

# **WinCNC Controller**

---

## **WinCNC Controller 3.0**

Revision 3.9

Copyright © 1999 - 2021

Microsystems World CNC, LLC

141 Old Weston Road, Buckhannon, WV 26201

(304) 472-7206

Microsystems World CNC, LLC makes no warranties either expressed or implied, including any implied warranties of fitness for a specific purpose, for this software. This package is provided solely on an “as is” basis. All risk involved in the use of this software is the responsibility of the purchaser.

Program errors, improper software configuration, hardware failures, or other problems can result in damage to the machinery being controlled. Failure of limit switches, cabling, or port components can result in machine movement past limits, potentially damaging the machine and endangering operators. It is imperative that a separate Emergency Stop system be in place, which bypasses all computer control. Software features to abort motion from the keyboard and limit switch inputs are provided as conveniences only, and must not be relied upon to stop motion in an emergency.

Microsystems World CNC, LLC disclaims all liability for any and all damages, including incidental and consequential damages in connection with use of this software. The sole and exclusive liability of Microsystems World CNC, LLC will be limited to the purchase price of this software package.

If you do not understand and agree with these disclaimers, please do not use this product. Return all materials to the place of purchase for a full refund.

<b>3</b>	<b>Hardware &amp; Software Installation</b>
<b>11</b>	<b>Program Operation</b>
<b>57</b>	<b>System Configuration</b>
<b>94</b>	<b>Screen Configuration</b>
<b>108</b>	<b>Command Reference</b>
<b>144</b>	<b>Macro Configuration</b>
<b>146</b>	<b>Messages and Error Codes</b>
<b>152</b>	<b>Keypad Configuration</b>
<b>162</b>	<b>Appendix</b>
<b>170</b>	<b>Index</b>

## Installation

CAUTION: DO NOT USE THIS SOFTWARE ON A COMPUTER CONNECTED TO A MACHINE UNLESS THERE IS AN EMERGENCY STOP READILY AVAILABLE WHICH WILL INDEPENDENTLY STOP THE MACHINE.

Software features to abort moves from the keyboard and limit switch inputs can NOT be relied upon in emergency situations and are provided as conveniences only!

## Requirements

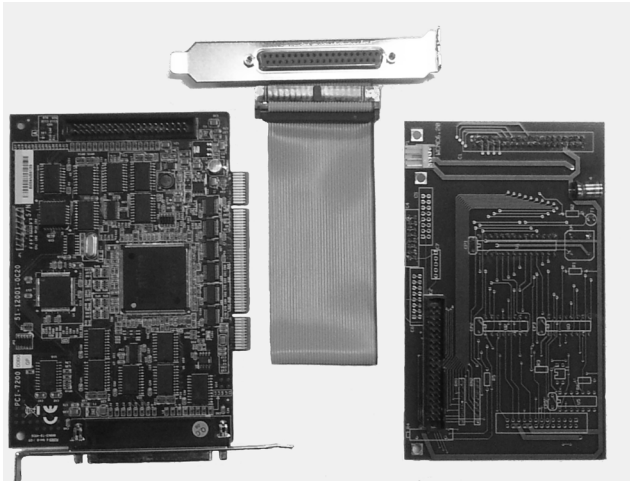
WinCNC uses PCI bus interrupts to achieve precise step pulse accuracy. When used with the right PC build WinCNC can output 100khz on all six step outputs with microsecond accuracy and 100% reliability. Fast PCI bus interrupt response is necessary to get the best performance from your CNC machine.

- CPU:** A Pentium CPU is recommended. Pentium Core2Duo, Sandy Bridge, Ivy Bridge, Haswell, Skylake, Kaby Lake, or Coffee Lake CPUs are supported. For AMD systems a socket FM2+ CPU is required.
- RAM:** 2GB of RAM is the minimum requirement, with 4GB of RAM recommended.
- Hard Drive:** Very little hard drive space is necessary. A standard WinCNC installation requires less than 100MB.
- OS:** WinCNC supports Vista, 7, 8, and 10. Both the 32 and 64 bit versions of Windows are supported.
- Case:** A MicroATX Mini or larger case is required. Hardware dimensions should be verified for a MiniITX case.
- Motherboard Expansion:** One PCI/PCIe expansion slot is required. Hardware dimensions are slightly larger than a single slot. Additional room below the PCI slot is required in some cases.
- USB Port:** An enabled USB port is required.

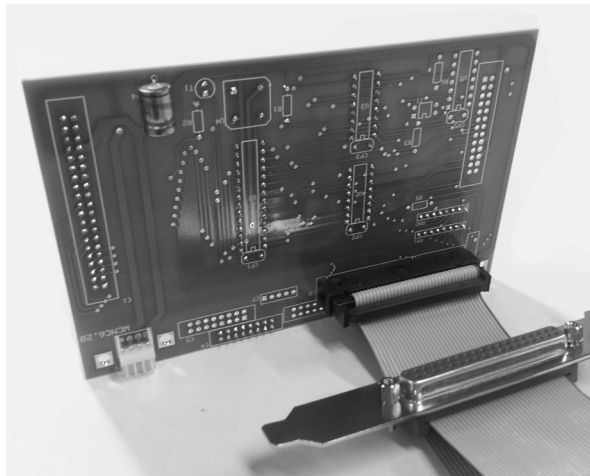
## Hardware Installation and Configuration of the PCI-7200 Card

\*\*It is highly recommended that you use an anti-static wrist strap before handling any components.

Below are the components for the PCI-7200 installation. These include a PCI-7200 card, a Daughter Board, and a ribbon cable connector.

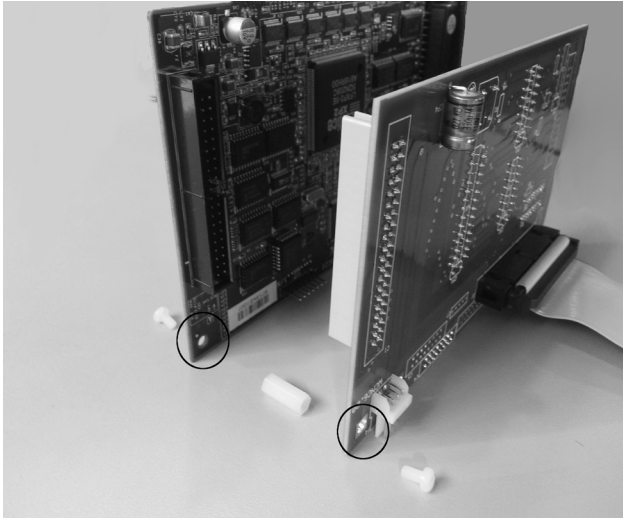


Insert the ribbon cable(s) connector(s) into the Daughter Board.



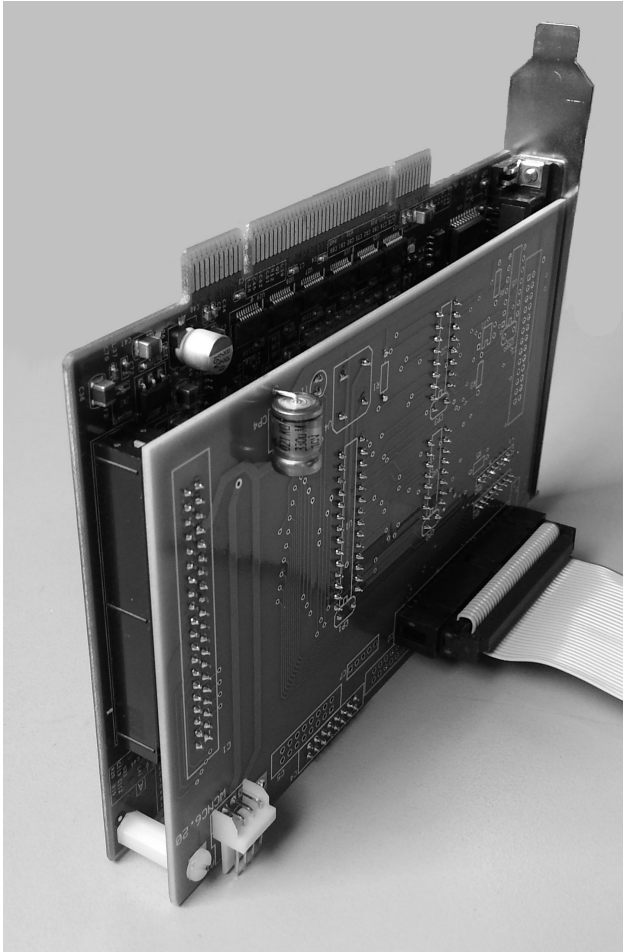


Below is the PCI-7200 card on the left, the Daughter Board on the right and the spacer/screw assembly. Note the matching holes in the corners of the Daughter Board and the PCI-7200 card. There is another set in the other end of the cards as well, just beyond the ribbon cable connector.



Note: The next step may already have been done prior to shipping.

Insert the Daughter Board onto the 40-pin connector of the PCI-7200 card. Attach the two cards with the included spacers and screws. This will ensure constant connection between the two boards.



Power down and unplug the PC from any power.

Connect the ribbon cable (or cables) into the matching header (or headers) on the Daughter Board. Now, install the PCI-7200, with the Daughter Board connected, into an available PCI slot on the PC motherboard. Next, install the DB37 or DB25 ribbon cable connectors into empty slots on the PC rear panel. Screw or fasten the PCI-7200 and DB connectors into place. Now you can connect the 4-pin PC power supply (floppy drive style) cable to the Daughter Board. This direct power supply connection is optional and may not be available in your PC.



Once installed there will be a **black** DB37 connector (CN1) available at the rear of the PC, and either one **blue** DB37 connector (CN2), or in some cases one or two **blue** DB25 connectors. The number and size of connectors available depends on the Daughter Board type that is required for your CNC machine.

**VERY IMPORTANT!** If you are unsure about where each cable from your CNC machine should connect at the back of your PC the contact technical support to verify the correct cable locations. Damage to the PCI-7200 or Daughter Board is possible if cables are connected incorrectly.

After installing your PCI-7200 card and the extra connector, boot up your computer for driver and software installation.

**\*\*Note:** When your computer boots up, you may get a hardware installation wizard. Close it. You will be installing the driver after the software is installed.

## Install software

Insert the WinCNC Controller CD-ROM into the CD-ROM drive. If the installation does not automatically start, you will need to manually run the "Setup.exe" file on the disk. To do this, click "START", then "My Computer", then right click on the "D:, E:, or F:" drive that is labeled 'WinCNC.'. You should get a menu that pops up.

On that menu, click "Explore". Once the files on the CD are visible, double click the "Setup.exe" file. Follow the prompts of the installation wizard. The install routine will create the specified directory if it does not exist and copy the necessary files into it. If there is a previous installation of WinCNC controller the files will be overwritten.

## Driver Installation for PCI-7200 Card:

**\*\*Normally, driver installation is done automatically by the installation CD.**

### **For software version 2.4.xx:**

A driver is required for the PCI-7200 card. The driver is included in the PCI-7200 folder on the root directory of the WinCNC installation disk for user convenience. If the installation disk isn't available, the driver can be downloaded from the Microsystems World CNC web site. After connecting to the internet, go to [www.wincnc.net](http://www.wincnc.net) and click on the "Resources" link at the top left of the page, then click "Downloads". Next, find the "CNC Windows/" link and click that, then click "Device Manager", click "2.4", then click "PCI7200", and finally click "PCI7200.exe". Next click "Save" and browse to your desktop, and then click "Save" again. Once the download is complete, click "Run", and then "Run" again. Make sure at this point, that you "browse" to the WinCNC folder on your C:\ drive before clicking "Unzip".

Go into the PCI7200 folder inside of the WinCNC folder and find "uninstdrv.bat". Run that file, and be careful not to close it before it is done. Watch for prompts asking you to press a key to continue. After that is done, run "instdrv.bat". Same thing.... Either or both of these may take longer than you think, and may pause and not do anything long enough for you to think it might not be working correctly. Sometimes those two files run fairly quickly, but there have been instances when they take a couple of minutes to complete.

After running both files, reboot your computer and try opening WinCNC.

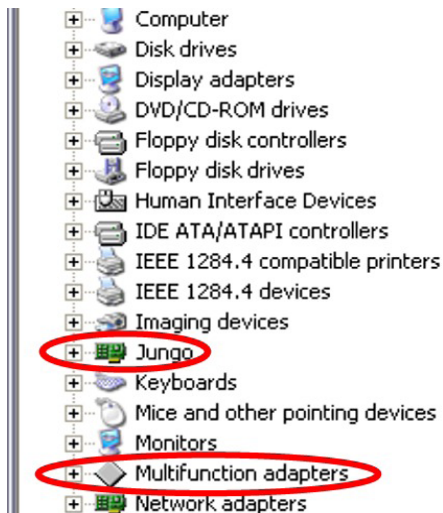
Run the program by double clicking on the WinCNC icon on your Windows desktop.

**For software version 2.5.xx, 2.6.xx, or 3.0.xx:**

In the WinCNC folder, there is a file named “driverinstaller.exe”. This should be the only file you need to run in order to install the PCI card drivers. Double click on this file to run the driver installer program which will automatically detect the operating system and install the correct device driver (32 bit or 64 bit). After the program is finished, reboot your PC, and the driver should be installed correctly.

**To verify the correct installation of the device driver, for all 2.4, 2.5, 2.6, and 3.0 versions, do the following:**

Click “Start”, right-click on “My Computer”, click “Properties”. Then find the “Hardware” tab, click it, then click the “Device Manager” button. In the Device Manager list, there should be two components as shown and circled below.

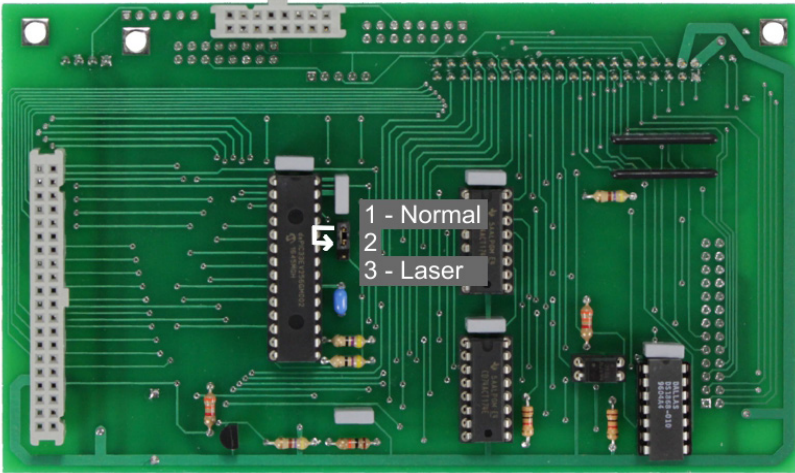


# WinCNC Controller

---

## Configuring the 37-pin Daughterboard for Laser

The 37-pin daughterboard features a 3-position jumper header in the middle of the board on the side where the chips are socketed. The jumper is placed on pins 1+2 by default for normal operation. In order for WinCNC to be used for laser rastering/vectoring, this jumper will need to be moved onto pins 2+3 before installing in the computer.



