, the license is stored on the client computer so that NCPlot may be run while disconnected from the network. While it is checked out the license will not be available for other users.

Check In License

This will return a stored license back to the license manager. This tool may also be used to add a license to the license manager. If you have registered NCPlot on a computer, its license may be moved to the license manager by first enabling the license server and then checking the license into the license manager. If the license does not already exist in the license manager, it will be added. This is very handy if you are adding a license manager to a group of computers that already have NCPlot installed and registered.

License Manager Settings

Opens the license manager settings dialog. This dialog allows you to specify a network IP address or hostname for the license server or let NCPlot automatically detect the server.



Network License Manager Settings

● Find license manager automatically

○ Specify license manager IP address

○ Specify license manager hostname

Supported G & M Codes

Mill G-Codes

G00	Rapid motion. NCPlot supports displaying rapid motion as interpolated (straight line), dog leg or Z first/last.
G01	Feed motion. NCPlot supports G01 corner rounding and corner chamfering. Including an R value in a G01 block will create a radius that is tangent to the lines created by the current block and the next block. Including a C value in a G01 block will create a chamfer between the current block and the next block.
G02	Clockwise arc. The G02 arc command supports G17, G18 and G19 modes, absolute and incremental I / J / K specified center point, R specified center point, and helical interpolation.
G03	Counterclockwise arc. The G03 arc command supports G17, G18 and G19 modes, absolute and incremental I / J / K specified center point, R specified center point, and helical interpolation.
G15	Polar coordinate system off
G16	Polar coordinate system command
G17	XY plane selection
G18	ZX plane selection
G19	YZ plane selection
G28	Reference point return
G50	Coordinate system scaling off
G51	Coordinate system scaling
G50.1	Mirror image off
G51.1	Mirror image on
G52	Local work shift
G53	Position relative to machine zero
G54	Work offset 1. Also supports extended work offsets P1 through P300.
G55	Work offset 2
G56	Work offset 3
G57	Work offset 4
G58	Work offset 5
G59	Work offset 6
G54.1	Extended work offsets P1 through P300.
G65	Macro subprogram call

G66	Macro modal call
G67	Macro modal call cancel
G68	Coordinate system rotation
G69	Coordinate system rotation off
G80	Canned cycle cancel
G70-G89	Canned cycles
G90	Absolute coordinate system
G91	Incremental coordinate system
G92	Coordinate system setting
G98	Canned cycle initial point return
G99	Canned cycle R point return

NCPlot does not handle the following G-Codes internally. Instead, they are simulated by external macro programs. This allows them to be customized for a particular control.

G12	Clockwise circle mill
G13	Counterclockwise circle mill
G34	Bolt circle canned cycle
G35	Holes on line at angle canned cycle
G36	Holes on arc canned cycle
G37.1	Grid pattern canned cycle
G70	Bolt circle cycle
G71	Bolt hole arc cycle
G72	Holes on line at angle cycle

NCPlot assigns each G-Code a group number. Only one G-Code within a group may be active at a time.

Group 0	G4, G9, G10, G11, G12, G13, G28, G34, G35, G36, G37.1, G52, G53, G65, G92
Group 1	G0, G1, G2, G3
Group 2	G17, G18, G19
Group 3	G90, G91
Group 5	G94, G95
Group 6	G20, G21
Group 7	G40, G41, G42
Group 8	G43, G44, G49
Group 9	G70-G89
Group 10	G98, G99
Group 11	G50, G51

Group 12	G54-G59, G54.1
Group 14	G66, G67
Group 16	G68, G69
Group 18	G15, G16
Group 19	G50.1, G51.1

Lathe Format A G-Codes

G00	Rapid motion
G01	Feed motion
G02	Clockwise arc
G03	Counterclockwise arc
G28	Reference point return
G50	Coordinate system set
G52	Local coordinate system set
G53	Position relative to machine zero
G54	Work offset 1. Also supports extended work offsets P1 through P300.
G55	Work offset 2
G56	Work offset 3
G57	Work offset 4
G58	Work offset 5
G59	Work offset 6
G54.1	Extended work offsets P1 through P300.
G65	Macro subprogram call
G66	Macro modal call
G67	Macro modal call cancel
G80	Canned cycle cancel
G70-G89	Canned cycles
G90	Turning cycle
G92	Thread cutting cycle
G94	Facing cycle
G96	Spindle constant surface speed (CSS) mode
G97	Spindle RPM mode
G98	Feed per minute mode
G99	Feed per revolution mode

NCPlot assigns each G-Code a group number. Only one G-Code within a group may be active at a time.

Group 0	G4, G9, G10, G11, G28, G50, G52, G53, G65
Group 1	G0, G1, G2, G3
Group 5	G98, G99
Group 6	G20, G21
Group 7	G40, G41, G42
Group 9	G70-G89, G90, G92, G94
Group 12	G54-G59, G54.1
Group 14	G66, G67
Group 17	G96, G97

Lathe Format B G-Codes

G00	Rapid motion
G01	Feed motion
G02	Clockwise arc
G03	Counterclockwise arc
G28	Reference point return
G52	Local coordinate system set
G53	Position relative to machine zero
G54	Work offset 1. Also supports extended work offsets P1 through P300.
G55	Work offset 2
G56	Work offset 3
G57	Work offset 4
G58	Work offset 5
G59	Work offset 6
G54.1	Extended work offsets P1 through P300.
G65	Macro subprogram call
G66	Macro modal call
G67	Macro modal call cancel
G80	Canned cycle cancel
G70-G89	Canned cycles
G90	Absolute coordinate system
G91	Incremental coordinate system

G92	Coordinate system set
G94	Feed per minute mode
G95	Feed per revolution mode
G96	Spindle constant surface speed (CSS) mode
G97	Spindle RPM mode
G98	Canned cycle initial point return
G99	Canned cycle R point return

NCPlot assigns each G-Code a group number. Only one G-Code within a group may be active at a time.

Group 0	G4, G9, G10, G11, G28, G52, G53, G65, G92
Group 1	G0, G1, G2, G3
Group 3	G90, G91
Group 5	G94, G95
Group 6	G20, G21
Group 7	G40, G41, G42
Group 9	G70-G89
Group 10	G98, G99
Group 12	G54-G59, G54.1
Group 14	G66, G67
Group 17	G96, G97

M-Codes

This list is all the M-Codes recognized by NCPlot.

The spindle control M-Codes are used to provide warning messages when a program commands feed motion without the spindle running. These messages may be enabled on the ***Preferences*** dialog.

M02, M30	End of program
M03	Spindle forward
M04	Spindle reverse
M05	Spindle stop
M13	Spindle forward & coolant on
M14	Spindle forward & coolant on

The mirror image M-Codes may be changed to match your control. This may be set on the ***Machine Configuration*** dialog.

M87	X axis mirror ON
M88	Y axis mirror ON
M89	X & Y axis mirror OFF

Multiple subprogram call formats are supported. Use the ***Machine Configuration*** dialog to select the format that your control recognizes. For details about choosing a subprogram format, please see Plotting Subprograms.

M97	Local subprogram call
M98 P1 L1	L, K = Repeat count
M98 P1 K1	xxxx = Repeat count
M98 Pxxxxyyyy	yyyy = Program number
M98 P1 H1 L1	H = Block number
M98 O1	O = Block number
M98 P1 Q1 L1	Q = Block number
M99 P1	Return from sub program

When M99 is commanded in a sub program, the P value specifies the block number to return to. When commanded in the main program, the P value causes execution to jump to the specified block number.

Adding Custom G-Codes

Additional cycles may be added to simulate G-Codes not explicitly supported by NCPlot. Open the ***Machine Configuration*** dialog to the ***G/M Codes*** page and enter additional G-Codes in the ***G-Code Macros*** list. The active configuration folder must contain a corresponding program file to handle the simulation of the additional codes. The macro program file name is the corresponding G-Code value times 10. For example, the file name for G81 is G810.PRG. This is to accommodate decimal G-Codes such as G37.1.

Custom G-Codes are handled like a G65 subprogram, addresses in the block are copied to local variables and the appropriate file is called as a subprogram.

Macro B Programming Support

What is Macro Programming

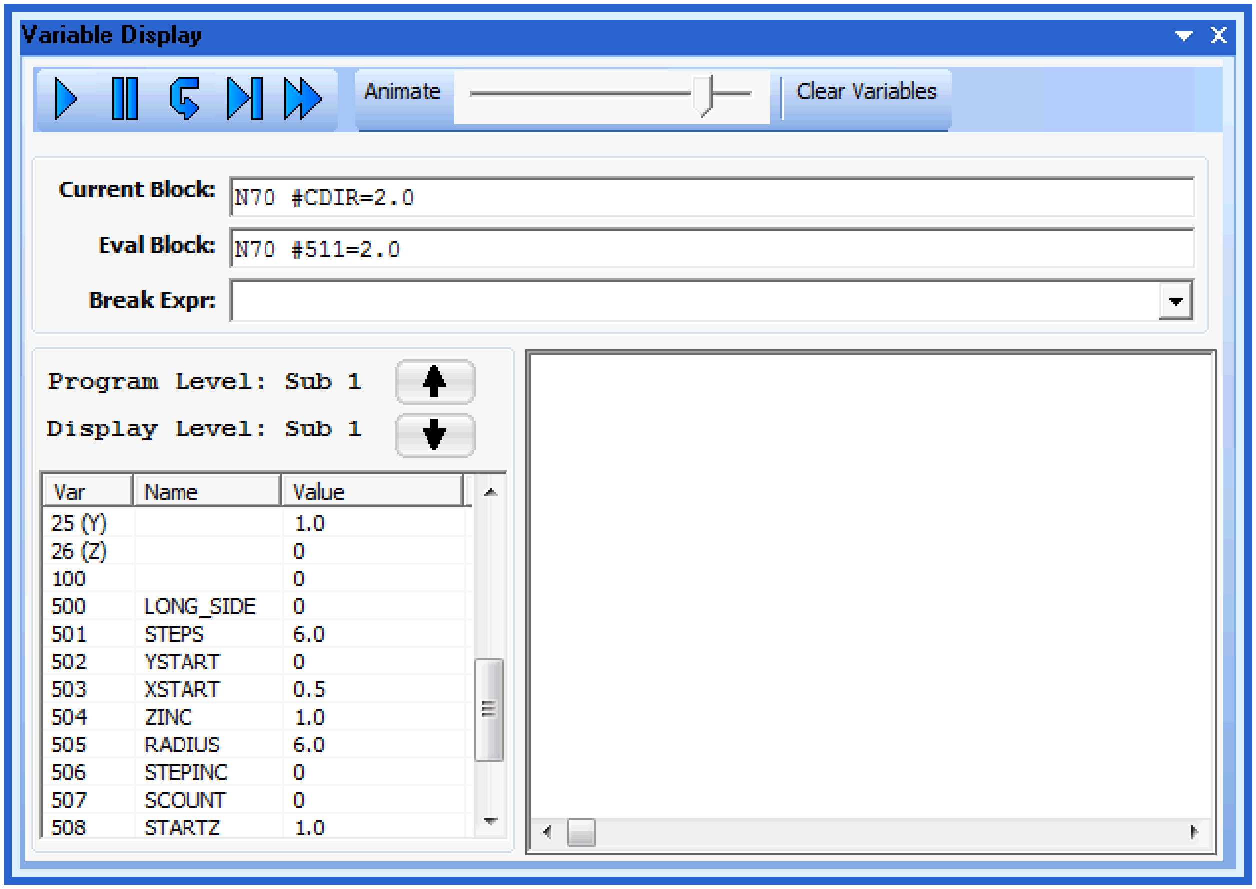
The macro language, often referred to as Macro B or Custom Macro B, is a programming language that gives the CNC programmer the ability to write very flexible programs. This is done through the use of variables, mathematical expressions and program flow control statements. The macro language combined with standard G-code programming can create reusable programs much like canned cycles. These programs can do many useful things like custom pocketing or automatic tool measurement.

This document describes how NCPlot processes macro programs and may differ from your specific control.

Macro Debugging

The Variable Display window is an indispensable tool for anyone writing macro programs. This window not only displays the current state of program variables, it enables you to execute macro programs one block at a time. This window can be accessed from the "Calc" menu, via the "Show Variables" item, or by pressing the "Alt-B" keyboard shortcut.

Local variables may be displayed for each of the subprogram levels. This window also allows you to follow program execution into subprograms not contained in the loaded program file. This program stepping differs from the main window. When stepping from the variable display window, the program is executed as it is stepped so you can see the results of variable commands. On the main window, the program is fully executed before stepping begins, so the state of the variables reflect the end result of the program.