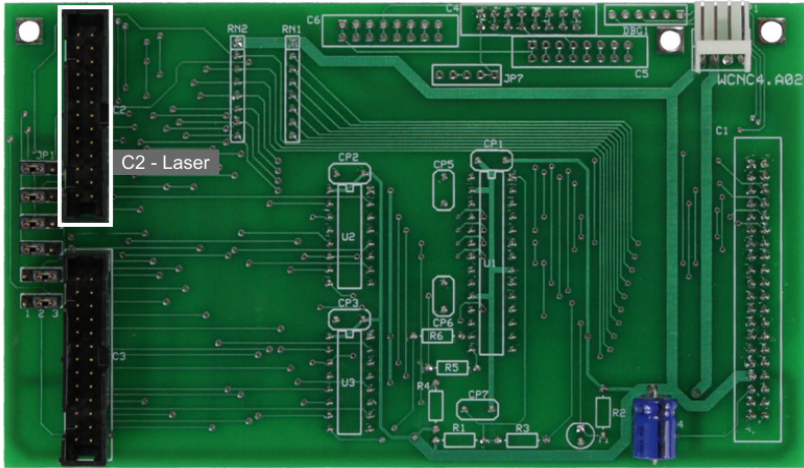
Setting the jumper to the laser position will cause the fifth axis to output a laser pulse rather than traditional step and direction signal. For as long as the jumper is on pins 2+3, this axis must be dedicated to laser operation. The laser pulse can then be accessed on pin #17 of the CN2 connector. Refer to 6-Axis 37-Pin Daughterboard Connector CN2 in the index for additional pin-out information.

## Configuring the Dual DB25 Daughterboard for Laser

For jumper configuration of the dual-25 daughterboard, refer to the illustrations 6 Axis Dual DB25 C2-3 Axis Connector and C3-4 Axis Connector in the index. After jumper configuration is complete, the laser pulse can be assessed through pin #14 of the C2 connector, highlighted in the picture below. If C2 was in use prior to the addition of laser, then the cable previously used will need to be switched to C3. Be aware that if there is not an axis free to be dedicated to laser then laser utilization will not be possible\*.

# WinCNC Controller

---



\*Note that a new revision of the dual DB25 board is currently being designed and field tested in order to remedy the issue of laser not being available on 4-axis machines without an unused axis. For additional information please contact us.

## Checking PIC33 Firmware with L920R1

It is important that WinCNC 3.0 users check the firmware version of the PIC33 chip installed on the daughterboard when considering the addition of laser to their machine. Currently, firmware version **PIC0015** is required and can be checked by entering "L920R1" in the command line and pressing enter. A replacement PIC33 chip with updated firmware can be purchased as needed.

## Laser Raster/Vector Support

Software feature enables will need to be purchased for a laser to be properly supported. Selecting "Help" and "About" on the toolbar will show the enables "Vector Laser", "Laser Engrave", and "Laser Carve" with a checked box after being unlocked in the license file.

## Additions to WINCNC.INI for Laser

There are two line entries that need to be added to the WINCNC.INI for laser calibration. These lines are **laser=** and **bmpoffset=**. Refer to the respective definitions as listed in the index for additional information.

This page is intentionally left blank.



**Overview**

WinCNC Controller has advanced features to provide the smoothest possible cuts. G-Code input is constantly buffered to 'vector match' moves. This means your machine only slows down when it needs to and then only as much as needed to stay within the acceleration parameters programmed. WinCNC Controller also does S-Curve acceleration through arcs and matches arc tangential velocities with straight moves to provide smooth arc moves. The result of these features is simply smoother cuts and less need for finishing operations. WinCNC Controller has a highly customizable, user-friendly interface that lets you take control of your machine, providing features previously only found on custom industrial controllers.

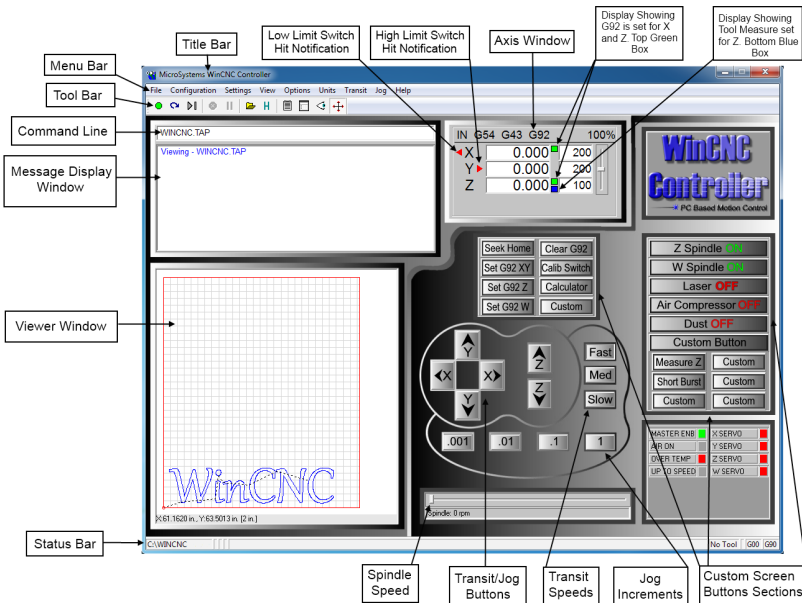
Any stepper, micro-stepper, or servo system with step and direction inputs can be interfaced using the included PCI card and microprocessor daughter board, or one of several available PCI I/O expansion cards. In addition to axis motion, up to 128 auxiliary outputs and 128 separate inputs may be defined. The intuitive user interface is easy to learn and convenient to use. All the features you need and expect are readily accessible via industry standard G-Codes. Simple keystrokes provide the ability to start, pause, restart, jog, and rapid transit any axis without navigating multiple screens and windows. Programs may be run from any line as needed. The programmed feed rates may be adjusted from 1% to 200% without stopping machine motion.

Run the program by double clicking the WinCNC icon on your Windows desktop. Program options may be selected from the pull down menu using the mouse, ALT+ menu key, or function keys while the machine is stopped. Files can either be typed into the command line or you can use the File menu's open option to bring the file name to the command line. Using the open command does not automatically run the file, it just brings the name to the command line.

## Screen Display

The following section contains a sample of the WinCNC software display. Common components are labeled to provide clarity.

**NOTE** : This screen may differ from your screen layout, as it is highly customizable and may have been altered by the CNC machine manufacturer.



## Screen Display Component Breakdown

### Menu Bar

The Menu Bar contains many of the main features of the WinCNC software. It is broken down into several generalized sections, as shown below. The menu bar selections are described in the following pages. If a menu option has a shortcut key, the shortcut key combination is listed in parenthesis ( ) after the name of the menu option.

File Configuration Settings View Options Units Transit Jog Help

## File Menu

**Open** (CTRL+O) - Opens a file.

**Edit** (CTRL+E) - Opens the editor. If a filename is in the command line, the editor opens that file.

**Simulate** (CTRL+S) - Simulates a file running to check for errors and estimate run time.

**View** (CTRL+V) - Views a file in the viewer window.

**Restart** (CTRL+R) - Opens the file restart dialog box to restart a file.

Choose the file you want to restart, either from the **Recent Files** list or by selecting the button with the 3 dots beside the **File Path** textbox.

The line that the file was stopped on should be shown in the **Line Number** textbox. You can start from that line or any other line that is legitimately in the file.

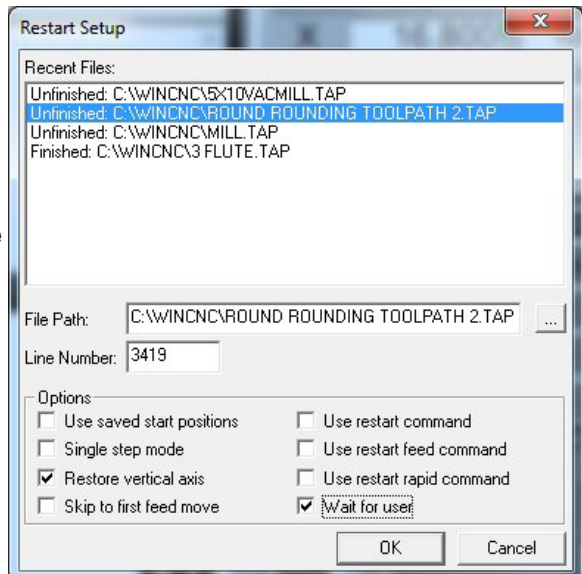
**Options** include:

**Use saved start positions** - Local positions that were used with the file are stored with the filename and can be used.

**Single step mode** - User must hit the Enter key each time to execute next like of code.

**Restore vertical axes** - Set's vertical axes to height commanded in file.

**Skip to first feed move** - Browse though file from the starting line number and starts on the next vertical feed move.



**Use restart command** - Use restart command **cmdrestart=** stored in the WinCNC.ini file.

**Use restart feed command** - Use restart feed command **cmdrestartfeed=** stored in the WinCNC.ini file.

**Use restart rapid command** - Use restart feed command **cmdrestartrapid=** stored in the WinCNC.ini file.

**Wait for user** - Pause until user hits Enter key before executing the file.

**History (TAB)** - Opens a dialog box with a list of recently ran files and/or commands. User can select one and click OK to execute it. Caution: The file/command will execute as soon as you click OK.

**List** - Opens a dialog box that allows the user to load and save a list of G-Code files or commands. User can select a file/command from the list to run. When file/command is completed, the list box will re-appear, ready to execute the next file or command. Files/commands that have already been executed are denoted by an asterisk (\*). This helps the user keep track of which files/commands have been run in the list.

**Create Home File (CTRL+H)** - Creates a home file at the current machine position.

**Import => DXF** - Allows the user to import DXF (Drawing Exchange Format) file.

**Import => HPGL** - Allows the user to import HPGL (Hewlett-Packard Graphics Language) file.

\*Note: The DXF and HPGL import feature is intended for cutting simple designs. WinCNC does not compensate for tool diameter when converting these files. For more complicated design, where intelligent tool pathing is important, please use a compatible CAD/CAM program.

**Digitize => Manual** - Enable/Disable the Manual Digitizing Toolbar.

**Digitize => Automatic** - Automatic digitizing (L802, L803).

**Digitize => Frame** - Skeletal Digitizing (L810). This scanning method scans a center line or spine, and horizontal lines or ribs of the object.

**Digitize => Outline** - Creates an outline trace around an object.

**Digitize => Laser => Run Scan** - Opens the laser digitization dialog box to set up laser scanning parameters.

**Digitize => Laser => Reprocess Data** - Opens a data file from a previous laser scan used to re-generate a 3D object with different filtering options to improve the quality of the 3D object.

**Exit** - Closes WinCNC.

## Configuration Menu

**D/A Calibration** - Calibration Settings for D/A (Digital to Analog) spindle speed control. This can only be used if D/A has been activated in the WinCNC Program.

**THC Configuration** - This allows the user to access the THC material configuration from either the drop down menu or the “Config” button on the THC screen display.

## Settings Menu

**Positions** - Opens the positions dialog box with options to view and edit stored positions. (Ex: G53 P# to call position in G-Code)

**Home Positions** - Opens the positions dialog box with options to view and edit stored home positions. (Ex: G0 H# to call home pos.)

**Tool Positions** - Opens the tool positions dialog box with options to view and edit stored tool positions for use with automatic tool changers. (This requires an additional software feature to be enabled)

**Tool Library** - Edits and views the defined tool library.

**Local Coordinates** - View XYZ coordinates.

## View Menu

**Resolution** - View current resolution settings for each axis.

**Acceleration** - View current acceleration settings for each axis.

**Limits** - View current limit settings for each axis.

**Inputs** - Views real-time input states. Useful as an aid to check for incorrect limit switch settings verses bad hardware/wiring.

**Outputs** - Views and allows changes to real-time output states.

**Toolbars** - Enable/Disable the main toolbar.

**Toolbars=> Size** - Change toolbar from small to large.

**Refresh View (F5)** - Refreshes the screen.

**Clear Screen (CTRL+C)** - Clears the message display window.

### **Options Menu**

**Soft Limits (CTRL+L)** - Enable/Disable the software limits.

**Keyboard (CTRL+K)** - Enable/Disable the keyboard transit/jog control used to move the machine with the keyboard.

**Buttons** - Used to customize user buttons.

**Auto Repeat** - Auto populate the command line with the last command executed.

**Auto Run** - Enable/Disable the auto run feature. Allows files dropped into **autodir** to automatically run. See **autodir=** and **autofdir=**.

**Auto Preview** - Enable/Disable the auto preview feature.

**Single Step (CTRL+P)** - Enable/Disable single line execution of G-Code files.

### **Units Menu**

**Inches** - Sets the unit of measure to inches.

**Centimeters** - Sets the unit of measure to centimeters.

**Millimeters** - Sets the unit of measure to millimeters.

### **Transit Menu**

**Slow (F2)** - Sets the transit speed to slow.

**Medium (F3)** - Sets the transit speed to medium.

**Fast (F4)** - Sets the transit speed to fast.

### **Jog Menu**

**0.001 (F6)** - Sets jog increment to 0.001 units of measure.

**0.01 (F7)** - Sets the jog increment to 0.01 units of measure.

**0.1 (F8)** - Sets the jog increment to 0.1 units of measure.

**1 (F9)** - Sets the jog increment to 1 unit of measure.

**Custom (F10)** - Allows the user to set a custom jog increment.

## Help Menu

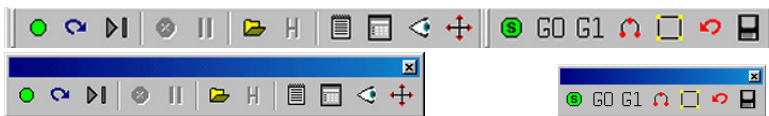
**Help Topics (F1)** - Displays the WinCNC Users Manual or other Help files set up by your manufacturer.

**Update Program** - Opens the update utility.

**About** - Displays the WinCNC “About” box which contains important information about your specific software package. The About box displays your security key serial number, the software version number, your user level, the maximum number of axes, the table size limitation, and which additional features you have enabled. If you do not have a feature enabled that you would like or need, you can contact your software vendor for an upgrade.

## Tool Bar

The tool bar is a collection of shortcut buttons that perform specific actions. The toolbars in WinCNC are floating and can be positioned or docked to the user’s preference. The image below shows both the standard and manual digitizing toolbars both docked in the normal toolbar area. The additional images show each toolbar separately as a floating toolbar. The explanation of each button on the toolbar is as follows. The explanations move sequentially from left to right along the toolbar.



**Start Motion** - This button will begin the command or job listed in the command line.

**Restart** - Allows you to pick a job file and the line to start that file on. This is useful if you want to skip over lines in a job file, or if you have aborted a job and want to start back at the point you aborted from. You can also choose to run the file in single step mode.

**Single Step** - When selected, this allows you to execute one line of a G-Code file each time you hit ENTER.

**Abort Motion** - Aborts a command or job that is running.

**Pause/Continue Motion** - Pauses/Continues a command or job that is running.

**Open File** - Opens a browse box that is used to open a job file.

**View History** - Opens the command history box, which allows a user to execute a command used previously. When an error is found in the WINCNC.INI file, the line containing the syntax error is displayed here in red when WinCNC starts.

**Edit** - Opens the default editor specified in the WINCNC.INI file. WinCNC uses Notepad by default. The editor can be used to open job files, listed in the command line in the editor.

**Simulate** - Simulates a file to check for errors and run-time.

**View** - Opens a file for viewing in the viewer window.

**Soft Limits** - Enables/Disables Soft limit features.

**Manual Digitize Buttons** - The seven buttons on the manual digitization toolbar are for use only with the Manual Digitize feature, and will only be visible after showing the manual digitize toolbar under the View->Toolbars section of the menu bar.

**Start a Manual Digitized File** - Starts a manual digitized file and enables the manual digitize mode.

**Add a Rapid Move** - Adds a rapid move to the manual digitized file.

**Add a Feed Move** - Adds a feed move to the manual digitized file.


**Add an Arc Point** - Used to add arcs into a manual digitized file.

**Close Shape** - Used to close the last move in a shape without moving the machine.

**Undo** - Used to undo previous moves.

**Save** - Allows you to save the manual digitized file in either G-Code or DXF file format.

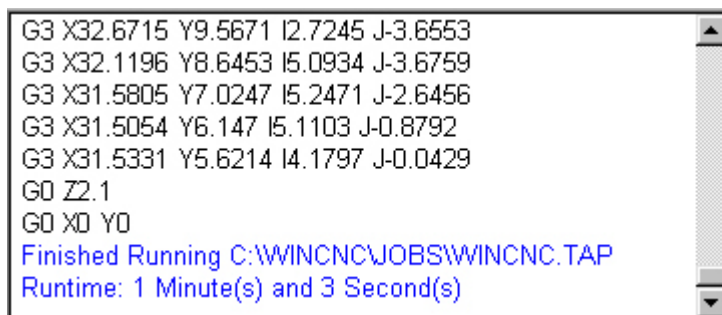
## Command Line



WTEST.TAP

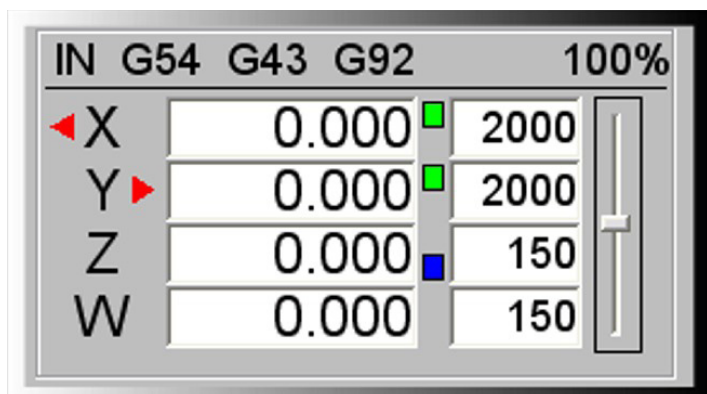
The command line is the input line for users to enter commands or job files to be executed.



Message Display Window

```
G3 X32.6715 Y9.5671 I2.7245 J-3.6553
G3 X32.1196 Y8.6453 I5.0934 J-3.6759
G3 X31.5805 Y7.0247 I5.2471 J-2.6456
G3 X31.5054 Y6.147 I5.1103 J-0.8792
G3 X31.5331 Y5.6214 I4.1797 J-0.0429
G0 Z2.1
G0 X0 Y0
Finished Running C:\WINCNC\JOBS\WINCNC.TAP
Runtime: 1 Minute(s) and 3 Second(s)
```

The message display window is the main output window displaying the commands that have been executed, messages to the user, or errors that have occurred during an operation.

Axis Window

The axis window is the primary display for current information about each axis on your machine. The current positions and velocities for each axis are displayed in the text boxes. Displayed above the position and velocity boxes are the current units of measure, any currently active modes, and the current feed rate override percentage. Red triangles to the left and right of each axis label indicate low or high limit switch hits. A red triangle to the left of an axis label indicates a low limit switch hit, and to the right indicates a high limit switch hit. A green box displayed to the right of the position box indicates a temporary workspace (G92) setting, and a blue box indicates a tool length measure or workspace setting. To the far right side of the axis window is the feed rate override control slide bar.



**Transit Speed Buttons**

These buttons allow you to specify the transit speed of your machine.

Fast transit is the default.

In the cncscrn.ini file, change the “default speed” field to 1, for the speed you would like to be default, all others must stay 0.

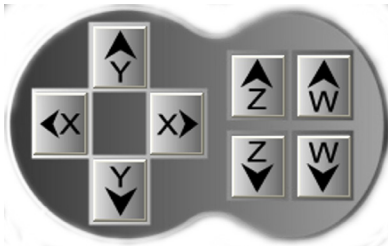
Ex: for default medium speed, use the following:

```
“JogMode”, 0,5,35,25,7,1,“Med”, “”, “”, -1,-1,“med.bmp”
```

**Jog Increment Buttons**



These buttons allow you to specify the increment of a jog move.



**Transit/Jog Buttons**

These buttons are what allow you to initiate a manual transit or jog movement from the console of the computer that runs the machine.

In transit mode, holding the button down will continuously move the machine. In jog mode, each time a button is pressed the machine

will move according to the jog increment that has been selected. Holding the button down in jog mode will move the machine only once.

## Custom Screen Buttons Section

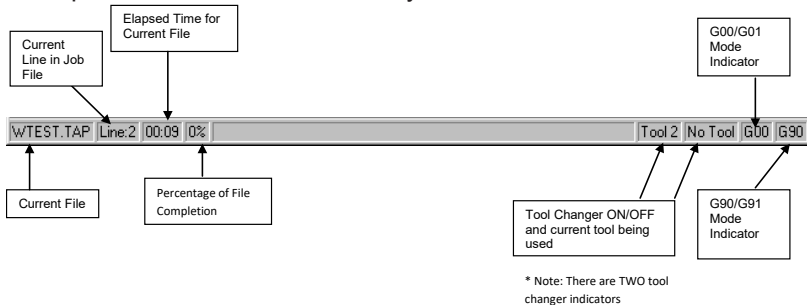


This section of controls is a customizable button section that allows the user to place shortcut buttons to activate heavily used commands.

**Note:** See Section 4 Screen Configuration

## Status Bar

The status bar is the label along the bottom of the WinCNC window that looks similar to the one shown below. This bar provides the user with the status of several features in WinCNC. Each section in the example below is labeled for clarity.



**Shortcut Keys**

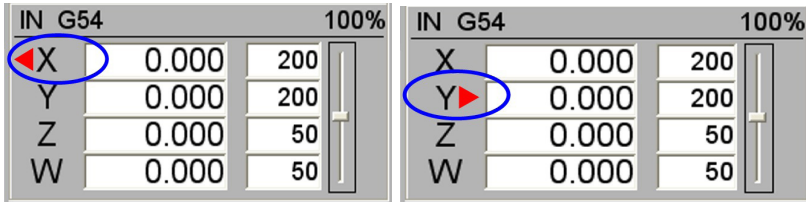
<b><u>File Menu Shortcuts</u></b>	
Open	Enter (with blank command line), CTRL+O
Edit	CTRL+E
Simulate	CTRL+S
View	CTRL+V
Create Home File	CTRL+H
<b><u>View Menu Shortcuts</u></b>	
Refresh View	F5
Clear Messages	CTRL+C
<b><u>Options Menu Shortcuts</u></b>	
Keyboard	CTRL+K
Soft Limit	CTRL+L
<b><u>Transit Menu Shortcuts</u></b>	
Slow	F2
Medium	F3
Fast	F4
<b><u>Jog Menu Shortcuts</u></b>	
.001	F6
.01	F7
.1	F8
1	F9
Custom	F10
<b><u>Help Menu Shortcuts</u></b>	
Help	F1

Shortcut Keys continued

<b><u>Tool Bar Shortcuts</u></b>	
ESC	Aborts the current file or command
SPACE	Pauses a file or command
ENTER	Starts/restarts a file or command
TAB	Opens the command history box
CTRL+R	Opens the restart file box
<b><u>Feed Rate Override Shortcuts</u></b>	
INSERT	Increases override rate
DELETE	Decreases override rate
CTRL+either	Resets feed rate to 100%. No override settings
<b><u>Manual Digitize Shortcuts</u></b>	
After opening the Manual Digitize toolbar and clicking the green start button, the following shortcut key combinations become active for adding moves to the digitized file.	
Add Rapid Move	CTRL+J
Add Feed Move	CTRL+F
Add Arc Midpoint/Endpoint	CTRL+A
Undo Last Move	CTRL+U
Close Current Shape	CTRL+D
Save	CTRL+W
<b><u>Spindle Speed and Laser Power Shortcuts</u></b>	
Spindle Speed Alone or Laser Power Alone	+ - (plus and minus)
Spindle Speed and Laser Power Together	Laser Power uses + - Spindle Speed uses Shift + -
CTRL+(Function Key)	See "ctrlf#=" in configuration settings

**Limit Switches**

The default channel settings will normally provide proper interfacing with the machine’s limit switches. Before attempting to move the machine, verify that all limit switches are functioning properly. Limit switch status is displayed in the position window using a red triangular indicator to the left or right of the axis label.



The indicator to the left of the axis label shows low limit status for that axis. The indicator to the right of the label shows high limit status. If the indicator is present then the limit switch is closed, otherwise it is open. The lim\_mode setting is used to control how WinCNC responds to limit switch triggers.

First toggle your limit switches by hand. Verify that the appropriate limit display toggles. If it does not toggle then you need to adjust WINCNC.INI. See the WINCNC.INI reference sections of the manual if the default settings are incorrect for your machine.

It is recommended that you verify limit switch operation at the beginning of each session. Do not attempt movement until the limit switch displays toggle correctly. Once the limits are working try a small move. If nothing moves, the wrong axis moves, or movement is in the wrong direction then adjust WINCNC.INI.

Perhaps the easiest way to do preliminary testing of movement is with the Jog mode. Select “Jog .1”, and select the KEYBOARD option under the OPTIONS menu. If you are using the default jog key configuration, use the LEFT and RIGHT arrow keys to move X, the UP and DOWN arrows to move Y, the PAGEUP and PAGEDOWN keys to move Z, and the HOME and END keys to move W. Otherwise use the buttons that are defined for your system.

## Homing the Machine

When WinCNC Controller is started, the display is set to the last known position. If the machine has been moved manually or has drifted while powered down, this position will not be accurate. The G28 command must be used to home the machine. G28 homes your machine in the order specified in the WINCNC.INI file on the "g28move=" lines. The first listed homes first. Normally the vertical axes will home first, then horizontal axes would home next. Vertical axes usually home to high limit switches. Horizontal axes will home to either the low or high limit switches as set up by your machine manufacturer. After an axis touches off its homing limit switch, it move off the distance specified in the WINCNC.INI file and set to zero. This position is Machine Zero for that axis.

It is important to use G28 to set Machine Zero since many WinCNC features are calculated from this position. Soft Limits and Boundaries cannot be used if Machine Zero is not properly set. G28 will search for the limits at a slow feed rate set in the WINCNC.INI file. The "lim\_step=" setting in WINCNC.INI can help minimize the impact to the machine when stopping at the limit switches.

G28 can also be used to home only specified axes. (I.e. G28Z homes only the Z axis).

Normal machine operation using WinCNC Controller would be:

Start WinCNC.

Enter G28. Push ENTER. (Machine Goes Home)

Enter part program to cut. Push ENTER. (Job file runs)

**NOTE:** Most machine manufacturers have a button on the screen that handles homing the machine for you. It might be named "Home", "Seek Home", "Initialize", or something similar, but clicking it will run the G28.

## Feed Rate Override

WinCNC accepts feed overrides to increase/decrease the feed rate. The feed rate can be adjusted from 1% to 200% of the programmed rate. The override rate cannot be set higher than the max velocity or G0 rate for a given axis. Use the Insert key to increase and the Delete key to decrease the programmed feed rate, holding CTRL and pressing either Insert or Delete will result in the feed rate being reset to the programmed value. There is also a slider bar to the right of the feed rate screen display that can be used to initiate a feed rate override. The override feed rates will be displayed in place of the

programmed feed rates in the display. The feed rate will change colors to depict that an override is in use. If the override rate is higher than the programmed rate the color of the feed rate display will be red, if it is lower than the programmed feed rate it will be blue.

### **Coordinate System**

The Machine Zero (MZ) set by homing the machine becomes the anchor point for all positions specified in subsequent G-Code commands. This is also considered absolute zero, not taking into account any temporary homes (G92's) or tool measures.

Local Zeros (LZ) are set using the G92, G92.1 and G92.2 commands.

The position display box displays the current axis coordinates and the axis coordinate modes. The axis coordinate mode is displayed to the right of the axis position.

A green box indicates that the axis has a Local Zero applied. (G92, or temporary home).

A blue box on a horizontal axis indicates that a workspace other than G54 is in effect. (G55, G56, G57, G58, G59: ex: using W axis instead of Z).

A blue box on a vertical axis indicates that tool length measure is in effect.

The coordinate modes show how the axis values in a given G-Code command will be interpreted.

MZ Coordinates - values are relative to MZ. (Or absolute zero)

G92 Coordinates - values are relative to LZ specified by G92  
(Ex: enter the command G1 X10 Y10, then enter. Next type G92X0Y0, then enter. X & Y will now show as X0, Y0)

G55, G56, G57, G58, G59 Coordinates for horizontal axes - horizontal axes values are offset for vertical axes heads.

G43 Coordinates for vertical axes - vertical axes values are offset for M37 tool measure values.

G92 and G55, G56, G57, G58, G59, G43 - values are offset for both.



**G91 Mode**

G91 mode is called relative or incremental mode. In G91 mode values specify distances. For instance if the X position is currently 20 and G0 X8 is specified the machine would move +8 units of measure in X to X28.

A G-Code program written in G91 mode may be run from any position. Since the moves are relative, the starting position does not matter. The program will run properly from any starting position. However, to get the proper results, the machine must be set to the proper position in relation to the workpiece before the piece is run.

The axis coordinate status does not have any effect on how the program runs. It does however affect the coordinate display while the program runs.

**G90 Mode**

G90 mode is called absolute mode. In G90 mode values specify positions. If the X position is 20 and G0 X8 is specified the machine would move -12 units of measure in X to X8.

In G90 mode the situation is different. The initial of the machine position will not affect where the part runs since the first move will go to the absolute position specified on the table. Instead of setting the machine position properly before running a G90 program.

The G92 commands are used to change the coordinate system so that any position may be temporarily made to be Local Zero (LZ).

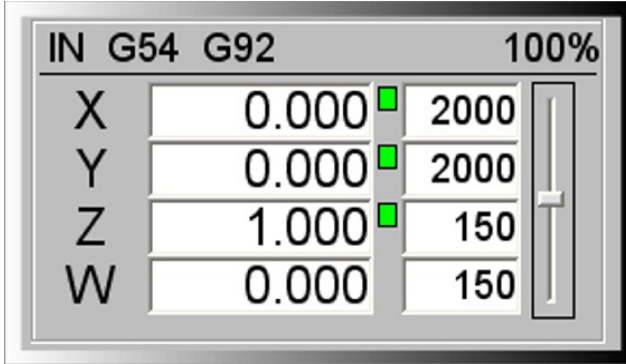
Most G90 programs are written relative to a starting position of X0Y0 with Z0W0 being either the bottom or top of the workpiece. All that is then necessary to run the program is to insure that the current LZ is set to match the program before running. There are several ways to do this.

The machine may be positioned to the proper position and the G92 command used to set the proper coordinates for running.

Example: The workpiece is fixtured at X20Y20, is 1" thick and Z0 in the part file refers to the top of the workpiece.

G90	Commands in G90 Mode
G92	Turn off any current G92s
G0Z2	Set Z 2" above the table and 1" above the workpiece
G0X20Y20	Go to corner of workpiece
G92X0Y0Z1	Set this position to be 1" above LZ

The Axis Window shows:



Alternately LZ may be set by shifting MZ the desired amount using G92.1. Using G92.1 it is not necessary to move to the workpiece first.

**Example**

G90	Commands in G90 mode
G92.1 X20Y20Z1	Move LZ 20" in XY and 1" above the table

The coordinate display shows the current position relative to the workpiece.