Current Block: N70 #CDIR=2.0

Eval Block: N70 #511=2.0

Break Expr:

Program Level: Sub 1

Display Level: Sub 1

Var	Name	Value
25 (Y)		1.0
26 (Z)		0
100		0
500	LONG_SIDE	0
501	STEPS	6.0
502	YSTART	0
503	XSTART	0.5
504	ZINC	1.0
505	RADIUS	6.0
506	STEPINC	0
507	SCOUNT	0
508	STARTZ	1.0

This window provides several important tools for debugging macro programs. Several controls are combined into a toolbar:

- **Macro Animate** – Executes the program one block at a time with a timed interval in between blocks. The animation stops when pause is clicked or a break point block is encountered. The variable display is updated after each block.
- **Pause** – Stops a program being executed.
- **Macro Step** – Executes one block of the program and updates the variable display.
- **Run to Cursor** – Executes the program at high speed until it reaches the block that the cursor is on.
- **Run to Break** – Executes the program at high speed until a break expression becomes True or a break point block is encountered. The break expressions are tested after every program block.
- **Clear Variables** – Allows clearing of all local and common variables to empty.

The ***Current Block*** display shows the next block to be executed as it appears in the program.

The ***Eval Block*** display shows the same block after customizations have been applied to it and any expressions replaced with their evaluated values. This allows you to see the resulting block before it is executed.

The break expression drop down is a list of conditional macro expressions that will stop program execution when the evaluated result transitions from FALSE to TRUE or any non-zero result. This is useful for stopping macro execution when a variable reaches a particular value, for example. When you type an expression and press 'Enter' it will be added to the drop down list. All of the expressions in the list are evaluated after every block. When a break occurs the expression that caused it will be displayed as the selected item.

You can place explicit break points in your program by adding a line containing the text "(BREAK)" where you want the break point to occur.

The local variable display level may be changed using the up and down arrow buttons. This allows checking local variables for each of the subprogram levels.

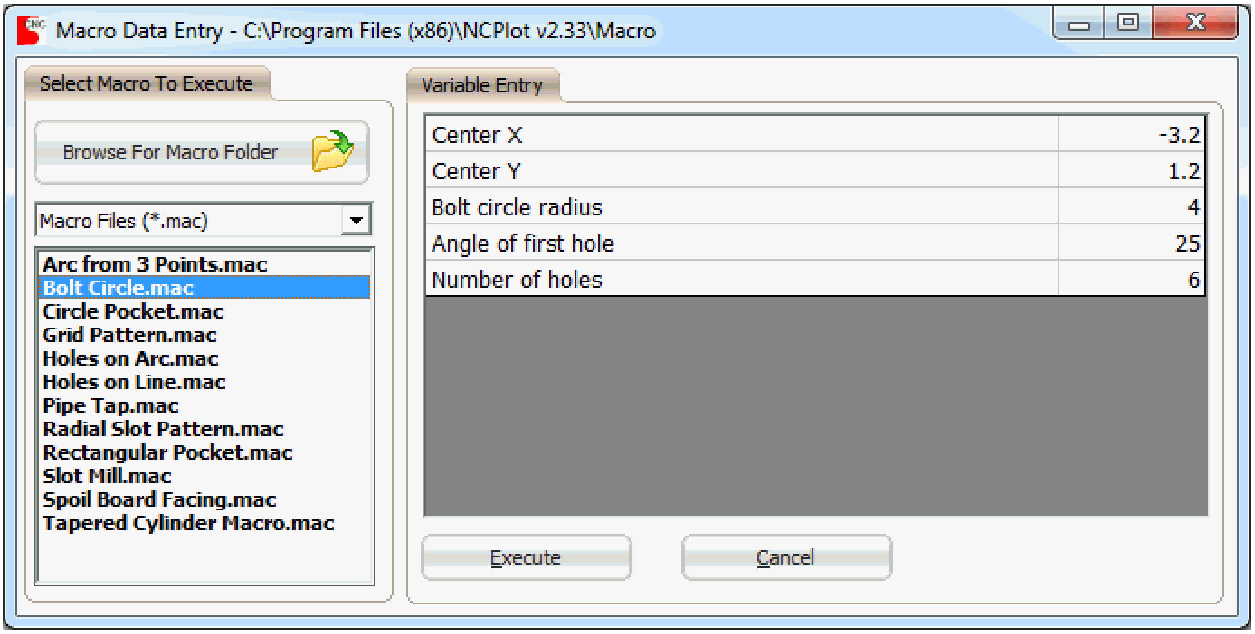
Right-clicking the variable display will pop up a menu with these options:

- **Edit Variable Value** – This option prompts for a new value for the selected variable. This can also be done by double-clicking on the variable list.
- **Edit Variable Name** – This option prompts for a new name for the selected variable. Note that adding or editing variable names this way is not restricted by the named variable number range set by the machine configuration.
- **Delete Variable Name** – Will clear the variable names associated with any of the selected variables.
- **Add System Variable** – This option allows adding any system variable to the variable display. You will be prompted for the variable to add, simply enter the variable number you want to add. Once added it will remain until you remove it. You can add multiple variables at a time by specifying two variable numbers separated by a dash, like so: "5021-5024".
- **Sort Variable List** – Sorts the variable list by variable number. Occasionally the list may become out of order as variables are added or removed, use this tool to re-sort the list.
- **Remove System Variable** – This option will remove the selected system variables from the variable display. If a selected variable is not a system variable then it will not be removed.
- **Copy Variables to Clipboard** – Copies the current variable list to the clipboard as text.
- **Print Variables** – Sends the current variable list to the default printer.

Macro Translator

This tool will execute a variable macro and translate it into standard G-Code blocks. Any blocks that contain variable commands will be output with the variables replaced with their current values. This will expand the macro program into an equivalent longhand G-Code program. This process will also expand any macro program loops and subprograms. In fact, the executed program does necessarily need to be a variable macro, this tool can be used to expand a main program and sub programs into one continuous program file.

Before the macro is executed, you can set any required variable values on the translator dialog window. Comment blocks at the head of the macro are used to define the required variables and are displayed on the translator dialog when the macro program is selected. The values that are entered are saved to a file and reloaded each time that the macro is selected, so that you don't need to re-type the settings each time you use it.



- Arc from 3 Points.mac
- Bolt Circle.mac
- Circle Pocket.mac
- Grid Pattern.mac
- Holes on Arc.mac
- Holes on Line.mac
- Pipe Tap.mac
- Radial Slot Pattern.mac
- Rectangular Pocket.mac
- Slot Mill.mac
- Spoil Board Facing.mac
- Tapered Cylinder Macro.mac

Center X	-3.2
Center Y	1.2
Bolt circle radius	4
Angle of first hole	25
Number of holes	6

As an example, the Bolt Circle cycle asks for the X, Y center location, radius, start angle and number of holes. After this information is entered, click the "Execute" button and the resulting G-Code is inserted into your program. There are a number of cycles included with NCPlot, but you may also add your own.

The header format for using macro programs with this tool is very simple:

```
(Bolt hole circle)              Name of macro or cycle

(#24 Center X)                  Required variables and
descriptions
```

```
(#25 Center Y)
(#4  Bolt circle radius)      These descriptions appear on the
(#5  Angle of first hole)     translator dialog.
(#6  Number of holes)
```

The translator also allows you to create programs that contain blocks or characters that would not normally be output. To do this, a special comment block format is used:

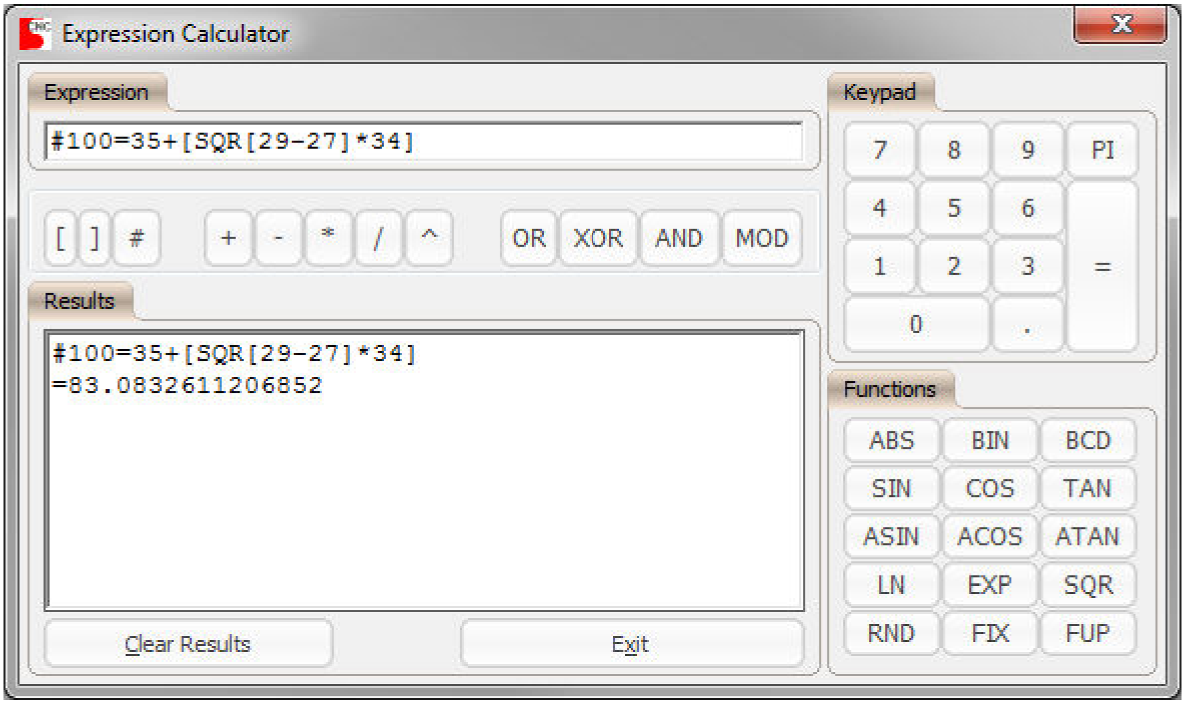
```
("%")
("O1234(TEST PROGRAM)")
```

When the translator encounters these blocks the characters inside the double quotes are output exactly as-is.

Macro Calculator

This calculator solves mathematical expressions. There are 15 functions and 9 operators and allows unlimited bracket groupings. The expression and the result are added to the result window for easy referencing back to any earlier expression.

This calculator allows the use of local, common and system variables in expressions and can also be used to assign values to variables.



Expression:

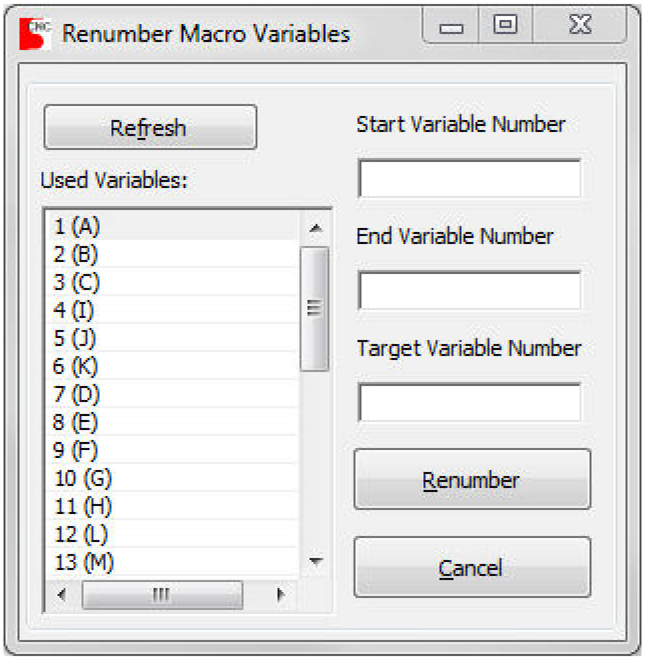
```
#100=35+[SQR[29-27]*34]
```

Results:

```
#100=35+[SQR[29-27]*34]
=83.0832611206852
```

Macro Variable Renumber Tool

This tool can renumber the macro variables used by your programs as well as display a list of the variables being used.



To use this tool enter the start and end variable number that you want changed as well as the target variable number then click "Renumber". The block of variable numbers from start to end are renumbered beginning at the target variable number.

Variables before renumbering:		Variables after renumb
100	Start Variable Number:	33
101	100	34
105		38
106	End Variable Number:	39
107	113	40
108		41
110	Target Variable Number:	43
111	33	44
112		45

113		46

From this example you can see that the variable numbers are not re-sequenced, but rather they are moved in their existing sequence to a new block of variable numbers. Be careful to specify a target variable number that does not overlap with variables that are currently being used.

The used variable list is refreshed after renumbering or when the "Refresh" button is clicked.

Note that this tool will not report or renumber variables that are indirectly referenced.