When writing a G-Code program there must always be a starting point. Typically it would be the lower left corner (in XY) of the piece to be machined and either the upper or lower surface of the workpiece or a position a known distance above the workpiece (ZW). In a G91 mode program this point does not need to be given a coordinate value. In G90 mode this point is assigned a coordinate value (usually 0,0,0) and all positions are specified relative to this starting point.

When running the program the machine is positioned to this starting point and G92 is used to set position to the start position for the program. Alternately G92.1 may be used to shift MZ.

**Workspace**

The G54/G55/G56/G57/G58/G59 functions are used to allow for easy switching between workspace coordinates. Each G54/G55/G56/G57/G58/G59 line you wish to use must be specified in the WINCNC.INI file.

**WINCNC.INI Setup:** For descriptions see the Systems Configuration Section of the manual.

G54=X# Y# T# A# R# O#

After configuring the G54/G55/G56/G57/G58/G59, the G54/G55/G56/G57/G58/G59 commands can be used to switch workspace coordinates, or for rotating a rotational axis.

If using the type 0 G54/G55/G56/G57/G58/G59 for switching vertical heads just use the G54/G55/G56/G57 commands alone to switch workspace coordinates.

If using the type 3 G54/G55/G56/G57 for rotational axis command G54/G55/G56/G57/G58/G59 X# Y#, where X and Y are the positions to rotate towards.

**Head Swap**

L12 provides a means of cutting with a head not specified within a program and of using multiple heads simultaneously.

Example:

L12WZ moves Z whenever W is specified and moves W when Z is specified.

L12ZZ moves Z and W together whenever Z is specified.

L12 alone or L12ZW sets normal operation.

**Tool Length Offset**

M37 is used to measure tool length and set ZWUV workspace.

By using M37 it is not necessary to measure tool length for every workpiece thickness change. Using M37 in combination with Soft Limits and Boundaries also provides a means of protecting the table from being routed accidentally. After M37 is set G28 will set the ZW to the actual position of the tool tip above the table.

Example: (without automatic tool changer)

Move the Z head tool tip to the table.

Command M37 Z0.

M37 turns on G43 mode, indicating that tool length offsets are active. Tool length offsets can be disabled using G49 and re-enabled with G43 without re-measuring.

**Soft Limit / Boundaries**

Soft Limits and Boundaries are used to keep programs and command from moving into the limit switches. Values must be set in WINCNC.INI before using.

**Soft Limits** check absolute position command against the limit switch positions. If a position is commanded that would result in a limit switch being hit, a limit error will be displayed and the move will not be run.

lolim=X# Y#	Distance from MZ (machine zero) to XY low limit switches less .1”
hilim=X# Y# Z# W#	Distance from MZ (machine zero) to XYZW high limit switches less .1”

ZW low limits normally are not specified since desired positions vary with tool length.

**Boundaries** check workspace positions. In G54 mode the Z head will not be allowed to leave the table. In G55 mode the Z head might move off the table but the W head will be kept on the table.

lobound=X# Y# Z# W#	Distance from MZ to XY table edge and top
hibound=X# Y#	Distance from MZ to XY table edge

ZW high boundaries normally are not specified since desired positions vary with tool length.

ZW low boundary checking is available if M37 is used. Use M37 to set ZOW0 to the tabletop. The ZW hi limits and low boundaries are then both set to 0. This allows movement between the limits and the table.

Limits and Boundaries may be enabled or disabled in the Settings Menu and in WINCNC.INI using softlim=t0 or t1. T0 will disable them, t1 will enable them. Soft limits and boundaries are linked so this setting controls both.

## Repeat Command

The command or file name history may be viewed by pushing the TAB key. To repeat a command from the history list select the desired command or file name using the mouse, and then click OK.

## Simulate

WinCNC supports the simulation of files to check for errors, estimate runtime, and check min/max positions. To use the simulate function of WinCNC make sure the file name is typed into the command line and instead of running the file, you will use the simulate call to simulate the file. This is done by using either the Simulate command found under the File menu, by pressing CTRL+S, or by pressing the simulate button on the toolbar. Once you initiate the simulate feature, the software will display a screen showing the specifics of the file simulated which includes the starting (x,y) positions, the ending (x,y) positions, the min and max (x,y) positions, the number of lines ran in the file, and the total time it will take to run the file.

**NOTICE:** The time function in the simulation feature does not take into account smoothing. If you have smoothing set up on your machine, the time it takes for the file to run should be less than what is displayed in the simulation data.

**Viewer**

WinCNC contains a built in viewer window that will allow you to display the output of G-Code files before the job is actually ran. The viewer can also display a G-Code file line by line as it is run by WinCNC. When the file is displayed in the viewer you can distinguish G0 moves from G1 moves by the color of the line. By default, G0 moves are shown as a black dashed line. G1 moves are displayed as a solid blue line. See the Screen Configuration section if a different color configuration is desired.

You can also re-center and zoom in/out of the display within the viewer. To view a file before you run it, enter the file name into the command line and then press CTRL+V, select View from the File menu, or press the viewer button on the toolbar. The file will be displayed line by line automatically when you run the file. Once the object is loaded into the viewer the following controls can be used:

Zoom In Incrementally - click the left mouse button.

Zoom Out Incrementally - click the right mouse button.

Selected Zoom - hold down the left mouse button and drag the box around the area to zoom in on.

Pan or Re-Center Object - hold CTRL and click either mouse button.

Reset Image - hold shift and click either mouse button.

To set up the viewer window you must include lines in both the WINCNC.INI and the CNCSCRN.INI files.

WINCNC.INI Setup (for descriptions see the System Configuration section of the manual)

table=X# Y# W# H# B#

CNCSCRN.INI Setup (for descriptions see the Screen Configuration section of the manual)

“Display”, “Viewer”, 370, 150, 270, 270

## Pump Control

### Wiring for the Daughter Board

#### **Pulse Width Modulation**

-Jumper Pin 1 of connector 4 (C4) on the daughter board to Pin 2 of C4.

-Disconnect trace between Pin 2 of C4 and Pin 3 of C4.

-Jumper Pin 3 of C4 to Pin 4 of C4

-Pin 11 of CN2 is the pump control signal.

-Pin 15 of CN2 is the pump reverse signal.

#### **Frequency Mode**

-Pin 11 of CN2 is the pump control signal.

-Pin 15 of CN2 is the pump reverse signal.

**WINCNC.INI Setup** (for descriptions see the System Configuration section of the manual)

pumpcont=T# L# H# A# B# I# R#

**Commands** (for descriptions see the Commands section of the manual)

L55, L56, L57, L58, L59

**Buttons** (see also the State Buttons section in the Screen Configuration section of the manual)

#### **Pump Mode On**

“StateBtn”, 280, 5, 50, 25, 4, 0, “Pump On”, “L55”, “”, -1, -1, “”

#### **Pump Mode Off**

“StateBtn”, 280, 35, 50, 25, 5, 0, “Pump Off”, “L58”, “”, -1, -1, “”

#### **Purge Mode**

“StateBtn”, 280, 65, 50, 25, 6, 0, “Purge”, “L56”, “L57”, -1, -1, “”

**Manual Digitize**

WinCNC supports the ability to manually digitize a part and then output the file as either G-Code or DXF format. The manual digitize toolbar is started by clicking File->Digitize->Manual. The toolbar icons are displayed and summarized in the screen display section of the manual. The Manual Digitize feature supports the addition of rapid moves, feed moves, and arc moves. It also has support for undo, close shape, and save. To digitize a part manually, place the part on the table and transit the machine to the selected starting point and then enable the manual digitize mode.

Now that manual digitization is enabled, to add a move into your digital file, transit the machine to the desired point and then press either the add rapid (G0) or add feed (G1) button on the toolbar depending on the type of move needed. To add an arc, move to the beginning of the arc and add either a rapid (G0) or feed (G1) move, then transit to any point along the arc and press the add arc point button on the toolbar. The first time the add arc button is pressed it is setting a mid-point for the given arc. Once you have the midpoint selected transit to the end of the arc and press the add arc button again. This will complete the arc move and add it to your digital file. Continue transiting along the part adding in the required move types.

Once you have your part completely digitized click the save button on the toolbar. If you saved the digitized file as a G-Code formatted file, you can then use that file with WinCNC. If you saved the file as a DXF file, you can import this file into any CAD/CAM package that supports DXF file formats and make any needed adjustments.

The Manual Digitize feature of WinCNC also supports the ability to undo any previous moves during the digitization. If at any point you wish to go back before moving the machine or adding a move press the undo button in the toolbar and the machine will lift the vertical head, move back to a previous location, and lower the vertical head back to its initial position. The location the undo button moves to depends on the last type of move you made. If it was simply a transit move out into the table, undo will move the machine back to the point you started at. If undo is used after adding in a new move, the machine will move back to the last point that was added to the file.



**Manual Digitize continued**

The manual digitize feature also has support for closing shapes. The close shape button works by adding a feed style move from the current point to the first point. The close shape feature handles only 1 shape at a time, and any rapid moves that are added repositions the first point of the shape to the ending point of the rapid move. If no rapid moves are added, the starting position for the shape remains at the initial point.

For example: Two squares that are 10"x10" with a 5" separation along the X Axis would be digitized as follows. Start by transiting the machine the desired starting point and enabling the manual digitize feature. Once it is enabled, transit the machine to X10Y0 and add a feed move. Next transit the machine to X10Y10 and add a feed move. Next transit the machine to X0Y10 and add a feed move. At this point, three of the four needed sides of the square have been added. We will now use the close shape button to continue with the example. Once the close shape button is pressed, the vertical head will lift, the machine will move to the starting point of the shape, and the vertical head will drop back to its previous position. Once the machine finishes moving, transit the machine to X15Y0 and then add a rapid move using the add rapid button in the toolbar. This will set the current position as the starting point of the next shape. This example would then output a file with two separate 10"x10" squares with a 5" separation between them.

**Frame Scan Digitizing**

Default Values	
XY Increment	0.5
Z Increment	0.1
XY Feed rate	30
Z Feed rate	30
Bottom Offset	0
Top Offset	0
XY tolerance	0.01
Scan Width	0
Scan Length	0
Scan Center	Unchecked
Center Scan Direction	Negative
Slice Scan Direction	Positive
Side to Scan	Top
Z Height to Calibrate	0
Distance to Raise Stuck Probe	1

**Scan Increment** - the distance the probe will attempt to move to reach new points. The actual amount can be less if the probe hits something during a move.

**Scan Feed rate** - the speed the machine will travel between points.

**Bottom Offset** - distance to shift the bottom scan data. This distance can be positive or negative.

**Top Offset** - distance to shift the top scan data. This distance can be positive or negative.

**XY Tolerance** - the allowable difference between points on the X or Y axis when finding an edge. Example: XY Tolerance is 0.01, the current X point is -0.4370 and the previous point was -0.4467. The difference is less than the allowable tolerance, so the slice will end.

**Maximum Scan Dimensions** - these dimensions should be set to the maximum distance that the user wants the probe to search for an edge. If the board will be clamped on the ends or sides, input in the distance between the clamps for that axis. For the other horizontal axis, set the size to the maximum amount that the probe should travel assuming that it doesn't find an edge. Always use positive values.

**Center Scan Direction** - the direction that the probe should travel on the Y axis. If set to negative, the probe will travel from high Machine Y to low Machine Y.

**Slice Scan Direction** - the direction that the probe will travel on the X axis. If set to negative, the probe will travel from high Machine X to low Machine X.

**Slice Position Boxes** - the input boxes beside the checkboxes in the slice scan group. This is the position on the Y axis where the slice will be measured. Always use positive values, even for negative scans.

**Side to Scan** - the side of the board to be scanned, Top or Bottom.

**NOTE:** A complete frame scan will consist of a Top and a Bottom scan. This means that the Frame Scan utility must be ran once for each side. The values that you set in the previous scan will be retained in their appropriate locations, so there is no need to retype them. The utility will compile all slices after each scan.

## How to Use the Frame Scan Digitizing Feature

1. Start WinCNC. **Important!** Make sure to do a G28Z or an L28Z to set the machine Z zero to a level above the highest point of the workspace. This is necessary because the machine does a G53 (moves to machine Z zero) between slices. If you do a G53 and machine Z zero is below material height, the probe will hit the surface at rapid speed and be forced to stop very quickly, which could result in damage to the probe or the part.
2. Move the probe to the XY location where you want to start the scan.
3. Do a G92XY to set this beginning point to local zero.
4. If you want to do an L82, move the probe to the desired position.
5. Click File->Digitize->Frame to open the Frame Scan dialog box.
6. Click Browse and set the file name and path to use. The file will always save with a .CSV file extension.
7. Set up the Scan Increments (positive numbers).
8. If a Top or Bottom offset is to be used, set it. The default is zero.
9. Check the Scan Center box if a center scan is to be performed.
10. Choose the center scan direction.
11. Choose the slice scan direction.
12. Check the box next to a slice, then enter a Y position (positive). Repeat this for the number of slices desired.
13. After slices have been chosen and values are entered, choose the side to scan - Top or Bottom.
14. If an L82 is to be performed, enter the new Z value for the surface and press Calibrate. The probe will descend to the board and, when it touches, will calibrate the board surface to the height set in the box. This is very useful if you know the thickness of the board. Just calibrate to half of the thickness for each side and the final point cloud will center around Z zero. Example: The board is measured at 2.5" thickness. If the board is calibrated to 1.25" for each side at the point where it was measured, the finished scan should center around Z zero - half above zero, half below.
15. Set the height for the probe to raise if stuck. If the probe's Z zero is below its current position and it sticks, the probe will try to raise the distance specified here. If the Z zero is above the current position and the probe sticks, the probe will raise to zero. In either case the user is asked to "unstick" the probe and hit Enter - after which the scan will continue starting with the point where the probe stuck.

16. Click Run. **Important!** This will start the digitizing process and the machine will begin moving. If Options->AutoPreview is enabled the user will have to press ENTER again to start digitizing.
17. After the first side scan is completed, turn the part over to set up scanning for the other side.
18. If the starting point has changed, do a G92XY to reset the local zero.
19. If you are using the L82 setting for bringing the top and bottom together, move the probe the same X,Y point that was used on the other side of the part.
20. Click File->Digitize->Frame to open the Frame Scan dialog box.
21. The values for the previous scan will still be set. Leave all values set "as is" except the slice direction and the side to scan.
22. If the L82 is to be performed, make certain that the correct value is in the box and click Calibrate.
23. Click Run. **Important!** This will start the digitizing process and the machine will begin moving. If Options->AutoPreview is enabled, the user will have to press ENTER again to start digitizing.
24. After this side scan is completed, your CSV file will be complete.

**NOTE:** A large XY increment, combined with a large Z increment, can cause rough edges. Too large of an XY increment, combined with too small a Z increment might cause the scan to end before the true edge is found, especially if the XY tolerance is set too large.

Auto Digitizing

**Automatic Digitize**

Probe Type

Up/Down  Up/Down/Side

Axis

Width: X Length: Y Retract: Z

Area

Width: 10.0000 Length: 10.0000 Retract:

Increment/Resolution

Width: 0.0100 Length: 0.0100

False Bottom

Use False Bottom Bottom Position: 0.0000

Output File

C:\WINCNC\PART.STL Browse

File Split Segments: 1 Reprocess

OK Cancel

WinCNC supports 3D digitized scanning. This is a feature and must be enabled by the machine manufacturer (or Microsystems World CNC) before it can work.

The procedure for using a digitizer is to position the probe directly above the Z0 plane and perform an L82 command to set the 0 (Zero) plane. See L82 reference in this manual.