Canned Cycles

The backplotting of canned cycles in NCPlot is not hard coded. These cycles are Macro B programs stored in a folder that belongs to the selected Machine Configuration. The canned cycle macros are located at:

\Program Files\NCPlot v2.xx\Config\*config name*

The Config folder contains the machine configuration files along with a corresponding sub folder for each configuration. These sub folders contain the canned cycle macro programs. Every machine configuration uses it’s own set of canned cycle macros, which means that you can have different canned cycles for each machine configuration.

For information about adding support for additional G-Codes, please see Adding Custom G-Codes.

The G-Codes that are handled as external macros by default are:

G70 – G79, G81 – G89 Standard canned cycles

You can add cycles for G-Codes that fall within these ranges by simply adding an appropriately named G-Code program file to the desired configuration folder. No additional configuration settings are required.

G12	Clockwise circle cutting
G13	Counterclockwise circle cutting

The G34 through G37.1 cycles are included in the default mill configurations.

G34	Bolt circle cycle
G35	Holes on line at angle cycle
G36	Holes on arc cycle
G37.1	Grid pattern cycle

The G70 through G72 bolt pattern cycles are included in the HAAS mill configuration.

G70	Bolt circle cycle
G71	Holes on arc cycle
G72	Holes on line at angle cycle

NCPlot uses system variables to pass some additional information to the canned cycles. These variables are specific to NCPlot.

#5100 Machine Type
0 = Mill
1 = Lathe Radius
2 = Lathe Diameter

#5101 Canned Cycle Absolute Depth
The absolute position of the commanded canned cycle depth.

#5102 Canned Cycle Absolute R plane
The absolute position of the commanded canned cycle "R" plane.

#5103 Canned Cycle Absolute Initial point
The absolute position of the commanded canned cycle initial point.

#5110 4th axis address assignment
This variable will contain a value that indicates the letter address assigned to the 4th axis. The possible values are 1 (A), 2 (B), 3 (C), 21 (U), 22 (V) and 23 (W).

Macro Programming Reference

What are Variables

Variables are the heart of macro programming. Variables are like numbered storage units, each of which can either be empty or contain a number. When they contain a number, the variables can be used in place of almost any numeric value in your G-code program.

Variables are designated with the "#" symbol and are followed by a number or expression that designates the variable number. Variables can be used as the value following any letter address except “N”.

Examples of how variables can be used:

```
G1 X#100 Y#101 F10.
```

In this example of a feed move, the X and Y endpoints are determined by the values contained in the variables #100 and #101.

```
G#100 X#101 Y#102
```

In this example, the G-code to be executed is determined by the value contained in variable #100.

What makes them such a powerful tool is the fact that new values can be assigned to variables by your program.

Here is an example of assigning a value to a variable:

```
#100=10.0
```

The value 10.0 is stored into variable #100. Once this has been set, the value 10.0 will be used in place of #100.

```
G1 X#100
```

This command is then equivalent to `G1 X10.0`

By simply changing the value stored in this variable, you can make the same program do different things.

Multiple variables may be assigned in the same block. However, variable assignments must be on a block by themselves and may not be combined with G-code blocks or other macro statements.

Examples:

`#101=1 #102=2 #103=3` This is valid

`IF [#1EQ0] THEN #105=5` This is the only macro statement where variables can be assigned.

`G01 X#101 #102=4` This is not valid

`WHILE [#1NE0] DO1 #104=4` This is not valid

It is good programming practice to decide the meaning of your variables before you begin writing. For example, you may want variable #100 to be used to specify a pocket width, or you may want variable #250 to specify a corner radius. It is completely up to you what the variables mean, but it can save a lot time and confusion to decide before you begin writing.

What are Expressions

Expressions are formulas that mathematically combine values to get a resulting number. The result of an expression is then assigned to a variable or used in a conditional statement.

Example of using an expression to assign a value to a variable:

```
#100 = #101+1
```

The expression is the part following the "=" sign and is "#101+1". So 1 is added to the contents of #101, and this value is then stored in variable #100.

Expressions may use any combination of operators, functions and comparisons. If no brackets are used, the values are calculated in the standard arithmetic order. That is, multiply and divides are performed first, followed by addition and subtraction.

Examples of expressions:

#100 = 10 + 2 * 3 – 5

The value 11 is stored in variable #100

#101 = 8 * 3 – 3 * 4

The value 12 is stored in variable #101

Operators are basic mathematical operations and include:

+	Addition
-	Subtraction
*	Multiplication
/	Division
^	Raised to the power of
MOD	Modulus, the remainder of a division operation
XOR	Bitwise XOR
OR	Bitwise OR
AND	Bitwise AND

Brackets may be used to change the order that the expression is evaluated in. When brackets are used in an expression, the calculations inside the brackets are performed first, then the rest of the expression is calculated. The calculations inside the brackets are still performed in standard arithmetic order:

```
#100= [10 + 2 * 2] * 3 – 5
```

This expression evaluates to 37

`#100= 8 * [3 - 3] * 4`

This expression evaluates to 0

The variable numbers themselves may be replaced with expressions, as long as the expression evaluates to a valid variable number. For example:

`#[100+5] = 10`

Variable number #105 is assigned a value of 10

Variables may also be used to store variable numbers. This is called variable indirect and is a very useful feature. It can also be confusing, so here are some examples:

`#100 = 105`	
`#[#100] = 10`	The value 10 is assigned to variable number #105

Here's what happens:

`#100 = 105`	Variable #100 now contains the value 105
`#[105] = 10`	The expression in brackets is evaluated first
`#105 = 10`	The variable is assigned using the result of the expression as the variable number.

It is possible to use multiple levels of indirection:

`#1 = 2`	
`#2 = 3`	
`###1 = 10`	Variable #3 ends up with the value 10.
`#[#[#1]] = 10`	The same expression using brackets.

Functions

In addition to operators, there are many functions that may be used in an expression. A function takes a value and calculates a resulting value based on it's particular function. Unlike an operator, a function only needs one number to work from. The values used for functions must be enclosed in brackets.

For example:

`#100 = SQR[2]`

Variable #100 is assigned the value 1.414213 which is the square root of 2.

Here is a complete list of available functions:

`SIN`	Calculates the Sine of an angle in degrees
`COS`	Calculates the Cosine of an angle in degrees
`TAN`	Calculates the Tangent of an angle in degrees

`ATN, ATAN`	Calculates the ArcTangent
`ACOS`	Calculates the ArcCosine
`ASIN`	Calculates the ArcSine
`LN`	Calculates the natural logarithm
`SQR, SQRT`	Calculates the square root
`ABS`	Calculates the absolute value
`BIN`	Converts decimal to hexadecimal
`BCD`	Converts hexadecimal to decimal
`RND, ROUND`	Value rounding
`FIX`	Returns the integer portion of a value
`FUP`	Fractional values are rounded up to the next whole number
`EXP`	Exponent

The value passed to a function may be any valid expression:

```
#102 = SQR[#100 + #101]
```

The contents of variables #100 and #101 are added and the square root of this result is calculated.

Comparisons

The comparators are typically used with the IF - GOTO or IF - THEN macro statements. However, they may be used in any expression. When used as part of an expression, they return a value of 1 for TRUE or 0 for FALSE as their result. Comparators are like operators in that they require two operands.

List of comparators:

`EQ`	Equal to
`NE`	Not equal to
`LT`	Less than
`LE`	Less than or equal to
`GT`	Greater than
`GE`	Greater than or equal to

Comparison examples:

```
#100 = #101 EQ #102
```

If #101 equals #102 then #100 is assigned a value of 1. If not, it is assigned a value of 0.

If comparators are used as part of a larger expression, the comparison should be enclosed in brackets like this:

```
#100 = 10 * [#101 EQ #102] + 5
```

If #101 equals #102 then the expression evaluates to 15, otherwise it evaluates to 5.

Bitwise operators

Bitwise operators are a convenient way of combining comparisons. The bitwise operators are OR, AND and XOR.

"OR" operator:

```
#100 = [#101 EQ #102] OR [#103 EQ #104]
```

Variable #100 is assigned a value of 1 if #101 equals #102 OR #103 equals #104, otherwise it gets a value of 0.

"AND" operator:

```
#100 = [#101 EQ 1] AND [#102 EQ 2]
```

Variable #100 is assigned a value of 1 if #101 equals 1 AND #102 equals 2, otherwise it gets a value of 0.

"XOR" operator:

```
#100 = [#101 EQ 1] XOR [#102 EQ 2]
```

Variable #100 is assigned a value of 0 if both expressions are true or both expressions are false. If one is true and the other is false, #100 is assigned a value of 1.

Program Levels and Local Variables

Local variables are variable numbers #1 to #99 and are typically used as temporary use variables for subprograms. Even though the variable numbers are the same, each program level has it's own set of local variables. The program level changes any time you call or return from a subprogram. The main program is always level 0. A subprogram call from the main program is level 1.

The G65 macro call command allows passing values to the subprogram through local variables. This table shows the correspondence between the letter addresses and the variable numbers. The "x" denotes letters that cannot be used to pass variables.

Variable	Address
#1	A
#2	B
#3	C

#7	D	
#8	E	
#9	F	
#10	G	x
#11	H	
#4	I	
#5	J	
#6	K	
#12	L	x
#13	M	
#14	N	x
#15	O	x
#16	P	x
#17	Q	
#18	R	
#19	S	
#20	T	
#21	U	
#22	V	
#23	W	
#24	X	
#25	Y	
#26	Z	

Local variable example:

```
G65 P9000 X10 Y5 Z1 A2.5 B3.6
```

When this block is executed, these values are assigned to local variables before program 9000 begins. Program 9000 can then read these values in these variables:

#24=10	(X)
#25=5	(Y)
#26=1	(Z)
#1=2.5	(A)
#2=3.6	(B)

These are called local variables because these values are only valid for the program they are passed to. If program 9000 also used a G65 macro call, the local variables are saved and the new subprogram gets it's own set of local variables. They do not overwrite the values being used by program 9000. So when the new subprogram is done, the variables being used by program 9000 have not changed even though both subprograms use the same variable numbers. When a subprogram is finished, it's local variables are cleared.

Common and System Variables

Common variables are variable numbers #100 to #999 and, unlike local variables, can be used and set by any program. These variables are also retained on exit.

System variables are variable numbers #1000 on up. The actual range will depend on your specific control. System variables are used by the CNC to store internal values needed for operation. These values are things like tool length offsets, diameter offsets and machine positions. These variables may be set by your program, but care should be taken when doing so.

Using Variable #0

Variable #0 is a special variable that cannot be set. Instead, its value is always <empty>. A variable that is <empty> is not the same as a variable that has been set to 0. An <empty> variable is a variable that contains no value, not even 0.

This variable cannot be set, but may be assigned to other variables and used in comparisons. This is especially useful for subprograms in determining if a value has been given for all required addresses:

```
G65 P9000 A0 B2
```

```
#1=0 (A)
#2=2 (B)
```

Variables #1 and #2 contain values because values were given in the G65 block. All other local variables are cleared to <empty>. If, for example, the address C is also required by the subprogram, it can check to see if a value was given by comparing it to variable #0. If #3 is equal to #0, then address C was not included in the G65 block that called the subprogram. Even though #1 was set to 0, this is a valid value and so it is not equal to #0.

Variables may be cleared to <empty> by assigning them the variable #0.

For example:
#100 = #0
The variable #100 is cleared to <empty>.

This also works using the variable indirect method:

```
#100 = 0
#110 = #[#100]
```

Variable #110 is cleared to <empty> because #100 points to variable #0.

When variable #0 is used in an expression, it is handled as the value 0 except for the comparisons EQ and NE.

Examples of using #0 in expressions:

```
#0 + 1 = 1
#0 * 10 = 0
#0 + #0 = 0
#[#101-#101] * 50 = 0

[#0 LT 0]  FALSE
[#0 LE 0]  TRUE

[#0 EQ 0]  FALSE
[#0 NE 0]  TRUE
```

These two comparisons are the only two that handle #0 and 0 as being different.

```
[#0 GT 0]  FALSE
[#0 GE 0]  TRUE
```

In general, the variable #0 shouldn't be used in expressions. Its use should be limited to EQ and NE comparisons and variable assignments.

Macro Statements

In addition to variables and expressions, the macro language uses a few macro statements that can control the flow of the program.

Here is a list of the macro statements:

```
     IF - GOTO
     IF - THEN
     GOTO
     WHILE - DO
     DO
     END
     SETVN
```

You'll notice that a few of the statements are grouped together. This is because the statements work together to determine the exact function performed.

These macro statements are recognized but not supported by NCPlot:

```
     POPEN
     DPRNT
     BPRNT
     PCLOS
```

These statements are recognized by the interpreter and the formatting tools, but are ignored by the backplotter.