Once this has been done, position the probe over the beginning of the scan rectangle, you can perform a G92XY at this point if you want the corner to be (X0, Y0). If the probe is enabled and it comes into contact with something while transiting, you must use the transit keys to move the probe up in order to release it.

Now select the File->Digitize->Automatic menu item to show the Auto Digitize dialog box.

With up/down type scanning, your probe will lift to clearance height each time, the probe will rapid to the next point location and lower to touch at feed rate, lift again to clearance height and rapid to the next point location and lower to touch at feed rate. It is very important to set the clearance height above the highest point on the surface you are scanning. This is done with the Retract setting for up/down scanning.

With up/down/side scanning the probe will lower until it touches on the first point, raise just enough to clear itself and move sideways trying to reach its next point. If it encounters anything in its path, it will raise to clear it. Once it clears the obstacle, it continues moving sideways until it reaches its next point. It then lowers and takes that point. It continues in this method until the part is scanned.

You can set a false bottom. An example of this: I have a 3" tall object with the center hollowed out that I just need the top section. I do not want the probe to lower into the center. I set a false bottom that is about .25" below the top surface. The probe tip will lower to this height and set a point. It will not lower beyond this point.

At the top of the dialog box, choose the type of scan you wish to do. If up/down/side scanning is not enabled in your software you will only be able to do up/down style.

Choose the axis for width, length, and the retract axis (usually Z). If you choose a rotary axis for width or length, you can even scan with a rotary axis. If you do choose to scan with a rotary axis, which ever direction you choose for the rotary must be set in degrees. Example: all the way around an object = 360. Half way around the object = 180.

Set the area you wish to scan, width and length. In the case of an up/down type scan you also need to set the retract height. Again, in the case of an up/down style scan, make certain the retract height is above the highest point on the scan area.

Set the Increment/Resolution for width and length. As with Area, if you're using a rotary axis as one of your axes, set that increment in degrees. Setting the increment to 1 means you would get 360 points for each full rotation of the workpiece. Setting it to .5 would give you 720 points per full rotation of the workpiece.

Click the Browse button, browse to the location where you wish to save your file, give the file a name, then click Save. Click the OK button to start the scan. There can be no spaces in either the name or the file path. **This step is mandatory for the scan to work.**

You can reprocess the last file you scanned by inputting the number of File Split Segments. It will split a file into 1 to 6 segments. After setting File Split Segments, click Reprocess.

A scan that is aborted in the middle of the scanning process will generate a partial file with all points after the abort being set to Z0.

To enable digitizing beyond having the feature enabled, the probe must be wired to an unused input pin and you need two lines added to the WINCNC.INI. First, an auxin must be defined for the probe. Example: the probe is wired to pin 10 of the daughter (an input), it's a side scanning probe and the vertical axis for the probe is Z. You need a line such as: "auxin=c1p3b7" (no quotes) See AUXIN= in this manual. Next you need another line: "scan=c1a2t2" (no quotes) See SCAN= in this manual.



Outline Digitizing

**Outline Digitize**

Output File Type:  DXF File  CSV File

File Name to Save As: C:\WinCNC\PART.DXF

General Settings: Scan Increment: XY: 0.000 Scan Feedrate: XY: 30.000

Scan Size:  180 Degree Scan  360 Degree Scan

X Scan Direction:  Positive  Negative

Y Scan Direction:  Positive  Negative

First Forward Move: Initial Y Move Distance: 1.000

Probe Settings: Degrees to Back Off on Stuck Probe: 90.000  
Probe Stylus Diameter: 0.000  
Probe Trigger Adjustment: 0.000

WinCNC supports simple X/Y only outline/edge digitizing. This is part of the Up/Down/Side scan feature and must be enabled by the machine manufacturer (or Microsystems World CNC) before it can work.

For outline scanning, the initial move will always be along the Y axis (front to back or back to front). Move the probe tip to within Initial Y Move Distance from the front edge of the part you are scanning. Go to File->Digitize->Outline. The dialog box shown above will appear.

Choose whether you want a CSV (a point cloud) or a DXF (lines) type file. Important: You must click Browse each time you want to create an outline scan, navigate to the folder where you want to create your file, type a filename into the filename box and click Save.

Set the scan increment. Start with something like 0.1" and slowly work your way down after you see how it will work. Set your scan feed rate. A word of warning here, a feed rate above 30 might result in erratic scans.

Set your Scan Size. If you only need to go half way around something, choose 180 Degree Scan. Otherwise choose 360 Degree Scan.

Set your X Scan Direction. Positive means the initial X move will be in the X+ direction. Set your Y Scan Direction. Positive means the initial Y move will be in the Y+ direction.

Set your Initial Y Move Distance. Try to keep this short. Something between 0.25 and 1 is fine. Whatever you set it to, make sure your probe tip is less than that distance from the front edge of the part you are going to outline scan.

Set your Degrees to Back Off On Stuck Probe. Usually 90 is a good place to leave it.

Measure your probe stylus diameter and enter it next. This is going to be the diameter of the tip that will come in contact with the object being scanned.

Set your Probe Trigger Adjustment. For a startup setting, try 0.005 to 0.007. That isn't going to be perfect, but it will be close. This is the distance the probe tip must travel sideways for the probe to trigger.

Click Run.

**Known issues with the outline scan**

There are limitations to the outline scan. If the scan passes the starting X position twice, the scan ends. This means if you scan rectangles, circles, or shapes that do not drastically deviate from these, it will probably scan.

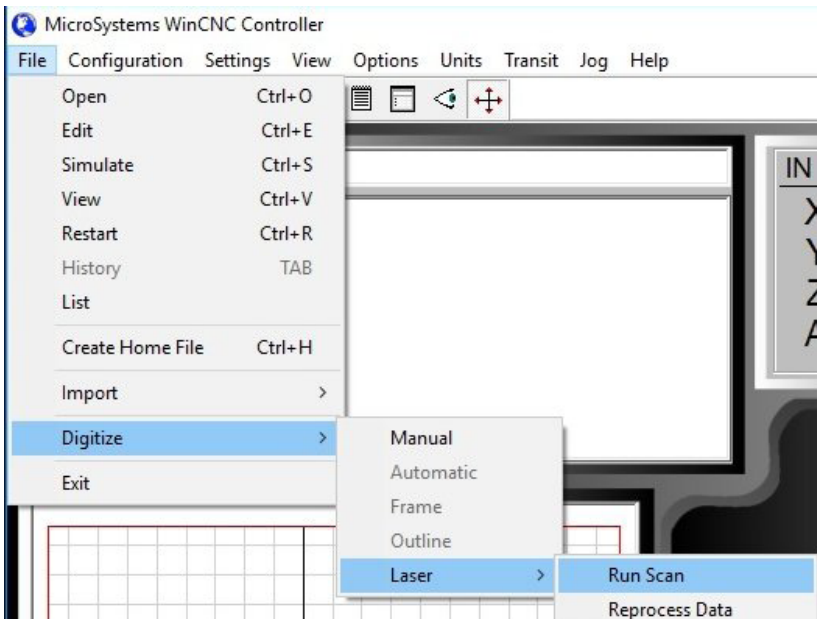
Example of an object that will not scan correctly: a part in the shape of the letter S – the tip will pass the starting X position more than once before completing, thus it ends. You could probably turn that shape sideways and scan it though because it will only pass the starting X position once before coming back to the start/end point.

Another issue: The outline scan works by doing arcs. It arcs from where it is currently sitting and tries to do a circle that would end up back where it started. If it hits something while it is moving, it immediately tries to back off until it clears. If it comes in contact with some other object while backing out, it generates a stuck probe error.

Because the outline scan works in arcs, corners are usually rounded off. There will usually be some editing to do after the DXF is complete.

Keep these things in mind when setting up to do an outline scan.

Laser Scanning/Digitizing



The WinCNC Laser Scan feature supports the Optimet ConoPoint laser scanner. It allows fast, high resolution, non-contact digitizing of 2 dimensional as well as 3 dimensional objects.

Once the Optimet is installed and set up, just click **File > Digitize > Laser > Run Scan** to pop up the Laser Scan Parameters dialog box.

**Laser Scan Parameters dialog**

**X Start:** X starting position for this scan.

**Y Start:** Y starting position for this scan.

**X End:** X ending position for this scan.

**Y End:** Y ending position for this scan.

**X Res.:** X axis resolution for this scan. (DPI)

**Y Res.:** Y axis resolution for this scan. (DPI)

**Feed Rate:** XY feedrate for this scan. If scan is unidirectional the return passage will be at the axes rapid velocity.

**CCD Freq.:** Shows the detectors' exposure time for every measurement (measures per second).

**T Button:** Automatically calculate and set the (T)arget CCD frequency.

**Coarse:** **Fine:** Coarse and Fine combine to show total laser power.  $\text{Coarse} * 64 + \text{Fine} = \text{laser power}$ .

**Lens:** Select the current lens on the Cono-probe

**Show Probe Dialog Button:** Displays the Optimet Probe output dialog box (See next page).

**Directions:** Can be set to Uni-Directional or Bi-Directional. We highly recommend Uni-Directional scanning.

**Type:** Can be Surface or Rotary.

**Height Offset:** Scan zero will be set to this distance from probe Z zero.

**Scan Data File Name ... :** Click the dotted button and browse to the folder where you want the created file to be deposited. Give the file a name and click Save.

In this dialog box, set the X and Y starting and ending positions, the X and Y resolutions and the feedrate for scanning.

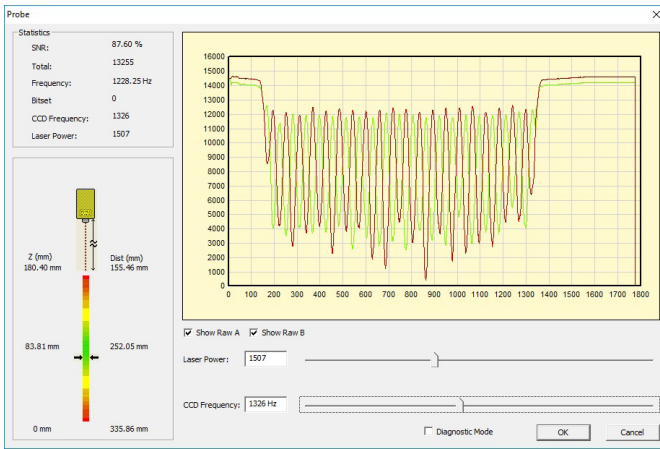
Depending on the model of Optimet scanner being used, you may need to select the lens you're using.

Set the direction of the scan, Uni-Directional usually provides a more accurate scan.

Set the type of scan you wish to do, Surface or Rotary.

To set the Scan Data File name, click the button with 3 dots and browse to the folder you want to scan to, give the file a name, and click Save.

Opimet Probe Dialog



**Statistics Section**

**SNR:** Signal to Noise Ratio. This is the percentage of signal/ (signal+noise).

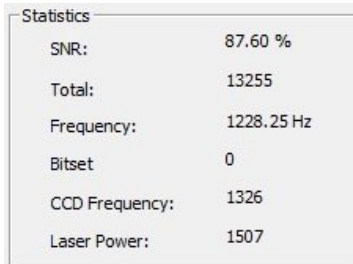
**Total:** Total light coming into the CCD detector. This can be adjusted by raising or lowering the Laser Power.

**Frequency:** The detector's present working frequency.

**Bitset:** Can be 0 (good measurement), 2 (object too close or bad signal), 4(object too far or signal too noisy) or 255 (unknown).

**CCD Frequency:** Shows the detectors' exposure time for every measurement (measures per second).

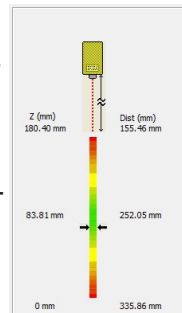
**Laser Power:** Displays the current laser power.

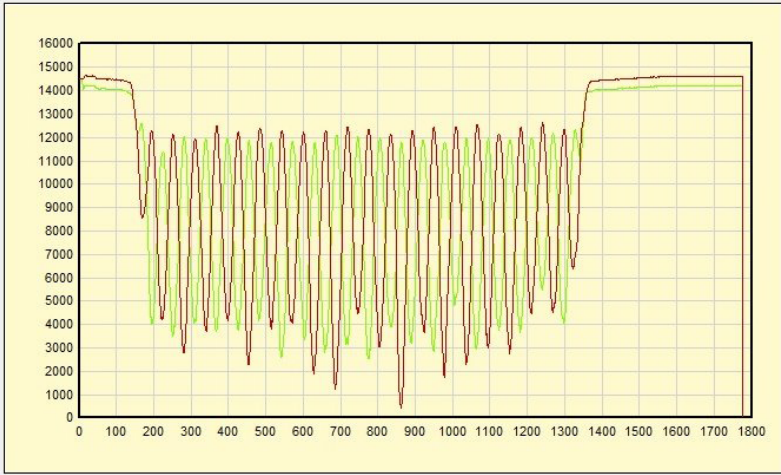


**Vertical Distance Scale**

Measurements on the left side of the color line represent the usable range of the current lens. Measurements on the right side represent the distance from the lens.

In the example here with a 250 mm lens, the total usable scanning distance is 180.4 mm (about 7 inches) and the object is 252.05 mm (about 10 inches) away from the lens.



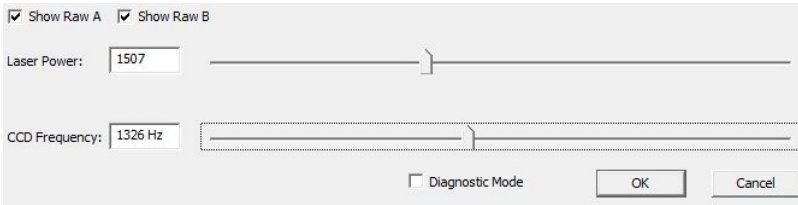


### **Signal Display**

Scattered light reflected off the object enters the sensor's optical system and is detected by two CCD/CMOS detectors inside the sensor. The signal display allows you to analyze and optimize the signal quality.

You can enlarge a section of the signal display graph for easier viewing. To enlarge a section of the graph, hold down the left mouse button and drag the cursor from the top-left to the bottom-right of the section you want to enlarge. To zoom out, hold down the shift button and press the left mouse button while dragging the cursor over the graph from bottom-right to top-left.

Had the sine waves been clipped, either at the top or bottom, that would mean saturation of the sensors. If that happens, adjust the Laser Power down or increase the CCD Frequency.



## Sensor Settings

**Show Raw A:** Display one of the sine waves on the signal display.

**Show Raw B:** Display the other sine wave on the signal display.

**Laser Power:** Display the laser power that is selected either by dragging the slide bar or by typing a new laser power into the text box.

**CCD Frequency:** Display the CCD Frequency that is selected either by dragging the slide bar or by typing a new frequency into the text box.

## Laser Scan Output

Once a scan is completed, the **Laser Scan Output** dialog appears.

**Scan Data File Name:** Use the button with 3 dots to browse to the BIN data file that you just scanned. Type the name and click Save.

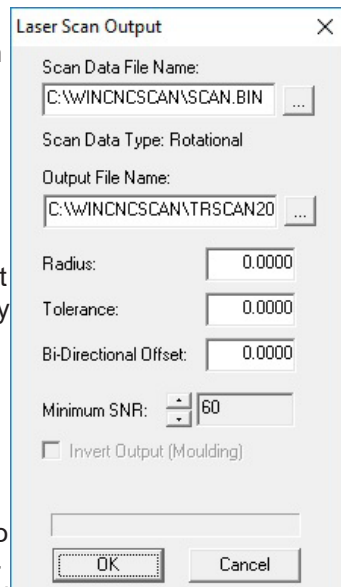
**Scan Data type:** This is the type of scan that was performed for the chosen BIN file. It will automatically populate from the saved scanned file.

**Output File Name:** Use the button with 3 dots to browse to the location where you want to save the STL file. Type the name and click Save.

**Radius:** Distance from the point you set as your Z zero to the center of the rotary axis. Leave this at zero on a surface scan.

**Tolerance:** If the height change from one data point to the next is less than the tolerance, the old point height is used instead of the current height.

**Bi-Directional Offset:** An adjustment to compensate for offset differences on bi-directional scans. Although bi-directional scans are possible, better scans will always be made using uni-directional scans.





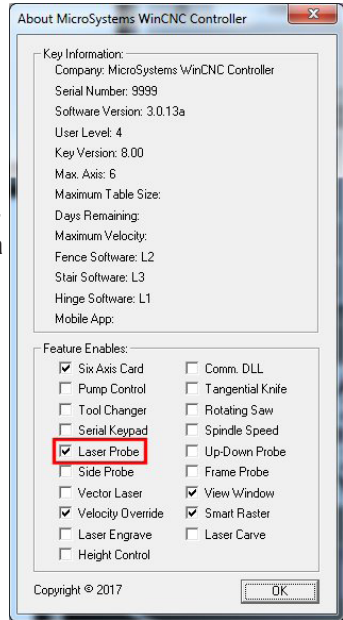
**Minimum SNR:** The minimum signal to noise ratio that will be used to process this data scan. Normally you would want all points, but if a point is in data that has less than about 70%, it is probably a bad point. Running at 60% allows nearly all possible good points to be in the file while eliminating points that might have been created by noise.

This dialog also can be accessed by clicking **File > Digitize > Laser > Reprocess Data** if you ever want to reprocess a saved BIN data file.

### Enabling the Optimet Laser Scanner

To make the Optimet scanner work, you first have to have the feature enabled in your WinCNC security file. Contact your manufacturer or WinCNC.

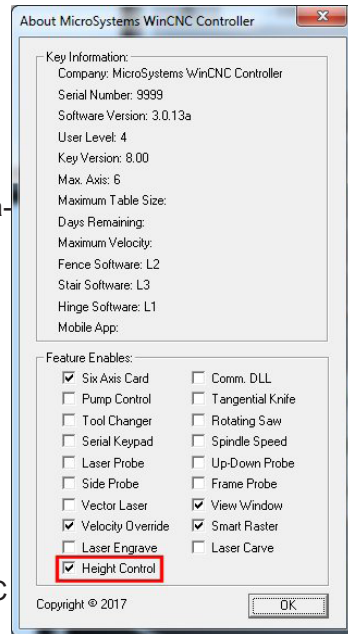
To verify that you have the feature enabled, Click **Help>About**. Once the feature has been enabled you should see a checkmark beside of it in the About box as shown to the right.



## Plasma Torch Height Control

To make THC work, you first have to have the feature enabled in your WinCNC security file. Contact your manufacturer or WinCNC.

To verify that you have the feature enabled, Click **Help>About**. Once the feature has been enabled you should see a check mark beside of it in the About box as shown to the right.



## THC Operation

Plasma torch height operation uses macros that are defined in WinCNC. Those macros are normally called with user defined M Codes (M61 - M66).

## Torch Height Control Window

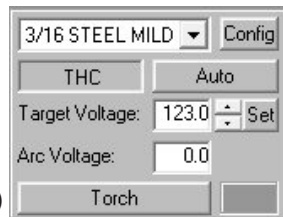
To add the THC window to the WinCNC screen, a line must be added to the CNCScr.ini file. See the Screen Configuration - Display Sections, THCWindow for a description and example of the line to add to the CNCScr.ini file.

**Material Selection Dropdown:** Clicking the drop down arrow allows the user to select from all of the materials defined in the **Configuration** dialog.

**Config Button:** Clicking this button pops up the Configuration dialog.

**THC Activation Button:** This button serves a dual purpose. It allows the user to manually activate the torch height control, and it also displays (is depressed) when THC is active.

**Auto Button:** Another dual purpose button. It allows the user to manually activate the auto-sensing height control (not recommended), and it also displays (is depressed) when auto-sensing mode is active.



**Target Voltage:** The desired voltage to use for the selected material. It can be adjusted in three ways. 1. Click the Config button and set it in the THC Configuration dialog. 2. Temporarily adjust it up/down with the arrow keys. 3. Click the **Set** button and enter the temporary target voltage into the dialog that pops up.

**Set:** Pops up a small dialog box that allows the user to input a tem-

porary target voltage (does not change the setting in the configuration list).

**Arc Voltage:** This text box displays the arc voltage coming in from the plasma unit. WinCNC raises or lowers the torch tip to keep the Arc Voltage at or near the Target Voltage.

**Torch Button:** Turns on (fires) the plasma torch. It also turns the plasma torch off.

## Saved THC Configurations Dialog Box

**Configurations Window (unnamed):** This window holds a list of all of the configured material setups for this machine. The user can select any one of them and view their stored settings. The user can alter any setting and resave this material by clicking on the “Save” button. The user can delete a selected material by clicking on the “Delete” button. The user can also add a new material by clicking on the “Add” button.

**Material Name:** User defined material name. Best to give a short description of the material.

**Torch Amps:** Torch amperage setting can be inserted here. This is for the user’s benefit only.

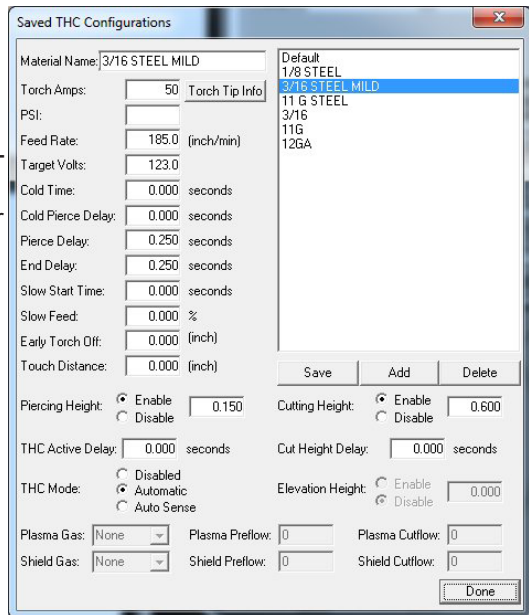
**Torch Tip Info Button:** If there is a .BMP file by the same names as the material located in the WinCNC folder (or folder where WinCNC is located), clicking the Torch Tip Info button will display that image. This would normally be a bitmap image of a page from a plasma unit

manual with torch tip information. It might also contain other recommendations and information as well.

**PSI:** PSI (pounds per square inch) setting can be inserted here. This is for the user’s benefit only.

**Feed Rate:** XY axes feed rate for this material.

**Target Volts:** Target voltage (will also appear in Torch Height control



window).

**Cold Time:** The number of seconds the torch must be inactive to qualify for the alternate “Cold Pierce” delay. If a zero ‘0’ value is specified for the “Cold Time” then the standard pierce delay is always used.

**Cold Pierce Delay:** The number of seconds to use for a “Cold” pierce.

**Pierce Delay:** The number of seconds to pause motion after firing the torch before moving on with the cut.

**End Delay:** The number of seconds to pause motion after shutting the torch off before moving to the next cut.

**Slow Start Time:** The number of seconds that the torch will move at the percentage of full speed declared in “Slow Feed” textbox.

**Slow Feed:** The percentage of full feed rate to use during “Slow Start Time”.

**Early Torch Off:** Distance (in units of measure) before the end of the cut that the torch will be shut off.

**Touch Distance:** Specifies the minimum distance (in units of measure) that the torch must move from the last material touch location before another material touch is executed.

**Add Button:** Click this button to add an unnamed, blank material setup to the Configuration Window.

**Save Button:** Click this button when the user is satisfied with changes made to a selected material in the Configuration Window. This material, along with any changes, will be saved to a file in the WinCNC folder.

**Delete Button:** Click this button to delete a selected material from the Configuration Window. A dialog box will pop up and ask the user to verify deletion. Click “Yes” to delete. Click “No” to cancel deletion.

**Piercing Height:** When enabled, this setting is used by a “Touch” macro that lowers the torch to touch the material, sets a zero for the vertical axis, then raises it to this defined height (in units of measure) before piercing the material. When disabled, “Touch” is ignored and the torch fires from whatever height it is at.

**Cutting Height:** When enabled, the torch will move to the defined height to continue after it pierces.

**Cut Height Delay:** When Cutting Height is enabled, the torch pierces at Piercing Height, then continues moving at pierce height for the number of seconds shown in the Cut Height Delay. It then moves the torch to Cutting Height and continues with the cut.

**THC Active Delay:** Seconds before height control will be allowed to control the THC vertical axis. Horizontal axes can move during this

time.

**THC Mode:** There are 3 modes for torch height control.

1. **Disable:** This will disable height control. The torch will not automatically raise or lower.
2. **Automatic:** Height control is enabled. It constantly matches the incoming arc voltage to the preset target voltage by raising or lowering the torch tip. This is the recommended method of height control.
3. **Auto Sense:** Acquires a new target voltage each time the torch fires.

**Done Button:** Save the information and close the THC Configuration dialog box.

**The following settings are under development and definitions will be added as they are enabled.**

**Plasma Gas:**

**Shield Gas:**

**Elevation Height:**

**Plasma Preflow:**

**Shield Preflow:**

**Plasma Cutflow:**

**Shield Cutflow:**

# **WinCNC Controller**

---

This page is intentionally left blank.

## System Configuration

WINCNC.INI is read on program startup. The pre-configured file is set up to work with minimal change for normal installations. To change configuration use a text editor to edit WINCNC.INI. If the WINCNC.INI is changed while WinCNC is running, you must restart the program to activate your changes.

### **CAUTION!!**

Make sure you have some idea of what you are doing before changing WINCNC.INI. All input and output is controlled by these settings. Improper settings will cause limit switches to not work, wrong axis movement in the wrong directions, and/or other bad and potentially dangerous or damaging incorrect operation.

Please call for help instead of experimenting if you are not very familiar with motion control concepts and computer configuration.

Double-click the WINCNC.INI file from within Windows to open it in a text editor. Change values for parameters as specified below. WINCNC.INI is read every time the program is started. Make sure the file gets saved as ASCII text (without formatting).

## WINCNC.INI Settings

### **Unit Settings**

The default unit of measure is inches. If you want to use this WINCNC.INI setting to set the unit of measure, **it must be the first line in the INI file.** You can also set units by choosing Units from the menu bar and selecting the desired unit of measure.

<b>unit=unit</b>
Unit is the unit of measurement you want to use. <b>IN</b> - sets units to Inches <b>CM</b> - sets units to Centimeters <b>MM</b> - sets units to Millimeters  Example: unit=IN would set up your configuration to use Inches as the unit of measurement.

### **Timer Card Setup**

<b>timertype=#</b>
# must be 7200

## Axis Settings

Each axis for a machine must be configured in the WINCNC.INI file. Axis settings MUST be included in the following order:

**Note:** Values to the right of the equals signs (=) vary by installation. \*\*\*Adding a negative value (-) will make the keys work in reverse direction. Example, K-1 instead of K1.

[Axis Labels]

axischar=XYZ (others may be added here, such as U, B, A W etc)

[X Axis]

axisspec = p0 r818.5111 a400

axisvel = r450 f100 s50 m200 h450

axislo = p2 b6 o1

axishi = p2 b7 o1

[Other axis settings related to the X axis (axismode=, axisadj=, etc.) would follow]

[Y Axis]

axisspec = p1 r818.5111 a400

axisvel = r450 f100 s50 m200 h450

axislo = p3 b0 o0

axishi = p3 b1 o0

[Z Axis]

axisspec = p2 r400 a200

axisvel = r250 f50 s50 m100 h200

axislo = p3 b2 o0

axishi = p3 b3 o0

**NOTE:** Axes must be configured in the order they appear on the axischar= line. In the example above, the X axis would be set up first using the axisspec, Y next, Z next, etc.

<b>axischar=###</b> (# is desired alpha axis label. )
Example: axischar=XYZWUAB (The first axis MUST be X, and the second MUST be Y)



**axispec=P# R# A# F# O# E# T# B# K#**

Configures basic functions for a motor on a machine.

SEE DIAGRAMS IN APPENDIX

**Note:** Settings listed in this section that are associated with “axis-spec=” will be denoted with an asterisk (\*).

**P#** is port number. **IMPORTANT! This has changed from previous WinCNC versions. There is no longer S# and D# assignments. Ports are chosen as 0 - 5. See Appendix.**

**R#** is the actual resolution in steps per unit.

**A#** is the actual acceleration in units per minute per second.

**F#** sets acceleration for feed moves. If no F# is used the A# parameter is used for both rapid (G0 and G53) and feed (G1, G2 & G3) moves.

**O#** changes motor direction, can be 0 or 1. (Default = 1)

**E#** sets the step signal for negative edge triggered drives, can be 0 or 1. (Default = 0) (For Pos edge triggered drives, use E1)

**T#** sets the type of axis. 1 is first horizontal, 2 is second horizontal, 3 is vertical, 4 is rotational.

**B#** is the amount of backlash to compensate for on the axis.

**K#** is used for keyboard mapping to this axis. (If no K value is present, default is K1 for X axis, K2 for Y, K3 for Z, K4 for W)

Ignores Scroll Lock (Normal Default)

- 1 Left/Right Arrow Keys (K1)
- 2 Up/Down Arrow Keys (K2)
- 3 PageUp/PageDown Keys (K3)
- 4 Home/End Keys (K4)

Works Only with Scroll Lock Off

- 5 Left/Right Arrow Keys
- 6 Up/Down Arrow Keys
- 7 PageUp/PageDown Keys
- 8 Home/End Keys

Works Only with Scroll Lock On

- 9 Left/Right Arrow Keys
- 10 Up/Down Arrow Keys
- 11 PageUp/PageDown Keys
- 12 Home/End Keys

Adding a negative value (-) will make the keys work in reverse direction. Example, K-1 instead of K1.

**axisvel=R# F# S# M# H# A# C#**

\*

Sets up velocities for the specified axis.

**R#** sets the rapid (G0) velocity for the axis and also the maximum velocity that this axis can move at feed rate.

**F#** sets the default feed (G1) velocity for the axis.

**S#** sets the velocity for slow keyboard transit.

**M#** sets the velocity for medium keyboard transit.

**H#** sets the velocity for fast keyboard transit.

**A#** specifies an acceleration factor for keyboard transit moves (this value should normally not be more than 1.)

**C#** sets minimum velocity used in velocity matching. (Using this setting improves smoothing and reduces cutting times. The velocity set here will make it so that even while slowing for a change of direction, the axis will not go any slower than this speed (C#). This is often used in conjunction with the G09 setting.)

**axisadj=T# A# V# U# D# I# B# S#**

\*Can be used as a height control while the file is running. Adjusts the position of an axis by: transit keys and input switches. This should not be used when using an automatic height control (see THC=).

**T#** set to 1 to enable transit key control of the specified axis.

**A#** sets the adjust enabled axis. 0=X, 1=Y, 2=Z, etc.

**V#** sets the maximum velocity of the specified axis during adjustment, in units per minute.

**U#** sets the input channel number used to trigger a positive adjustment.

**D#** sets the input channel number used to trigger a negative adjustment.

**I#** is the initial state of axisadj (1=enabled, 0=disabled).

**B#** set the alternate adjustment axis (0=X, 1=Y, 2=Z,...)

**S#** sets the input channel used to switch between the primary and alternate axis.

**axismap=S# E# M#**

\*

Creates a position map for the axis to straighten the path of motion for an axis. Add this line under the "axispec=" line for each axis to be mapped.