Examples:

```
IF [#100EQ1] GOTO100
```

When this block is executed, the expression "[#100EQ1]" is evaluated, and if it is TRUE then the statement GOTO100 causes the program to jump to N100. If it is FALSE, the program execution continues to the next block.

The expressions used by IF statements do not have to be comparisons. The expression is considered to be TRUE if the result of the expression is non-zero. So any valid expression may be used with the IF statement.

```
IF [#100LT#101] THEN #102=5
```

When this block is executed and the expression "[#100LT#101]" is TRUE, then the variable #102 is assigned a value of 5. Otherwise, #102 is not changed and execution continues with the next block.

```
IF [#100GT#101] THEN G0 X#100
```

If the expression "[#100GT#101]" is TRUE, then the X axis moves to the position contained in variable #100. Any valid G-code block may follow the THEN statement.

```
GOTO200
```

When this block is executed, the program jumps to N200. Since no IF statement is used, this is called an unconditional jump.

```
WHILE [#100LT#101] DO1
...
...
...
END1
```

The WHILE - DO statements set up a conditional loop. When the WHILE - DO block is first encountered, the expression "[#100LT#101]" is evaluated. If this expression is FALSE, the program jumps to the block that contains the END statement. If this expression is TRUE, the program continues until the END statement is reached. When the END is reached, the WHILE expression is evaluated again. If this expression is still TRUE, the program jumps back to the WHILE - DO block, and the process repeats. So essentially, the program blocks between the WHILE - DO block and the END block are repeated until the expression evaluates to FALSE.

The DO and END statements have numbers after them that identify which END block belongs to which WHILE - DO block. This is because WHILE - DO loops may nested inside each other. The valid loop numbers are from 1 to 30.

```
DO2
...
...
...
END2
```

This is a WHILE - DO loop without the WHILE. It's conditional expression is always TRUE, which sets up a never ending loop. This type of loop must have some other means of breaking out of the loop, such as an IF - GOTO statement or even an M02 or M30 to end the program.

Named Variables

Some CNC controls allow you to assign names to variables. In your macro programs you can then use these names in place of the variable numbers. The ability to assign names to variables is typically limited to a range of variable numbers, such as #500 to #599.

Naming variables in NCPlot can be done in a couple of ways. There is the SETVN macro statement which can be included in your part program, or from the variable display window. The SETVN statement takes this form:

```
SETVN500[DIST,RADIUS,TOOL2]
```

The SETVN statement is followed by a variable number and then a list of names within square brackets. The names will be assigned to the variable numbers starting with the specified number. Variable names must start with a letter and can contain only letters, numbers and the underscore character (_).

The variable display window provides a means of assigning names to the displayed variables via a popup menu. When you right-click on a variable you can select the option to edit a variable's name.

The length of the variable names can be between 2 and 29 characters. The maximum length should be set in the machine configuration to match your control. Once the maximum length is set, attempting to set a variable name which is greater than this length will result in an error.

When used in a program, named variables must still be preceded by the pound (#) symbol.

Here is an example of using named variables in a program:

```
N410 WHILE [[#SCOUNT]LE[FIX[#STEPS]]] DO 2
N420 #CRAD=[FIX[#STEPS]-#SCOUNT]
N430 IF [[#CRAD*#17]GT#18-[#20/2]] GOTO 720 (NO RADIUS)
```

As you can see, using names instead of numbers can improve the readability of a program.

Macro Examples

Macro programming is very flexible. The examples given here are not necessarily the right way or the only way to do something. This section is simply to help you understand how the macro language works.

Example 1:
This example clears variables #100 through #199 to <empty>.

```
#1 = 100                   (Assign the value 100 to variable #1)
N1 ##1 = #0                (The variable pointed to by variable #1 is cleared to
<empty>)
#1 = #1 + 1                (Variable #1 is incremented by 1)
IF [#1 LT 200] GOTO 1      (This jump is taken as long as variable #1 is less
than 200)
```

Example 2:
Here is another way to do the same thing using a WHILE – DO loop.

```
#1 = 100
WHILE [#1 LT 200] DO 1
##1 = #0
#1 = #1 +1
END1
```

Variable Maps

Mill Variables

This map details the variables recognized by NCPlot. While most controls that can be programmed in Macro B format will conform to this layout, you should check your controls documentation to be sure.

#0 Always <empty>

#1-#99 Local variables – Note that on some controls this is #1-#33

Variable	Address	Variable	Address
#1	A	#14	N x
#2	B	#15	O x
#3	C	#16	P x
#7	D	#17	Q
#8	E	#18	R
#9	F	#19	S
#10	G x	#20	T
#11	H	#21	U
#4	I	#22	V
#5	J	#23	W
#6	K	#24	X
#12	L x	#25	Y
#13	M	#26	Z

#100 - #999 Common variables

The number and range of common variables will depend on your control. On many controls, adding more common variables is an extra cost option.

#1000 - ? System variables

System variables are used by the CNC and should only be changed by your macro program with great care. The actual range of variables depends on your particular control, but the variables recognized by NCPlot are described here.

#3000 Macro Alarm Message

Assigning a value to this variable causes NCPlot to display a macro message. Because this is an alarm message, this command also terminates the program. This command is in the format:

```
#3000=100(MACRO MESSAGE)
```

#3006 Macro Message

Assigning a value to this variable causes NCPlot to display a macro message. Unlike variable #3000, this command does not terminate the program. Program execution continues after acknowledging the message. This command is in the format:

```
#3006=100(MACRO MESSAGE)
```

#3009 User Prompt Message

This variable allows your programs to prompt the user for input during execution. This is a special variable in NCPlot and is not based on an actual function of a CNC control. Assigning a value to this variable causes NCPlot to display a user input box and allows the user to enter a value. The command format is:

```
#3009=100(ENTER POCKET DIAMETER)
```

The comment text is displayed in the input box along with the current contents of the specified variable number. In this example, the variable number is #100. A new value entered into this box is then saved to this variable.

#4001 - #4020 G Modals (Pre-Read Block)

#4201 - #4220 G Modals (Execution Block)

The group 0 codes are non-modal and do not appear in the system variables. They are included here for the sake of completeness. NCPlot does not support all of the G-Codes listed here, but will store them in the proper group when encountered in a program.

	Group 0	G4, G9, G10, G11, G12, G13, G28, G34, G35, G36, G37.1, G52, G53, G65, G92
#4201	Group 1	G0, G1, G2, G3
#4202	Group 2	G17, G18, G19
#4203	Group 3	G90, G91
#4205	Group 5	G94, G95
#4206	Group 6	G20, G21
#4207	Group 7	G40, G41, G42
#4208	Group 8	G43, G44, G49
#4209	Group 9	G70 – G79, G81 – G89, G80
#4210	Group 10	G98, G99
#4211	Group 11	G50, G51
#4212	Group 12	G54 – G59, G54.1
#4214	Group 14	G66, G67
#4216	Group 16	G68, G69
#4218	Group 18	G15, G16
#4219	Group 19	G50.1, G51.1

#4301-#4320 Other Modals

#4301	A	#4311	H

#4302	B	#4312	L
#4303	C	#4313	M
#4304	I	#4314	N
#4305	J	#4315	O
#4306	K	#4316	P
#4307	D	#4317	Q
#4308	E	#4318	R
#4309	F	#4319	S
#4310	G	#4320	T

Axis positions

	X Axis	Y Axis	Z Axis	4th Axis
Last block endpoint	#5001	#5002	#5003	#5004
Machine coordinate	#5021	#5022	#5023	#5024
Work coordinate	#5041	#5042	#5043	#5044

Work Offsets

	X Axis	Y Axis	Z Axis	4th Axis
G54	#5221	#5222	#5223	#5224
G55	#5241	#5242	#5243	#5244
G56	#5261	#5262	#5263	#5264
G57	#5281	#5282	#5283	#5284
G58	#5301	#5302	#5303	#5304
G59	#5321	#5322	#5323	#5324

The extended work offsets share values between the variable ranges listed here. This means that when a value is entered into #7001 the same value is also written to #14001 and #70001. This makes the offset values available at any of the three variable ranges.

Extended Work Offsets – 48 Offsets

	X Axis	Y Axis	Z Axis	4th Axis
G54.1 P1	#7001	#7002	#7003	#7004
G54.1 P2	#7021	#7022	#7023	#7024

G54.1 P3	#7041	#7042	#7043	#7044
...				
G54.1 P46	#7901	#7902	#7903	#7904
G54.1 P47	#7921	#7922	#7923	#7924
G54.1 P48	#7941	#7942	#7943	#7944

Extended Work Offsets – 300 Offsets

	X Axis	Y Axis	Z Axis	4th Axis
G54.1 P1	#14001	#14002	#14003	#14004
G54.1 P2	#14021	#14022	#14023	#14024
G54.1 P3	#14041	#14042	#14043	#14044
...				
G54.1 P298	#19941	#19942	#19943	#19944
G54.1 P299	#19961	#19962	#19963	#19964
G54.1 P300	#19981	#19982	#19983	#19984

Extended Work Offsets – 300 Offsets

	X Axis	Y Axis	Z Axis	4th Axis
G54.1 P1	#70001	#70002	#70003	#70004
G54.1 P2	#70021	#70022	#70023	#70024
G54.1 P3	#70041	#70042	#70043	#70044
...				
G54.1 P298	#75941	#75942	#75943	#75944
G54.1 P299	#75961	#75962	#75963	#75964
G54.1 P300	#75981	#75982	#75983	#75984

NCPlot uses system variables to pass some additional information to the canned cycles. These variables are specific to NCPlot.

#5100 Machine Type
0 = Mill
1 = Lathe Radius
2 = Lathe Diameter

#5101 Canned Cycle Absolute Depth
The absolute position of the commanded canned cycle depth.

#5102 Canned Cycle Absolute R plane
The absolute position of the commanded canned cycle "R" plane.

#5103 Canned Cycle Absolute Initial point
The absolute position of the commanded canned cycle initial point.

#5110 4th axis address assignment
This variable will contain a value that indicates the letter address assigned to the 4th axis. The possible values are 1 (A), 2 (B), 3 (C), 21 (U), 22 (V) and 23 (W).

Lathe Variables - Format A

This map details the variables recognized by NCPlot. While most controls that can be programmed in Macro B format will conform to this layout, you should check your controls documentation to be sure.

#0 Always <empty>

#1-#99 Local variables – Note that on some controls this is #1-#33

Variable	Address	Variable	Address
#1	A	#14	N x
#2	B	#15	O x
#3	C	#16	P x
#7	D	#17	Q
#8	E	#18	R
#9	F	#19	S
#10	G x	#20	T
#11	H	#21	U
#4	I	#22	V
#5	J	#23	W
#6	K	#24	X
#12	L x	#25	Y
#13	M	#26	Z

#100 - #999 Common variables

The number and range of common variables will depend on your control. On many controls, adding more common variables is an extra cost option.

#1000 - ? System variables

System variables are used by the CNC and should only be changed by your macro program with great care. The actual range of variables depends on your particular control, but the variables recognized by NCPlot are described here.

#3000 Macro Alarm Message

Assigning a value to this variable causes NCPlot to display a macro message. Because this is an alarm message, this command also terminates the program. This command is in the format:

```
#3000=100(MACRO MESSAGE)
```

#3006 Macro Message

Assigning a value to this variable causes NCPlot to display a macro message. Unlike variable #3000, this command does not terminate the program. Program execution continues after acknowledging the message. This command is in the format:

```
#3006=100(MACRO MESSAGE)
```

#3009 User Prompt Message

This variable allows your programs to prompt the user for input during execution. This is a special variable in NCPlot and is not based on an actual function of a CNC control. Assigning a value to this variable causes NCPlot to display a user input box and allows the user to enter a value. The command format is:

```
#3009=100(ENTER POCKET DIAMETER)
```

The comment text is displayed in the input box along with the current contents of the specified variable number. In this example, the variable number is #100. A new value entered into this box is then saved to this variable.

#4001 - #4020 G Modals (Pre-Read Block)

#4201 - #4220 G Modals (Execution Block)

The group 0 codes are non-modal and do not appear in the system variables. They are included here for the sake of completeness. NCPlot does not support all of the G-Codes listed here, but will store them in the proper group when encountered in a program.