**S#** sets the starting position for the map.

**E#** sets the ending position for the map.

**M#** sets the map increment used for this axis.

**altaxispec=P# O# E#** \*

See axispec= documentation. This setting is used for an axis that has two motors, and specifies the step and direction settings for the second motor. Uses the same resolution, acceleration, type, and backlash as axis specified in the previous axispec= line.

**axishi=P# B# O# A# D#** \*

**axislo=P# B# O# A# D#**

**altaxishi=P# B# O# A# D#**

**altaxislo=P# B# O# A# D#**

High and low limits for axis and altaxis.

**P#** is the input port for limit switch, can be 0-3.

**B#** is the bit on the port used with the switch, can be 0-7.

**O#** sets the switch polarity, can be 0 or 1.

**A#** sets the alternate bit.

**D#** is the debounce for the limit switch.

Example: axislo=P0 B2 A4 [bit 2, alternate bit 4]

An **altaxislo** can be enabled using the **L210** command followed by the axis character. Example: **L210Z** enables the **altaxislo** limit for the Z axis.

An **altaxishi** can be enabled using the **L211** command followed by the axis character. Example: **L211B** will enable the altaxishi limit for the B axis.

**accel=S#**

**S#** sets the max velocity increment during acceleration. Can be 0 to 200. The default value is 50.

This setting pertains to the s curve setting of acceleration and deceleration. Normally, using the default value of 50 is recommended.

**ad=L# H# X# Y# A# B#**

Reads incoming analog signal used to monitor spindle current.

**L#** is the low volts for the voltage range.

**H#** is the high volts for the voltage range.

**X#** is the low amperage for the voltage range.

**Y#** is the high amperage for the voltage range.

**A#** is the low alarm in amps.

**B#** is the high alarm in amps.

## anykey=A# P#

By default, the Spacebar is the Pause key and the Escape key is the abort key. By including the anykey= line in the WINCNC.INI file, the user can set WinCNC to read any unassigned key as a pause key or an abort key.

Example: if “anykey=a1” is in the WINCNC.ini file, then when any unassigned key on the keyboard is pressed, the file will abort all operation. If “anykey=p1” is in the WINCNC.ini file, pressing any key on the keyboard will pause all operation.

**A#** enables or disables abort on any unassigned key. 0 is disabled, 1 is enabled.

**P#** enables or disables pause on any unassigned key. 0 is disabled, 1 is enabled.

If both a1 and p1 are included, any unmapped key will be read as an abort, overwriting the pause command.

## arc\_err=#

**#** is the value (in units of measure) of allowable errors in arc specifications. Default value is .01. If you get radius errors from your files increase in .002 steps. If errors persist with larger values the problem is probably in your program. Arc radius errors indicate that the distance from the start point to the center point is not the same as the distance from the end point to the center point.

## arc\_min=#

**#** is the value (in units of measure) for smallest arc radius to be cut as an arc. Arcs with radius smaller than this value will be cut as a G1 move. Default value is .002.

## arcs=M# A#

**M#** - M# - Modal G2/G3. Use 1 for enable and 0 for disable. The default is disabled.

**A#** G2/G3 IJ values. Use 1 for absolute and 0 for incremental. The default is incremental.

**arctype=I# M# O#**

Allows configuring WinCNC to break up arcs into line segments.  
**I#** - the length of the segment to interpolate.  
**M#** - if set to 1 makes G2/G3 commands modal.  
**O#** - can be used to skip the “no step” arc check during G2/G3 commands.

**atc1=H# O# Z# A# M#**

**atc2=H# O# Z# A# M#**

Configures Automatic Tool Changers.  
**H#** is the height at which the tool is unloaded.  
**O#** is the offset to the tool sensor.  
**Z#** is the height of the bottom of the tool changing spindle.  
**A#** is the axis number for the tool changing spindle. X=0, Y=1, etc.  
**M#** can be set to 0 or 1. Mode 0 stores tool measures in an internal settings file, not accessible from the tool library. Mode 1 specifies the tool measures are stored in the tool library (see M37 and L110 documentation).

**autoarcfeed=F# S# L# M# A#**

**F#** sets the factor for the maximum force that is allowed. Increase to increase arc velocity. The best value should be found somewhere between 10 and 60, but this is dependent upon the machine. The default value is 30.  
**S#** sets the minimum arc radius. Arcs with a smaller radius will use the minimum velocity. The default value is 0.05".  
**L#** sets the maximum arc radius. Arcs with a larger radius will not reduce velocity. The default value is 10".  
**M#** sets the minimum velocity to be used for any arc. The default value is 20 inches per minute.  
**A#** sets the minimum arc angle for full velocity reduction. If the angle of an arc is less than the specified angle then the velocity will be reduced by a smaller amount.

**NOTE:** This is useful to prevent unnecessary slowing down for short arc moves. The default value is 0 degrees, and the suggested values are from 5 to 15 degrees.

**autoaux=C# I# S# E# T#**

**C#** is the auxiliary output channel number.

**I#** is the initial state of the autoaux mode (0=off, 1=on).

**S#** sets the amount of dwell time in seconds used when the autoauxout channel is set to on.

**E#** sets the amount of dwell time in seconds used when the autoauxout channel is set to off.

**T#** T0 turns specified output on during G1/G2/G3 moves and off during G0/G53 moves, using the M11 and M12 commands internally. T1 uses the M13 and M14 commands internally, which prevents WinCNC from coming to a stop when changing the output channel state.

This feature allows the user to set up an auxiliary output that will turn on during G1/G2/G3 moves, and will turn off for all other moves, only if the "autoauxout" mode is set to on. (Meaning, the initial state must = 1). The "autoauxout" mode can be enabled using the L34 command and disabled using the L35 command.

**\*\*NOTE: When using height control (such as the THC), you cannot use the autoaux=.**

**autodir=c:\wincnc\auto**

Specifies the directory where files to be automatically ran are copied.

**autofdir=c:\wincnc\done**

Specifies the directory where files are copied after autorun.

**autogas=T# C# B# N# G# D#**

T# is type.

T1 is for Hyperformance Plasma HPR models.

T2 is for the simpler PowerMax 65/85/105/125 units that allow serial configuration of only the amperage and air pressure settings.

Both T1 and T2 require the serial communication option.

C# is COM port number.

B# is baud rate. 19200 for type 1.

N# is N2 mix set.

G# is G2 mix set.

D# is serial response delay

**automode=#**

The autorun mode 0 is disabled, mode 1 will copy name and path to command bar, mode 2 will automatically execute the found file.

**auxin=C# P# B# O# D# F# L# I# M"<message>" A"<axes>"**

Configures auxiliary inputs.

**C#** is the channel number, 1-32. Numbers should not be skipped.\*

**P#** is the port number, refers to a port specified in a portin spec.

**B#** is the bit number, can be 0-7.

**O#** changes the input polarity, can be 0 or 1.

**D#** specifies debounce. When the input is a switch, debounce should be used. D10-50 is a typical value.

**F#** specifies the function type of the auxin. Can be set to the same function types as **enab**, or **signal** below. When set to 0, no special function types are enabled. (If the same F command is used for different channels, there will be a conflict.)

F1 - Enable

F100 - Signal Abort

F101 - Signal Pause/Continue

F102 - Signal Pause

F103 - Signal Continue

F110 - Signal Start Cycle

F201 - Signal User Spec 1, use L201 command

F202 - Signal User Spec 2, use L202 command

**L#** can be 201 or 202. If configured as an enable, this causes the input channel to be used as an enable when the machine is running, and as a type 201 or 202 signal when not running.

**I#** is the initial active/inactive state of an enable or signal.

0=active, 1=inactive.

**M"<message>"** denotes a custom error message. The message must be enclosed in double quotes - "".

**A"<axes>"** requires homing of the specified axes if this auxiliary input is used as an enable (F1) and the enable is triggered during motion.

**Note\*** auxin channel 1 is not the same as auxout channel 1. It is normal to start both auxin channels and auxout channels at 1 and assign the numbers sequentially as needed.

**auxout=C# P# B# O# S# E# T# X# W# R#**

Configures auxiliary outputs.

**C#** is the channel number, 1-32. Numbers should not be skipped.\*

**P#** is the port number, refers to a port specified in a portout spec.

**B#** is the bit number, can be 0-7.

**O#** changes the output polarity, can be 0 or 1.

**S#** set output state on WinCNC startup, can be 0 or 1.

**E#** set output state on WinCNC exit, can be 0 or 1.

**X#** set output state on abort, can be 0 or 1.

**W#** set output state on pause, can be 0 or 1.

**T#** sets the type of auxout. # can be 2 or 3. T2 sets the auxout channel to be a motion only auxout. This will result in the auxout only working while the machine is moving and will turn off when the machine stops. To use this feature you must also have the S# set to 1 to start up with the software. You can then turn it off and on using the M11 and M12 commands. You must have it enabled to work correctly during motion. The T2 auxouts are available on CN1 and CN2. Auxouts on additional I/O cards cannot use this setting. T3 sets the auxout to a programmed velocity percentage auxout type. T3 auxouts are on when the machine velocity is above the specified percentage (based on the auxoutvel=, see below) and off when the velocity is below the specified percentage. **R#** is used to specify whether the output should be turned on or off during a file restart. 0=off, 1=on. When a R value is used, the auxout state will be set by simulating the skipped portion of the file prior to restarting.

**Note\*** auxin channel 1 is not the same as auxout channel 1. It is normal to start both auxin channels and auxout channels at 1 and assign the numbers sequentially as needed.

**auxoutvel=#**

**#** is the velocity percentage to use (0-100). If the machine velocity, during motion, drops below the specified percentage of the programmed velocity then the auxout signal will be on, otherwise, the auxout signal will be off. (Used with the T3 setting of auxout=)



**backlash=B# S# F#**

**B#** sets initial backlash compensation to on or off, can be 0 or 1.

**S#** sets the smoothing factor for backlash compensation moves.

**F#** sets the “slow down” feed rate setting to eliminate “clicking” during arc backlash compensation moves.

**Example:** backlash=b1 f100 means enable backlash, slowdown feed rate of 100 units per minute.

**bitmapdir=c:\wincnc\bitmaps**

Specifies the directory where bitmaps used with the splash screen, background, and screen buttons are kept.

**bmpeditor=<filename>**

Filename is the full path and filename of any editor that will accept file names as a command line parameter.

Example: bmpeditor=paint.exe

**bmpend=<command>**

**bmpstart=<command>**

Commands to be automatically added at the beginning and end of each bitmap file.

Example: bmpstart=m11c1 AND bmpend=m12c1

or bmpstart=m98 start.tap AND bmpend=m98 end.tap

**bmppoffset=L# H# S# F#**

When using this method an offset between the low and high value is selected based on the speed of the raster axis. If a speed below the low velocity is used, then the low offset is used. If a speed above the high velocity is used, then the high offset is used.

**L#** is the low offset

**H#** is the high offset

**S#** is the low velocity

**F#** is the high velocity

**How to set bmppoffset=**

Set **L** and **S** to 0.

Set **F** to the highest engraving speed.

Set **H** by this measure:

First, set **H** to 0.

Run the laser test file (offset300.bmp) with the feedrate set at the **F#** value.

When the file finishes, Measure the Error – seen below:

$H = \text{Resolution of the axis} * \text{Measured Error}$

Example: Resolution = 2000, measured error = 0.025"

$2000 * 0.025 = 50$  In this case, set **H** to 50

You should run the test again and if it is still off, either add or subtract your new measured error from the current value.

Example 1: Error is now 0.050". We went to the wrong side. Set **H** to -50 and run the test again.

Example 2: New error is 0.008".  $2000 * 0.008 = 16$ . **H** = (current **H** + or – 16), depending on which side of the line it is on.

**How to measure the error:**

Download the file, offset300.bmp (<http://www.wincnc.net/webfiles/Uploads/offset300.bmp>), and run it as a raster file on your laser machine. If the BMPOFFSET is set correctly, there will be 3 perfect vertical thin lines. Any error will show up as a double line instead of a single one. Use a caliper to measure the distance between the lines. This is the Measured Error.

**btnoptions=#**

This setting enables/disables the Options->Buttons menu option.  
# can be one of the following:  
0 = disabled  
1 = enabled (default)

**cmdabort=<command>**

<command> is the command that will run when a file is aborted.  
Example: cmdabort=M12C1

**cmdend=<command>**

<command> is the command that will run at the end of every file.  
Example: cmdend=M12C1

**cmderror=<command>**

Can be used to assign a separate abort file/command to run when an abort is triggered by a parse error. This separate abort/file command is only available when the "errorabort=1" setting is used. If "errorabort=1" is used and there is no "cmderror=" entry then the standard "cmdabort=" file/command will run when an abort is triggered by a parse error.

**cmdrestart=<command>**

When specified, the cmdrestart line will run at the end of a file restart, before the remainder of the file is run.

**cmdrestartfeed=<command>****cmdrestartrapid=<command>**

Allows the insertion of commands prior to motion when using the file restart. These commands are only inserted if the first move at the restart point is a feed or rapid move. This allows restart setup to ensure that all device states are correct for cutting or non-cutting moves.

**cmdstart=<command>**

<command> is the command that will run at the start of every file.  
Example: cmdstart=M11C1

**\*NOTE: These settings can also be used to call a subroutine to use multiple commands**

**commatodec=#**

Allows G-Code files that use a comma ',' instead of a period '.' to denote a numeric decimal value. # can be 1 or 0. Default is 0.

**comchars=###**

### represents characters to use to designate comments in a G-Code file. When one of the listed characters is found at the beginning of a line, WinCNC skips the entire line. The characters "[(%" are regarded as comments by WinCNC by default.

**Warning: Any character listed here will be considered as a comment character, therefore everything that follows that character on any line will be ignored.**

**comdll=<path>**

<path> is the path including filename of the communications DLL file "WINCNC.DLL".

**contdelay=#**

Setting this value causes a # second delay when continuing from a paused state.

**cutcomp=T# D#**

Sets up cutter compensation to account for bit sharpening and wear.

**T#** is the type (0=does not insert entry move - default, 1=automatically insert entry move).

**D#** a negative number here will invert G41 and G42 commands left and right compensation.

**ctrlf#=#**

Used to assign macros, commands, or subprograms to the keyboard using CTRL+Function Key presses. The settings are, "CTRLF1=", "CTRLF2=", etc up to F8.

# can be 1 through 8.

**Example:** Adding "CTRLF1=G28" to the WINCNC.INI would configure WinCNC to run a G28 home command when the CTRL and F1 keys are pressed together.

**da=T# A# V# I# C# R#**

**T#** is the type of D/A port used. Can be 0 (parallel port) or 1 (PCI daughter board) or 2 (serial port).

**C#** is the COM port number and is only used when T is set to 2.

**A#** is the parallel port address and is only used when T is set to 0.

**V#** is the maximum voltage - generally 5 or 10 volts.

**I#** is the initial output voltage when WinCNC starts. It can be set from 0 to the maximum set in V#.

**R#** if R is used, WinCNC refreshes the spindle speed every # milliseconds. For instance, if set to R250, spindle speed would refresh every 250 milliseconds or 1/4 second.

**drillbank=C# N#**

Configures a multiple drill gang associating auxouts with the associated drill.

**C#** is the auxout channel, first one that lowers a drill.

**NOTE: AuxOut channels need to be in numerical order.**

**N#** is the number of channels.

**See Also the M11 command.**

**editor=<filename>**

Filename is the full path and filename of any editor that will accept file names as a command line parameter.