	Group 0	G4, G9, G10, G11, G28, G50, G52, G53, G65
#4201	Group 1	G0, G1, G2, G3
#4205	Group 5	G98, G99
#4206	Group 6	G20, G21
#4207	Group 7	G40, G41, G42
#4209	Group 9	G70 – G79, G81 – G89, G80, G90, G92, G94

#4212	Group 12	G54 – G59, G54.1
#4214	Group 14	G66, G67
#4217	Group 17	G96, G97

#4301-#4320 Other Modals

#4301	A	#4311	H
#4302	B	#4312	L
#4303	C	#4313	M
#4304	I	#4314	N
#4305	J	#4315	O
#4306	K	#4316	P
#4307	D	#4317	Q
#4308	E	#4318	R
#4309	F	#4319	S
#4310	G	#4320	T

Axis positions

	X Axis	Z Axis
Last block endpoint	#5001	#5002
Machine coordinate	#5021	#5022
Work coordinate	#5041	#5042

Work Offsets

	X Axis	Z Axis
G54	#5221	#5222
G55	#5241	#5242
G56	#5261	#5262
G57	#5281	#5282
G58	#5301	#5302
G59	#5321	#5322

The extended work offsets share values between the variable ranges listed here. This means that when a value is entered into #7001 the same value is also written to #14001 and #70001. This makes the offset values available at any of the three variable ranges.

Extended Work Offsets – 48 Offsets

	X Axis	Z Axis
G54.1 P1	#7001	#7002
G54.1 P2	#7021	#7022
G54.1 P3	#7041	#7042
...		
G54.1 P46	#7901	#7902
G54.1 P47	#7921	#7922
G54.1 P48	#7941	#7942

Extended Work Offsets – 300 Offsets

	X Axis	Z Axis
G54.1 P1	#14001	#14002
G54.1 P2	#14021	#14022
G54.1 P3	#14041	#14042
...		
G54.1 P298	#19941	#19942
G54.1 P299	#19961	#19962
G54.1 P300	#19981	#19982

Extended Work Offsets – 300 Offsets

	X Axis	Z Axis
G54.1 P1	#70001	#70002
G54.1 P2	#70021	#70022
G54.1 P3	#70041	#70042
...		
G54.1 P298	#75941	#75942
G54.1 P299	#75961	#75962
G54.1 P300	#75981	#75982

NCPlot uses system variables to pass some additional information to the canned cycles. These variables are specific to NCPlot.

#5100 Machine Type

0 = Mill
1 = Lathe Radius
2 = Lathe Diameter

#5101 Canned Cycle Absolute Depth
The absolute position of the commanded canned cycle depth.

#5102 Canned Cycle Absolute R plane
The absolute position of the commanded canned cycle "R" plane.

#5103 Canned Cycle Absolute Initial point
The absolute position of the commanded canned cycle initial point.

#5110 4th axis address assignment
This variable will contain a value that indicates the letter address assigned to the 4th axis. The possible values are 1 (A), 2 (B), 3 (C), 21 (U), 22 (V) and 23 (W).

Lathe Variables - Format B

This map details the variables recognized by NCPlot. While most controls that can be programmed in Macro B format will conform to this layout, you should check your controls documentation to be sure.

#0	Always <empty>
#1-#99	Local variables – Note that on some controls this is #1-#33

Variable	Address	Variable	Address
#1	A	#14	N x
#2	B	#15	O x
#3	C	#16	P x
#7	D	#17	Q
#8	E	#18	R
#9	F	#19	S
#10	G x	#20	T
#11	H	#21	U
#4	I	#22	V
#5	J	#23	W
#6	K	#24	X
#12	L x	#25	Y
#13	M	#26	Z

#100 - #999 Common variables

The number and range of common variables will depend on your control. On many controls, adding more common variables is an extra cost option.

#1000 - ? System variables

System variables are used by the CNC and should only be changed by your macro program with great care. The actual range of variables depends on your particular control, but the variables recognized by NCPlot are described here.

#3000 Macro Alarm Message

Assigning a value to this variable causes NCPlot to display a macro message. Because this is an alarm message, this command also terminates the program. This command is in the format:

```
#3000=100(MACRO MESSAGE)
```

#3006 Macro Message

Assigning a value to this variable causes NCPlot to display a macro message. Unlike variable #3000, this command does not terminate the program. Program execution continues after acknowledging the message. This command is in the format:

```
#3006=100(MACRO MESSAGE)
```

#3009 User Prompt Message

This variable allows your programs to prompt the user for input during execution. This is a special variable in NCPlot and is not based on an actual function of a CNC control. Assigning a value to this variable causes NCPlot to display a user input box and allows the user to enter a value. The command format is:

```
#3009=100(ENTER POCKET DIAMETER)
```

The comment text is displayed in the input box along with the current contents of the specified variable number. In this example, the variable number is #100. A new value entered into this box is then saved to this variable.

#4001 - #4020 G Modals (Pre-Read Block)

#4201 - #4220 G Modals (Execution Block)

The group 0 codes are non-modal and do not appear in the system variables. They are included here for the sake of completeness. NCPlot does not support all of the G-Codes listed here, but will store them in the proper group when encountered in a program.

	Group 0	G4, G9, G10, G11, G28, G52, G53, G65, G92
#4201	Group 1	G0, G1, G2, G3
#4203	Group 3	G90, G91
#4205	Group 5	G94, G95
#4206	Group 6	G20, G21
#4207	Group 7	G40, G41, G42
#4209	Group 9	G70 – G79, G81 – G89, G80
#4210	Group 10	G98, G99
#4212	Group 12	G54 – G59, G54.1
#4214	Group 14	G66, G67
#4217	Group 17	G96, G97

#4301-#4320 Other Modals

#4301	A	#4311	H
#4302	B	#4312	L
#4303	C	#4313	M
#4304	I	#4314	N
#4305	J	#4315	O
#4306	K	#4316	P
#4307	D	#4317	Q
#4308	E	#4318	R
#4309	F	#4319	S
#4310	G	#4320	T

Axis positions

	X Axis	Z Axis
Last block endpoint	#5001	#5002
Machine coordinate	#5021	#5022
Work coordinate	#5041	#5042

Work Offsets

	X Axis	Z Axis
G54	#5221	#5222
G55	#5241	#5242

G56	#5261	#5262
G57	#5281	#5282
G58	#5301	#5302
G59	#5321	#5322

The extended work offsets share values between the variable ranges listed here. This means that when a value is entered into #7001 the same value is also written to #14001 and #70001. This makes the offset values available at any of the three variable ranges.

Extended Work Offsets – 48 Offsets

	X Axis	Z Axis
G54.1 P1	#7001	#7002
G54.1 P2	#7021	#7022
G54.1 P3	#7041	#7042
...		
G54.1 P46	#7901	#7902
G54.1 P47	#7921	#7922
G54.1 P48	#7941	#7942

Extended Work Offsets – 300 Offsets

	X Axis	Z Axis
G54.1 P1	#14001	#14002
G54.1 P2	#14021	#14022
G54.1 P3	#14041	#14042
...		
G54.1 P298	#19941	#19942
G54.1 P299	#19961	#19962
G54.1 P300	#19981	#19982

Extended Work Offsets – 300 Offsets

	X Axis	Z Axis
G54.1 P1	#70001	#70002
G54.1 P2	#70021	#70022
G54.1 P3	#70041	#70042
...		
G54.1 P298	#75941	#75942
G54.1 P299	#75961	#75962
G54.1 P300	#75981	#75982

NCPlot uses system variables to pass some additional information to the canned cycles. These variables are specific to NCPlot.

#5100 Machine Type
0 = Mill
1 = Lathe Radius
2 = Lathe Diameter

#5101 Canned Cycle Absolute Depth
The absolute position of the commanded canned cycle depth.

#5102 Canned Cycle Absolute R plane
The absolute position of the commanded canned cycle "R" plane.

#5103 Canned Cycle Absolute Initial point
The absolute position of the commanded canned cycle initial point.

#5110 4th axis address assignment
This variable will contain a value that indicates the letter address assigned to the 4th axis. The possible values are 1 (A), 2 (B), 3 (C), 21 (U), 22 (V) and 23 (W).

DXF Drawing File Support

Exporting as DXF Drawing Files

Any viewport graphic may be saved as a DXF drawing file. This includes the part program backplot as well as any additional entities that have been created using the Calc tools. To save a DXF file click the menu ***File / Export DXF File*** and browse for or enter a filename to save to.

In addition, the menu ***File / Export Selected as DXF File*** allows saving only the selected viewport entities to a DXF file.

The saved DXF files will have one layer for each tool, with the layers named by tool number.

Using a DXF Drawing File as the Viewport Background

A DXF drawing file may be loaded and displayed as a persistent part of the viewport graphics. This is useful for displaying machine travels, part fixtures, interference areas, reference grids, etc.

Assigning a file as the viewport background is done on the ***Machine Configuration***, see the topic Viewport Settings.

Converting DXF Drawing Files to G-Code

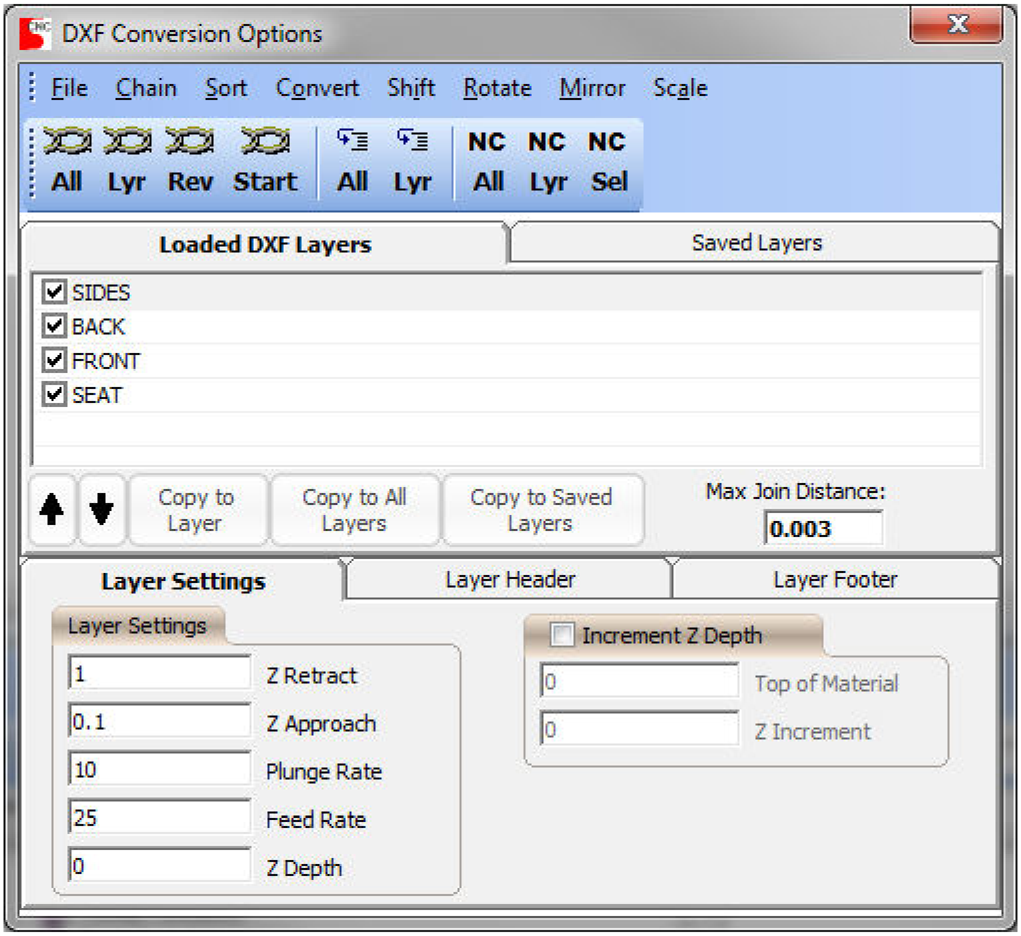
Introduction

NCPlot provides you with the capability of creating G-Code programs directly from a DXF drawing file. The process of going from a drawing to a program file requires several steps and it is important to understand each step in the process in order to get the best results. The next several topics explain in detail each step of the conversion process. Each step is outlined below:

Using the DXF Conversion Options Dialog
Arrange the Layer List
Set Machining Parameters
Chaining
Sorting
Converting to G-Code

The DXF Conversion Options Dialog

When a DXF file is loaded it is displayed on the viewport and the DXF Conversion Options dialog is displayed. Here you will define what parts of the drawing will be converted to G-Code and in what order. You will also use this dialog to specify how machining is to be done for each part of the drawing.



☑ SIDES
☑ BACK
☑ FRONT
☑ SEAT

Max Join Distance: 0.003

Layer Settings

1	Z Retract
0.1	Z Approach
10	Plunge Rate
25	Feed Rate
0	Z Depth

□ Increment Z Depth

0	Top of Material
0	Z Increment

The dialog is divided into four parts:

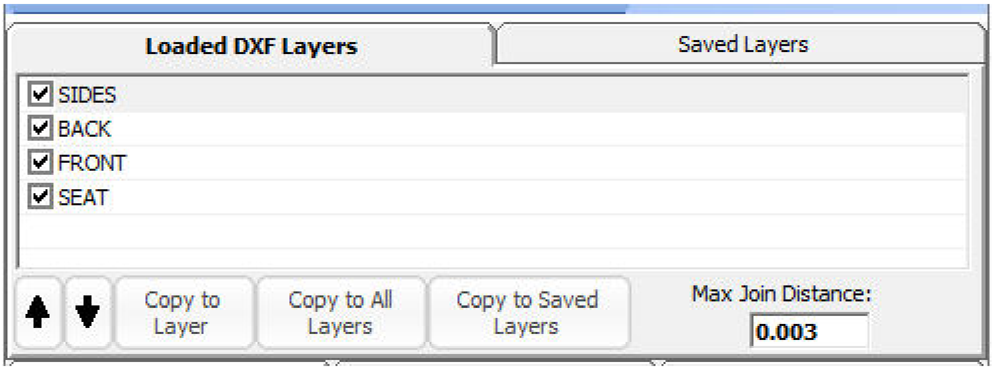
- **Menu bar** - Provides access to all the conversion features.
- **Tool bar** - Provides quick access to the chaining, sorting and conversion tools.
- **Layer List** - Displays the loaded drawing layers as well as the list of saved layers.
- **Layer Machining Settings** - The machining G-Code settings for each layer.

Before beginning the conversion process the menu provides translation tools that can be used to shift, rotate, mirror or scale your drawing.

Shift - Allows you to move the zero point of the drawing.
Rotate - Rotate the drawing by specifying a rotation center point and angle of rotation.
Mirror - Flip the drawing in either X or Y.
Scale - Enter

Arrange the Layer List

When a DXF drawing is loaded it appears on the viewport and the drawing layers appear on the DXF Conversion Options dialog.



☑ SIDES
☑ BACK
☑ FRONT
☑ SEAT

Max Join Distance: 0.003

Since a drawing may contain information that you don't necessarily want converted to G-Code such as a title block or dimensions, you can turn these layers off by unchecking the box next the layer name. When layers are turned off or on the viewport will update to display just the layers that are on.

This list also allows you to control the order in which the layers are converted to G-Code. This is especially important because it determines the order in which your part is machined. The layers will be converted in order from top to bottom, layers that are off will be skipped. To change the order, click to select the layer name then click either the up or down arrow buttons to move it up or down in the list.

Set Machining Parameters

When each layer is loaded they are initially assigned the default layer settings, which comes from the current machine configuration. The exception to this is when a loaded layer name matches one of the saved layer names. In this case the layer is assigned the saved layer settings.

The layer settings that appear on the lower half of the conversion dialog are for the currently selected layer. To select a layer, click it's name in the layer list. When a layer is selected, it's name is highlighted in the layer list and the layer settings will update to show the settings for the selected layer.

Because of the way that NCPlot creates the G-Code output, it is important to set the Z depth settings in a logical order:

- Z Retract should be the highest (most positive) value, followed by:
- Z Approach
- Top of Material
- Z Depth should be the lowest (most negative) value.

Changing one of the layer settings only affects the currently selected layer. To copy settings from one layer to another, first select the layer you want to copy then click the button Copy to Layer. This button turns green indicating that you should now click the name of layer you want to copy the settings to. To copy the same settings to all loaded layers, first select the layer you want to copy then click the button Copy to All Layers.

If your drawing has layers that you use often, you can copy them to the Saved Layer list for later use. To copy a layer to the saved layer list, first select it from the layer list the click the button Copy to Saved Layers. The layer name and all of it's settings are then copied to the saved layer list.

Chaining

Since a DXF drawing file doesn't provide the geometry data in any particular order, we need a means of identifying which parts of the drawing are connected together to form a continuous path. This is done with the chaining tool. The chaining tool will scan each layer and find all the geometry that appears to be connected together. The ***Max Join Distance*** setting determines how close the endpoints of two entities must be in order to be considered joined. This lets NCPlot create more efficient G-Code without a lot of seemingly random cutting. Since the converter is layer based, the chaining tool will only join geometry that is on the same layer.

There are two chaining tools: ***Chain All***, which will chain all the layers in the drawing and ***Chain Layer***, which will chain only the currently selected layer.

Once chained the drawing will display markers to indicate the points where the tool will enter the material as well as arrows to indicate the cutting direction. An additional benefit to the chaining tool is that it allows reversing the direction of chained geometry. To reverse the chain direction, click the ***Chain Reverse*** toolbar button. The chains belonging to any selected entities will be affected.

Chains that form a continuous path are considered closed and may have their start point at either endpoint of any of the entities that make up the path. The start point is the point where the tool enters the material before cutting the path profile. A new start point may be selected by clicking the tool ***Chain Start*** and then clicking the endpoint that you would like to be the start point of the path.

Note that chaining is required for layers that have the ***Increment Z Depth*** setting enabled.

Sorting

The sorting tool provides an additional means of optimizing the G-Code output by attempting to arrange the drawing in a way that will result in less rapid motion between parts of the drawing. It does this by starting at one corner and finding the closest part of the drawing. The next closest part of the drawing is found next and so on. This tool works with chained geometry, so the chaining tool must be applied before this tool can be used.

Converting to G-Code

There are three conversion tools, giving different levels of control over the order that the drawing is converted in. The Convert All tool will convert the entire drawing to G-Code in the order that the layers are listed. The Convert Layer tool will convert only the selected layer to G-Code. The third tool, Convert Selected, will convert only the chains belonging to any selected entities on the viewport. This gives the most control over the conversion process, but only works with chained geometry.

Customizing NCPlot

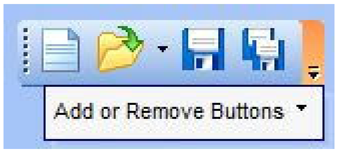
Customizing the Toolbars

The toolbars in NCPlot are highly customizable. You can hide individual buttons, hide entire toolbars or even create your own toolbars containing your most commonly used buttons or even menu shortcuts.

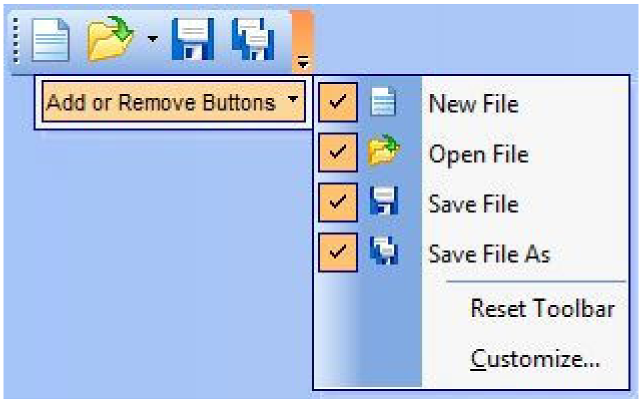
The toolbar layout, including custom toolbars is saved when you exit NCPlot. If you want to reset your toolbars to the default layout click the menu ***File / Reset Toolbars.***

Hiding toolbar buttons

To customize a toolbar, it must first be docked. Each toolbar has a band on its right side that may be clicked to display the option to Add or Remove Buttons.

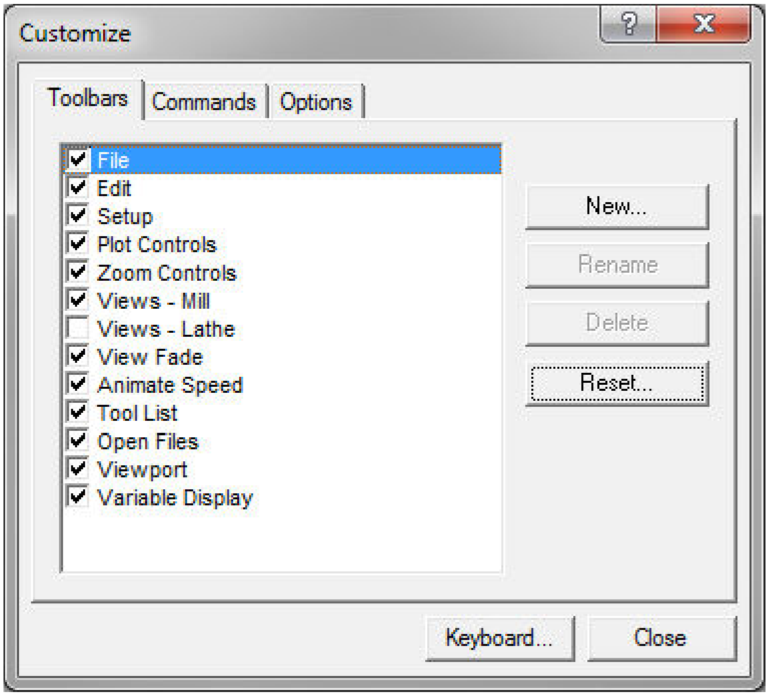


Hovering the mouse pointer over Add or Remove Buttons will then display a list of the buttons on the toolbar along with check boxes for each button. Unchecked boxes will hide the corresponding button.



Create custom toolbars

You may also create your own custom toolbars that can contain any combination of buttons from the other toolbars. The toolbar customize dialog can be accessed by right-clicking any of the toolbars and selecting ***Customize*** from the menu that appears.

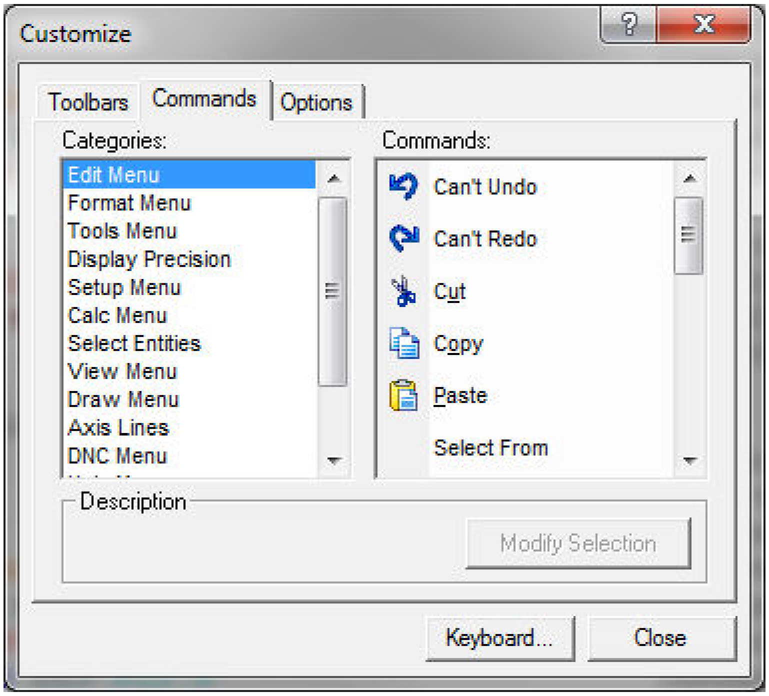


☑ File
☑ Edit
☑ Setup
☑ Plot Controls
☑ Zoom Controls
☑ Views - Mill
□ Views - Lathe
☑ View Fade
☑ Animate Speed
☑ Tool List
☑ Open Files
☑ Viewport
☑ Variable Display

On this dialog, click the ***New*** button and then enter a name for the new toolbar. The toolbar is then created and will appear on the screen as an undocked toolbar. To add buttons to the new toolbar the ***Customize*** dialog must be left open. Click and drag buttons to move them from the other toolbars to your new toolbar, or hold the ***Ctrl*** key while dragging to copy them.

Adding menu shortcuts to a toolbar

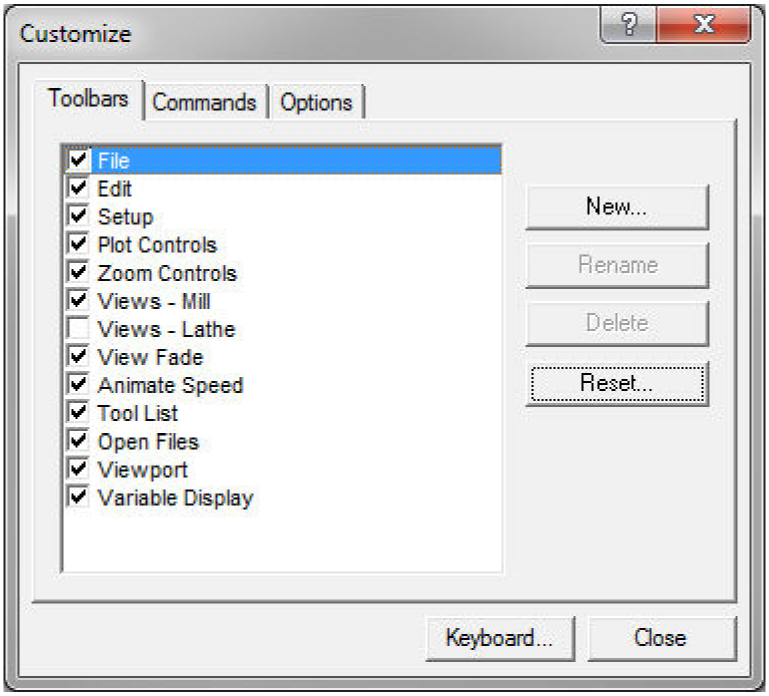
Menu commands may also be added to existing or custom toolbars.



On the toolbar customize dialog, select the ***Commands*** tab. The ***Categories*** list represents each of the menus in NCPlot and the ***Commands*** list represents the items in each menu. To add a command to a toolbar, drag it from the ***Commands*** list to an open toolbar.

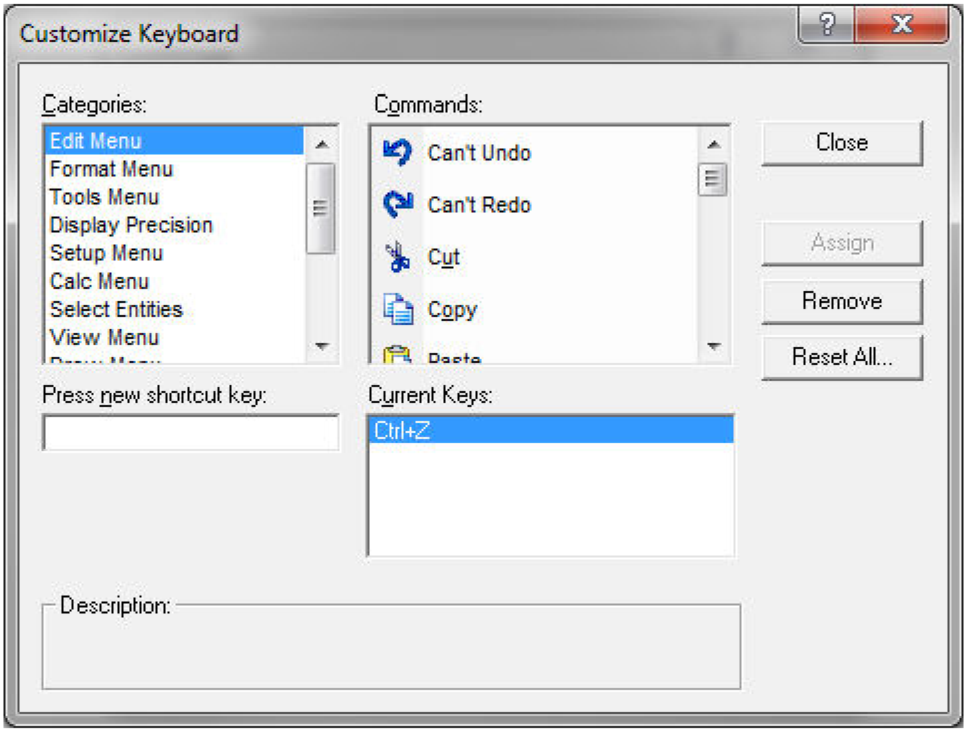
Menu Shortcut Keys

The menu shortcut keys can customized via the toolbar customize dialog. The toolbar customize dialog can be accessed by right-clicking any of the toolbars and selecting ***Customize*** from the menu that appears. The keyboard shortcut keys are part of the toolbar layout and will be saved when exiting NCPlot. Resetting the toolbar layout from the menu ***File / Reset Toolbars*** will also reset the keyboard shortcut keys. The shortcut keys may be saved via the ***Setup / Export Settings*** or loaded via ***Setup / Import Settings*** menus.



☑ File
☑ Edit
☑ Setup
☑ Plot Controls
☑ Zoom Controls
☑ Views - Mill
□ Views - Lathe
☑ View Fade
☑ Animate Speed
☑ Tool List
☑ Open Files
☑ Viewport
☑ Variable Display

On the ***Customize*** dialog, click the ***Keyboard*** button to access the ***Customize Keyboard*** dialog.



Current Keys: Ctrl+Z

Using this dialog you can assign or change any of the keyboard shortcut keys for any of the NCPlot menu items.

License Manager Support

Using the NCPlot Network License Manager

NCPlot supports the use of a network license manager. The NLM allows networked computers running NCPlot to share licenses by issuing licenses to instances of NCPlot as they start and then returning the license to the NLM as they shut down. Once all the available licenses have been issued, no more instances may be started until a license is returned to the NLM. This ensures that the number of instances of NCPlot running does not exceed the number of licenses.

The license manager uses the UDP and TCP protocols for communications with its client applications. Have your network administrator ensure that TCP port 50000 and UDP ports 50001 and 50002 are open for communications.

You can select whether you want to auto detect the presence of a license server or specify the network IP address of the server. This can be done through the ***License Manager Settings*** dialog.

The NLM is a free software application available from www.ncplot.com.

Scripting Reference

About Scripting

NCPlot supports scripts written in the VBScript language. This gives you a very powerful tool useful for automating common tasks. This manual assumes you already know how to write scripts in the VBScript language and only provides a reference for the functions made available by NCPlot.

VBScripting extends the capabilities of NCPlot by giving you, the script writer, access to many of NCPlot's internal functions.

This tool makes it possible to:

- Automate common conversion tasks, such as converting code that is written to run on one machine into code for another machine.
- Create custom code generation tools that can accept user input.
- Batch process whole file folders using the provided functions for wildcard file searching.

When used in your script, the function names listed in this manual must be preceded by the keyword 'ncplot', for example:

NCPlot.NCPFileLoad "c:\test.txt"

To execute a script file in NCPlot, go to ***File | Execute Script File***. Browse to the file you want to execute and click 'OK'. Script files may be edited with any text editor.

File Functions

NCPFileNew

NCPlot.NCPFileNew

Creates a new, blank document.

NCPFileLoad

NCPlot.NCPFileLoad strPath

Loads a file into the active document. Use With Caution!! This function does not prompt you to save current edits before loading the file. Be sure the active document file is saved before using this function.

NCPGetActiveFile

NCPlot.NCPGetActiveFile

Returns the full pathname of the active edit file. This function returns a NULL string if the active file is Untitled.

NCPSetActiveFile

NCPlot.NCPSetActiveFile strPath

Sets the document corresponding to the file strPath as the active document.

NCPFileSave

NCPlot.NCPFileSave strFileToSave, strPathToSaveAs

Saves the active or specified file.

strFileToSave - optional setting specifying the document corresponding to the given pathname. If omitted the active document is assumed.

strPathToSaveAs - optional setting specifying a pathname to save the file to. If omitted the file is saved as its current name.

NCPExportDXF

NCPlot.NCPExportDXF strPath

This function saves the current backplot graphic as a DXF file whose name is specified by strPath.

NCPSetConfig

NCPlot.NCPSetConfig cfgName

Set machine configuration. The value of cfgName should be the name of an existing configuration, “Mill” for example.

NCPBrowseForFile

NCPlot.NCPBrowseForFile

Opens the File Browse dialog for allowing the user to select a file. Returns the selected filename, with path. Returns a Null string if the user clicks Cancel on the browse dialog.

NCPBrowseForFolder

NCPlot.NCPBrowseForFolder

Opens the Folder Browse dialog for allowing the user to select a folder. Returns the selected path without the ending “\” delimiter. A Null string is returned if the user clicks ‘Cancel’ on the browse dialog.

NCPGetFirstMatchingFile

NCPlot.NCPGetFirstMatchingFile strPath

Returns the first matching directory entry that matches a given path string, including wildcard characters. This function returns the filename of the first matching entry, or a Null string if no matches were found.

NCPGetNextMatchingFile

NCPlot.NCPGetNextMatchingFile

Returns the next matching directory entry. When the function NCPGetFirstMatchingFile is used to start a wildcard file search, this function will return subsequent matches or a Null string when no more matches are found.

NCPGetFileCount

NCPlot.NCPGetFileCount

This function returns a value indicating how many documents are open.

NCPGetFilePath

NCPlot.NCPGetFilePath intIndex

Returns the pathname of the requested document number.

intIndex - the requested document number from 1 to the number of open documents.

NCPCloseFile

NCPlot.NCPCloseFile strPath

Closes the document corresponding to the given pathname.

strPath - optional setting corresponding to the requested pathname to close. If omitted the active document is closed.

NCPCloseAllFiles

NCPlot.NCPCloseAllFiles

Closes all open files.

NCPShell

NCPlot.NCPShell strShellCMD

This function allows a script to launch external applications. The operand should be the path/name of the executable file including any command line parameters.

Example:

```
dim strFilePath
dim strCMD

' Save the active file
NCPlot.NCPFileSave

' Get the path to the active file
strFilePath = NCPlot.NCPGetActiveFile

' Build the shell command
' Note that the passed path/file name must be surrounded by double quotes if it contains spaces
strCMD = "notepad.exe " & chr(34) & strFilePath & chr(34)

' Execute shell command
NCPlot.NCPShell strCMD
```

Edit Functions

NCPSelectLines

NCPlot.NCPSelectLines lngStart, lngEnd

Select range of program lines. Selects from line lngStart to lngEnd in the edit window. This is useful if you want to apply formatting or conversion operations to only part of a program.

NCPSetSelection

NCPlot.NCPSetSelection lngStart, lngLength

Set selection start and length. Selects text beginning at character index lStart for lLength characters.

NCPInsertText

NCPlot.NCPInsertText strText

Insert text into edit window. The given string txtString is inserted into the program at the current insertion point. The insertion point can be changed with NCPSetSelection by setting lLength to 0. If any text is currently selected, txtString will replace it.

NCPInsertLine

NCPlot.NCPInsertLine strText

Insert line into edit window. Same as NCPInsertText except this function also adds a carriage return / line feed to the end of the line.

NCPGetNumLines

NCPlot.NCPGetNumLines

Returns the number of lines in the program.

NCPGetLine

NCPlot.NCPGetLine lngNumber

Returns the contents of the requested program line number.

NCPGetLineIndex

NCPlot.NCPGetLineIndex lngNumber

Returns a pointer to the beginning of the requested line number.

NCPGetLineNumber

NCPlot.NCPGetLineNumber lngIndex

This function returns the line number of the file position pointed to by lngIndex. Useful for getting the line number from NCPFind results.

NCPGetCursorPos

NCPlot.NCPGetCursorPos

Returns a value indicating the current cursor position in the file.

NCPSetCursorPos

NCPlot.NCPSetCursorPos lngStart

Moves the cursor to the specified location in the file.

NCPSelectAll

NCPlot.NCPSelectAll

Selects the entire contents of the edit window.

NCPGetAll

NCPlot.NCPGetAll

This function returns the entire contents of the edit window as a string.

NCPGetSelected

NCPlot.NCPGetSelected

This function returns the selected portion of the edit window as a string.

NCPReplaceAll

NCPlot.NCPReplaceAll str1, str2

This function finds all occurrences of the string str1 and replaces them with the string str2.

NCPFind

NCPlot.NCPFind strFind, lngStart, Options

This function searches for the requested text in the active file and returns a pointer to the first matching string or -1 if no matches are found.

strFind - is the text to search for.

lngStart - is optional and specifies a position in the file to start searching from.

Options - is optional and is the sum of:
2 = find whole word only
4 = match case
8 = don't highlight found text

Format Functions

NCPRenumber

NCPlot.NCPRenumber StartBlock, BlockIncrement, MaxDigits, Rstyle, AddSpace

Renumbers the currently loaded file.

StartBlock	The starting block number.
BlockIncrement	Block number increment.
MaxDigits	Maximum number of digits in block number.
RStyle	Renumber style 0 = Remove block numbers 1 = Renumber all blocks 2 = Renumber all but blank and comment blocks 3 = Renumber referenced blocks only 4 = Renumber existing block numbers only
AddSpace	0 = No space after block number 1 = Add space after block number

NCPColorize

NCPlot.NCPColorize

Applies the address color settings to the loaded program.

NCPAddSpaces

NCPlot.NCPAddSpaces

Equivalent to the Add Spaces tool.

NCPRemoveSpaces

NCPlot.NCPRemoveSpaces

Equivalent to the Remove Spaces tool.

NCPRemoveLeadingSpaces

NCPlot.NCPRemoveLeadingSpaces

Equivalent to the Remove Leading Spaces Tool.

NCPRemoveTrailingSpaces

NCPlot.NCPRemoveTrailingSpaces

Equivalent to the Remove Trailing Spaces tool.

NCPRemoveBlankLines

NCPlot.NCPRemoveBlankLines

Equivalent to the Remove Blank Lines tool.

NCPRemoveComments

NCPlot.NCPRemoveComments

Equivalent to the Remove Comments tool.

NCPRemoveGarbage

NCPlot.NCPRemoveGarbage

Equivalent to the Remove Garbage tool.

NCPAllCaps

NCPlot.NCPAllCaps

Equivalent to the Convert To All Caps tool.

NCPRemoveRedundant

NCPlot.NCPRemoveRedundant

Equivalent to the Remove Redundant Endpoints tool.

Tools Functions

NCPConvertText

NCPlot.NCPConvertText

Invokes the Text To G-Code tool using the current settings. See the topic NCPSetTTGSetting for setting the Text To G-Code settings.

NCPMirror

NCPlot.NCPMirror Xpoint, Ypoint, Zpoint

Mirror axis endpoints. The MirrorPoint values specify the center point for mirroring on that axis. If a value is omitted, no mirroring is performed on that axis.

NCPRotate

NCPlot.NCPRotate xCenter, yCenter, zCenter, Angle, [Plane]

Rotate program coordinates. Specify X rotation center, Y rotation center, Z rotation center, rotation angle and optionally the rotation plane. Plane should be either 17 (XY), 18 (ZX) or 19 (YZ). If plane is not given or is any value other than 17, 18 or 19 then this function defaults to the XY plane.

NCPShift

NCPlot.NCPShift xShift, yShift, zShift

Shift axis coordinate values. The specified values are added to the programmed coordinates.

NCPScale

NCPlot.NCPScale xScalePoint, yScalePoint, ScaleFactor

Scale axis coordinate values. The ScalePoint values are the X, Y center point for scaling.

NCPAddressAdjust

NCPlot.NCPAddressAdjust adjAddrList, adjOp, adjValue, adjFormat

Apply address value adjustments and formatting.

adjAddrList String that contains the list of addresses to adjust. For example, the string “XYZ” will apply the specified adjustment to all X, Y and Z coordinates.

AdjOp Value that specifies the type of adjustment:
0 = add, 1 = subtract, 2 = multiply, 3 = divide.

adjValue The adjustment value.

adjFormat Formatting string to be applied to the result of the adjustment.

NCPAddressReplace

NCPlot.NCPAddressReplace strFind, strReplace

Replaces address identifiers.

strFind String that contains the address character to find.

strReplace String that contains the address character to replace it with.

NCPAddressRemove

NCPlot.NCPAddressRemove strFind

Removes address identifiers and their values.

strFind String that contains the addresses to remove.

NCPAddressSwap

NCPlot.NCPAddressSwap str1, str2

This function calls the address swap tool. The string values str1 and str2 should each contain a single address character “A” to “Z” excluding the address “G”. This tool will scan the program, swapping these two address values.

NCPConvertArcsToAbsolute

NCPlot.NCPConvertArcsToAbsolute

Equivalent to the Convert arc centers to absolute tool.

NCPConvertArcsToIncremental

NCPlot.NCPConvertArcsToIncremental

Equivalent to the Convert arc centers to incremental tool.

NCPConvertArcsToIJK

NCPlot.NCPConvertArcsToIJK

Equivalent to the Convert arc R to IJK tool.

NCPConvertArcsToR

NCPlot.NCPConvertArcsToR

Equivalent to the Convert arc IJK to R tool.

NCPConvertCoordToABS

NCPlot.NCPConvertCoordToABS

Equivalent to the Convert coordinates to ABS tool.

NCPConvertCoordToINC

NCPlot.NCPConvertCoordToINC

Equivalent to the Convert coordinates to INC tool.

NCPBreakLines

NCPlot.NCPBreakLines SegmentLength

This function is equivalent to the ***Tools / Break Lines Into Segments*** tool. The specified segment length determines how many smaller moves are created.

NCPBreakArcsIntoLines

NCPlot.NCPBreakArcsIntoLines MaxDeviation, [MaxLength]

This function is equivalent to the ***Tools / Break Arcs Into Lines*** tool. The specified maximum deviation amount determines how many line segments are created. The optional **MaxLength** parameter will break only arcs smaller than this setting. If omitted or less than or equal to zero then all arcs will be converted.

NCPBreakArcsAtQuadrants

NCPlot.NCPBreakArcsAtQuadrants

This function is equivalent to the ***Tools / Break Arcs At Quadrants*** tool.

NCPGetRunTime

NCPlot.NCPGetRunTime FeedLength, FeedTime, RapidLength, RapidTime, ToolChanges, ToolChangeTime

This function returns the run time estimation values for the active backplot.

FeedLength - returns the total feed length
FeedTime - returns the total feed time in seconds
RapidLength - returns the total rapid length
RapidTime - returns the total rapid time in seconds
ToolChanges - returns the total number of tool changes
ToolChangeTime - returns the total tool change time

NCPFormatTime

NCPlot.NCPFormatTime Seconds, noDHMS

This function returns a formatted string representing the given number of seconds.

Seconds - is the number of seconds to convert

noDHMS - is an optional True/False setting for the type of result:

False = returns a string in the format "00d:00h:00m:00s"
True = returns a string in the format "00:00:00:00"

NCPMsgWindow

NCPlot.NCPMsgWindow Message, [Seconds]

This sub displays a message window.

Message - is the text string to display as the message.

Seconds - is an optional number of seconds to display the message. When specified, the message window will close automatically after this time has expired.

A new message may be displayed without first closing an active message. If a message is being displayed when script execution ends it will be closed automatically.

NCPMsgClose

NCPlot.NCPMsgClose

If a message window is currently being displayed then this command will close it.

Setup Functions

NCPGetGeneralSetting

NCPlot.NCPGetGeneralSetting strSetting

Returns the current value of the requested general application setting. Valid setting names are:

AutoRefresh
ShowAxisLines
ShowRapid
ShowTicks
ShowMarker
ShowPlunge
LockVerticalRotate
BlockSkip
Colorize

NCPSetGeneralSetting

NCPlot.NCPSetGeneralSetting strSetting, newValue

Sets the requested general application setting to newValue. See topic NCPGetGeneralSetting for a list of the valid setting names.

NCPGetPrefSetting

NCPlot.NCPGetPrefSetting strSetting

Returns the current value of the requested Preferences setting. Valid setting names are:

AllowMultipleInstances
WarningFeedrateZero
WarningSpindleOff
WarningSpindleZero
WarningReset
AlwaysCaps
AutoArrangeFiles
OpenNewFileOnStartup
BackgroundColor
AnimateDelay
AutoRefreshDelay
Steps
AutoShowEntityInfo
RightClickMenu
PlotFileWhenLoaded
ZoomOnViewChange
ZoomAfterTranslate
StopAtM00
AnimateTCPause
ReverseMouseWheelZoom
SaveVariablesOnExit
AllowMacroExpressionUpdating
ScriptTimeout

NCPSetPrefSetting

NCPlot.NCPSetPrefSetting strSetting, newValue

Sets the requested Preferences setting to newValue. See topic NCPGetPrefSetting for a list of the valid setting names.

NCPGetSubprogramSetting

NCPlot.NCPGetSubprogramSetting strSetting

Returns the current value of the requested Subprogram setting. Valid setting names are:

M98Handling
G65Handling
DefaultPath

NCPSetSubprogramSetting

NCPlot.NCPSetSubprogramSetting strSetting, newValue

Sets the requested subprogram setting to newValue. See topic NCPGetSubprogramSetting for a list of the valid setting names.

NCPSubListAdd

NCPlot.NCPSubListAdd lngProgramNumber, strPath

Adds a new subprogram association to the subprogram association list.

lngProgramNumber - is the program number to associate with an external file.

strPath - is the pathname of the subprogram file.

NCPSubListRemove

NCPlot.NCPSubListRemove lngProgramNumber

Removes a subprogram association from the subprogram association list.

lngProgramNumber - The program number of the association to remove.

NCPSubListRemoveAll

NCPlot.NCPSubListRemoveAll

Removes all subprogram associations from the subprogram association list.

NCPGetSubListCount

NCPlot.NCPGetSubListCount

Returns the number of items in the subprogram association list.

NCPGetSubListItem

NCPlot.NCPGetSubListItem intIndex, lngProgramNumber, strPath

Returns the program number and pathname of the requested item number in the subprogram association list.

intIndex - The specified item number from 1 to the number of items in the subprogram association list.

lngProgramNumber - The returned program number of the requested item.

strPath - The returned pathname of the requested item.

NCPGetColorSetting

NCPlot.NCPGetColorSetting intAddress

Returns the value of the requested address color.

intAddress - value from 1 to 30 indicating the requested address.

NCPSetColorSetting

NCPlot.NCPSetColorSetting intAddress, newValue

Sets the value of the requested address color.

intAddress - value from 1 to 30 indicating the requested address.

newValue - the new color value.

Color values are calculated by the formula:

(BLUE * 65536) + (GREEN * 256) + RED

Where the RED, GREEN and BLUE are color intensity values between 0 and 255.

NCPGetFontSetting

NCPlot.NCPGetFontSetting strSetting

Returns the current value of the requested Font setting. Valid setting names are:

Name
Size
Bold
Italic
Color

NCPSetFontSetting

NCPlot.NCPSetFontSetting strSetting, newValue

Sets the requested Font setting to newValue. See topic NCPGetFontSetting for a list of the valid setting names.

NCPGetTTGSetting

NCPlot.NCPGetTTGSetting strSetting

Returns the current value of the requested Text To G-Code setting. Valid setting names are:

FontName
FontBold
FontItalic
Text
Height
XLocation
XCenter
YLocation
YCenter
Angle
Justify
OnArc
ArcCCW
ArcRadius
StartAngle
ZRetract
ZApproach
ZDepth
Feedrate

NCPSetTTGSetting

NCPlot.NCPSetTTGSetting strSetting, newValue

Sets the requested Text To G-Code setting to newValue. See topic NCPGetTTGSetting for a list of the valid setting names.

Draw Functions

NCPPlot

NCPlot.NCPPlot

Refreshes the graphics viewport. Equivalent to the Refresh Plot tool.

NCPViewSetOrientation

NCPlot.NCPViewSetOrientation intOrientation

This function sets the desired viewport orientation.

intOrientation - is a value that indicates the desired orientation

(Mill)
0 = Top
1 = Bottom
2 = Front
3 = Back
4 = Right
5 = Left
6 = Isometric Vertical
7 = Isometric Horizontal

(Lathe)
8 = Front Turret
9 = Back Turret
10 = Vertical Left
11 = Vertical Right

NCPViewZoomExtents

NCPlot.NCPViewZoomExtents

This function is equivalent to the Zoom Extents tool.

NCPViewZoomAll

NCPlot.NCPViewZoomAll

This function is equivalent to the Zoom All tool.

NCPPickPoint

NCPlot.NCPPickPoint X, Y, Z, strMessage

This function pauses script execution while it waits for the user to click a point on the viewport.

X - returns the X coordinate of the clicked point
Y - returns the Y coordinate of the clicked point
Z - returns the Z coordinate of the clicked point
strMessage - is a message to display in the status bar

NCPGetExtents

NCPlot.NCPGetExtents Xp, Xm, Yp, Ym, Zp, Zm

This function returns the axis extents for the active backplot.

Xp - returns the X axis plus extent
Xm - returns the X axis minus extent
Yp - returns the Y axis plus extent
Ym - returns the Y axis minus extent
Zp - returns the Z axis plus extent
Zm - returns the Z axis minus extent

Macro Functions

NCPEvaluate

NCPlot.NCPEvaluate strExpression, isEmpty, errCode

This function evaluates an expression given by **strExpression** and returns a numeric value. If the expression evaluates to <empty> then isEmpty returns TRUE. If there is an error in the expression then **errCode** returns an integer error code.

NCPParseExpression

NCPlot.NCPParseExpression strExpression, lPointer, errCode

This function parses an expression within a string, returning the parsed expression. Parsing starts at the character index specified by **lPointe**r and is updated by this function to the character immediately after the expression.

The possible error codes are:
0 = no error
1 = Expression syntax error
2 = Grouping mismatch
3 = Missing operator
4 = Invalid variable number
5 = SQR operand is negative
6 = LN operand is negative
7 = Division by zero
8 = MOD division by zero

NCPGetVariable

NCPlot.NCPGetVariable varNum, isEmpty, [pLevel]

This function returns the current value in the specified variable number. If the variable is <empty> then **isEmpty** returns TRUE. If requesting a local variable then the optional **pLevel** specifies the program level, 0 = Main program, 1 = sub 1, etc.

NCPSetVariable

NCPlot.NCPSetVariable varNum, varVal, isEmpty, [pLevel]

This function sets the specified variable number to the value given by **varVal**. Setting **isEmpty** to TRUE will set the specified variable to <empty>. If specifying a local variable then the optional **pLevel** specifies the program level, 0 = Main program, 1 = sub 1, etc.