**Example:** editor=write.exe

**enab=C# M"message" T# L# I# A"<axes>"**

**C#** specifies the input channel to be used for the enable.

**M"message"** denotes a custom error message. The message must be enclosed in double quotes - "".

**T#** auxin enable type.

1 is default and will abort from any command when the enable switch is triggered.

2 will abort only from commands that would result in machine motion when the enable switch is triggered.

3 will pause motion with no abort when the enable switch is triggered. Once the switch is released, motion can be resumed by hitting the Enter key.

4 allows non-motion commands and movement with the transit keys but aborts from any commands that would result in motion when the enable switch is triggered.

**L#** can be 201 or 202. If configured as an enable, this causes the input channel to be used as an enable when the machine is running, and as a type 201 or 202 signal when the machine is not running.

**I#** is the initial state (1=active, 0=inactive).

**A"<axes>"** requires homing of the specified axes if this auxiliary input is used as an enable (F1) and the enable is triggered during motion.

Used to set up emergency stops. First set up an auxin line for input. Then specify the enable to use the input channel as an enable switch.

**errorabort=#**

Can be used to set WinCNC to handle parse and command errors as user aborts. Setting # to 1 will cause parse and command errors to be treated as user aborts, triggering the "cmdabort=" command to run. Setting # to 0 (default) will cause parse and command errors to be handled as in previous versions.

**esc\_step=#**

# is the number of steps to move after ESC key is pushed.

**exaltaxislo=C# P# A# B# O#** \***exaltaxishi=C# P# A# B# O#** \*\*

Used to configure up to 10 alternate input pins that can be used as alternate limit switches.

**C#** is the alternate limit channel number. There are up to 10 low and 10 high channels available

**P#** is the switch port number.

**A#** is the alternate bit.

**B#** is the switch bit number.

**O#** sets the polarity for the switch (0 or 1).

\*Exaltaxislo references L210.1 (see L210.1 in L codes section)

\*\*Exaltaxishi references L211.1 (see L211.1 in L codes section)

**filetype=\*.tap;\*.nc**

Specifies file extensions that are recognized by WinCNC as G-Code files.

**forcelim=X# Y#**

Sets starting XY position for size limited machines.

Default is (-1,-1) from origin, or from low soft limits if used.

I.E. X-1 Y-1

**freq=L#H#**

For type 3 spindle speed control. Method of spindle speed control which uses CN2 pin 11 to generate a TTL square wave signal. Many VFD spindle controls support this method of commanding spindle speed in addition to the other supported analog voltage and digital preset speed control methods.

**L#** - Lowest frequency to be used.

**H#** - Highest frequency to be used.

Example:

Spindle=t3 r24000 i0 [Frequency output/24000 RPM maximum/0 RPM initial default]

Freq=i0 h100000 [0 Hz minimum/100 kHz maximum]

\*Note that the minimum frequency that can be generated is 65 Hz. Any speed command requiring a lower frequency will result in 0 RPM speed at the VFD.

\*Note that the maximum frequency that can be generated is 2 MHz. Any speed command requiring a higher frequency will result in maximum RPM speed at the VFD.

**fslock=#**

Adding "fslock=1" to the WINCNC.INI will disable the feed rate, tolerance, accel %, stylus diameter, trigger adjustment, and movement to clear stuck probe for the framescan.

**This is for FrameScan only!**

**g08=E# L# N# T#**

Detects arcs or curves in GCode files where only G1 linear cutting moves are used.

**E#** specifies if the G08 feature is initially enabled or disabled. 1 for enabled, 0 for disabled. Default is 0.

**L#** specifies the maximum length of a line that could be considered a segment of an arc or curve. If a line is encountered that is longer than this then no curve is assumed. Default is 0.05".

**N#** specifies the maximum number of GCode lines in the look-ahead to inspect for arc/curve detection. Default is 10 lines.

**T#** specifies the angle tolerance. If a curve has been detected and the angle then changes by more than the specified amount a new curve is assumed. Default is 5 degrees.



**g09=S# A#**

Used to set the smoothing factor. When the XY direction of motion changes, this setting can reduce the “slowdown” to improve the smoothness of motion. Increasing the S# decreases the slowdown of the machine when turning corners.

**S#** - smoothing factor to be used. (not a specific unit of measure)

**A#** - used to change the angle used in the velocity matching algorithm. The A# value represents an angle. If a small angle, such as say 20, is specified, the G09=# will not apply to any angle smaller than 20.

\*\*A minimum velocity can be set for each axis by adding a C# to the Axisvel= line for that axis. Default is 0.

**g2modal=#**

# can be 0 or 1, if 0 G2 and G3 commands are non-modal, if 1 G2 and G3 commands are modal. The default is non-modal.

**g28altmove=X# Y# Z# W#**

Used to square a machine with two motors on the gantry. The character to be used will correspond with the axis having the second gantry. The # represents the physical adjustment needed to square the gantry.

**g28check=#**

When enabled, if new absolute home positions are set using G28 or L28, they are checked against the previous positions and the user is prompted to accept or decline the new positions. (0=disabled/default, 1=enabled)

**g28checktol=X# Y# Z#**

# sets the tolerance, changes of less than the tolerance are ignored.

**g28coord=X# Y# Z# W#**

Specifies a position after a G28, not used if all positions are zero.

**g28\_lim\_cnt=#**

# sets the debounce for limit switches while homing. The standard “LIM\_CNT=” is used at all other times.

**g28move=X# Y# Z# W# R# F# T# A# M# L#**

**XYZW#** specifies the distance to move from each limit switch at the end of the homing cycle.

**R#** the positive distance to move off the limit switch after the first contact.

**F#** is the approach speed to use for the initial contact.

**T#** is the approach speed to use for second contact. (Default 10)

**A#** is used to specify that an alternate limit switch is to be used.

**M#** specifies that each axis will home individually.

**L#** maximum number of units to move.

**g28req=l# T#**

**l#** - '1' requires homing of all axis on WinCNC startup and '0' does not require homing on startup.

**T#** - '1' disables transit until successful homing and '0' does not disable transit.

**g37=X# Y# l# J# D# F#**

Configures tool measure switches.

**X#** is the X position of the tool measure switch.

**Y#** is the Y position of the tool measure switch.

**l#** is the X position of the second tool measure switch.

**J#** is the Y position of the second tool measure switch.

**D#** is the rapid descent distance.

**F#** is the feed rate for the remaining descent distance after rapid rate.

**g54=X# Y# T# R# O# M#**  
**g55=X# Y# T# R# O# M#**  
**g56=X# Y# T# R# O# M#**  
**g57=X# Y# T# R# O# M#**  
**g58=X# Y# T# R# O# M#**  
**g59=X# Y# T# R# O# M#**

Preconfigured workspace values (default is G54).

**X#** is the workspace shift value for the X axis.

**Y#** is the workspace shift value for the Y axis.

**T#** is the type of workspace shift. Use 0 for vertical head swapping, use 3 for rotating spindle, and use 5 for spindle with two points of rotation.

**R#** is the radius of rotation used for type 3.

**O#** is the offset value, which is used for correcting the 0 degrees position. The 0 degrees position should point toward the positive X direction.

**M#** is used to determine if axes will move to new positions in the new coordinate system when switching between modes. (0= do not move , 1=move to new positions, 2=lift vertical heads then move to new position)

**g81set=D# R# L#**

Configures parameters used in drill cycles.

**D#** is the dwell time for the drill in seconds.

**R#** sets the rapid drill distance from the bottom of the hold for the peck cycle before switching to feed mode.

**L#** is the lift distance above the top of the hole for the bit between each peck.

**g81dwell=#**

**#** is the value (in seconds) of the desired dwell at the end of the plunge move. The default value is 0.

**g84set=A# P# O#**

**A#** sets the axis for tap rotation. (0=X, 1=Y, etc.) The default is axis 3.

**P#** sets the default pitch for tapping.

**O#** can be set to 1 or -1 to invert the rotation direction. The default is O1.

**helpfile**=<path>

<path> - directory for the manufacturers help file.

**hibound**=X# Y# Z# W#

Boundaries for the X, Y, Z, and W axes that take into account work-space offsets.

**hilim**=X# Y# Z# W#

Absolute position soft limits for the X, Y, Z, and W axes that do NOT take into account workspace offsets.

**holddir**=#

# can be 0 or 1. The default mode is 0, (starting with version 3.0.14d, 1 became default) where the direction signal changes immediately after a step pulse (1us-3us) to preset the direction signal for the upcoming step pulse. The new mode is 1, where the direction signal does not change for one additional step pulse delay and then an additional step pulse delay occurs before the upcoming step pulse is generated.

An example of drivers that require this would be the Leadshine MX3660 and MX4660 and the Applied Motion Product 3540.

**homedir**=c:\wincnc\homes

Specifies the directory where created home files are written.

**keypad**=C# B# P# S#

Configures keypads for use with WinCNC.

**C#** is the number of the com port used with the keypad.

**B#** is the baud rate used with the specified port. Should be 9600.

**P#** is the type of keypad used. Type 1 is the standard serial keypad. Type 2 is the Q-Term keypad with LCD display. Type 3 is the 2Tech keypad with LCD display and optional emergency stop.

**S#** is the sorting type and can be one of the following values:

- 1 = sort alphabetically, ascending
- 2 = sort by file size, ascending
- 3 = sort by modification date, ascending
- 4 = sort by creation date, ascending
- 1 = sort alphabetically, descending
- 2 = sort by file size, descending
- 3 = sort by modification date, descending
- 4 = sort by creation date, descending

**keypaddir=c:\wincnc\keypad**

Specifies the directory where files that can be selected from a supported LCD keypad are stored.

**knifeangle=A# M# X# Y#**

**A#** - # specifies the angle of the knife blade when the rotary axis is at the zero degree position. This value should be between 89 and -89.

**M#** - angled knife mode. The angled knife only cuts lines, not arcs. The mode can be set to cut only in one direction or to the destination and back to the origin (thus cutting out a v shaped block). M0 (default) moves in one direction and stops. M1 cuts to destination, rotates the knife 180° and returns to the origin.

**X#** - Specifies the blade center offset along the X axis at zero degrees.

**Y#** - Specifies the blade center offset along the Y axis at zero degrees.

Works with tangential knife angled blade.

**knifemode=T# A# L# Z# P# N# C# D# M# R# V#**

**T#** type of knife. T0 = normal knife function. T1= target mode. (Beginning of XY move, a target position is set internally. During the XY move, the knife axis will move into position as quickly as possible based on step velocity of XY. This allows the rotating knife to be used without affecting speed and quality of XY axis motion.)

**A#** is the lift angle. If the angle of change is greater than this value then the head will lift before rotating.

**L#** is the XY distance where the knife rotation is blended at the beginning and ending of each line segment. This setting can be used to smooth corners.

**Z#** the lift position used to raise the knife above the material.

**P#** is the lower position used to lower knife into material for cutting.

**N#** is the rotational axis number, 0 is X, 1 is Y, etc.

**C#** specifies auxout channel used to raise and lower knife. If not used, then Z axis is used to raise and lower knife.

**D#** is the dwell time (in seconds). This is used with C#. Denotes the specific amount of time needed to raise or lower the knife.

**M#** can be 1 for automatic up/down positioning, or 0 (default) for manual up/down positioning.

**R#** can be 1 for auto lifting during G2/G3 moves (default), or 0 to disable the auto lift.

**V#** specifies the vertical axis for the tangential knife. Default is 2.

**laser=P# D# S# M# O# A# B# R#**

**P#** is the initial power scale - percentage.

**D#** is the laser power duty cycle.

**S#** sets smart engraving off/on (0/1).

**M#** is the margin for acceleration on either side.

**O#** is the step offset used to align the bitmap rows.

**A#** sets the primary (raster) axis number. 0=X, 1=Y, etc.

**B#** sets the secondary (row) axis number. 0=X, 1=Y, etc.

**R#** sets the default radius of the engraving surface when a rotary axis is used.

**laserscan=S# M# U# W# A# T# R# V# L# F#** (for scanning 3-D objects)

**S#** sets the step pin/bit used to trigger laser scanning (0-5).

**M#** sets the margin in units. The margin is used at the start and end of each pass to allow room for machine acceleration. A value of 0 enables the auto-margin feature.

**U#** set to 1 or greater to increase the auto-calculate margin size.

**W#** sets the number of “warm-up” readings used at the beginning and end of each pass to allow the Optimet system to collect more accurate data. Recommended 3-5.

**A#** is used to specify the axis 0=X axis, 1=Y axis.

**T#** is the type. 1 is Ethernet, 0 is LPT.

**R#** is the rotary axis number if applicable.

**V#** V1 (default) enables parameter validation for the feed rate, resolution, and CCD frequency values. V0 disables parameter validation.

**L#** When L# is set to 1, the laser power is reduced to zero at the end of a scan. L0 (default) disables this mode.

**F#** F0 disables post-scan data filtering and smoothing. F1 (default) enables data filtering and smoothing.

**lim\_buf=S# F#**

**S#** buffer size for slow moves (M28/M29 and G28 final touch). Default is 6.

**F#** buffer size for fast moves (G28 initial approach). Default is 60.

**lim\_cnt=#**

**#** sets the debounce for limit switches. (MAXIMUM is 30. If set higher, it will default back to 30)

**lim\_mode=#**

**0** = same debounce for on/off, any limit switch aborts.

**1** = same debounce for on/off, only limit switch in the direction of motion causes an abort.

**2** = less debounce for change to off, any limit switch aborts.

**3** = less debounce for change to off, only limit switch in the direction of motion causes an abort.

**lim\_step=#**

# is the number of steps to compress limits after switch hit. Some machines may be damaged if stopped suddenly at higher speeds. Abort cushions provide a method of decelerating quickly but not instantly when a limit is hit or the escape button is pushed. lim\_step specifies the number of steps to continue after a limit is hit. This number must be calculated from the distance your switches may be compressed without damage and the lowest axis resolution of your machine. For instance if your X and Y resolution are 1018.592, your Z and W resolutions are 2000 and your limit switches may be compressed .1" then you would set lim\_step as follows:  
lim\_step=1018.592\*.1=100 steps  
esc\_step may be set to any value that results in a quick stop without moving too far. A 200 setting is suggested.

**lobound=X# Y# Z# W#**

Boundaries for the X, Y, Z, W axes that take into account workspace offsets. Recommend using Z0 & W0 only so that the vertical head will not go below the tool measure of Z0.

**logdir=c:\wincnc\log**

Specifies the directory where the log file is written.

**logfile=wcnc.csv**

Specifies the name of the log file.

**lolim=X# Y# Z# W#**

Absolute position soft limits for the X, Y, Z, W axes that do NOT take into account workspace offsets.

**mapping=!# R#**

!# is the interpolation increment used for all axes.  
R# adding R1 to the 'mapping=' line will enable axis mapping to be active during rapid moves.



**maxparam=#**

# Maximum parameter setting to prevent problems that can occur from accidentally inputting an extremely large position parameter into one of the GCode commands. This maximum parameter limit is applied in both the positive and negative directions. The default maximum parameter is 99,999.

**maxstepv=#**

# is the maximum number of pulses per second per axis. The default value is 30000, and can safely be increased to 50000 using the 7200 timer type.

**maxtrans=#**

# is the distance, in units, of the largest possible transit move. Setting this value to a distance equal to or slightly larger than the largest table dimension on a machine will make the transit moves respond more quickly.

**numkp=#**

The # value can be set to 0 or 1 to disable or enable the + and - keys of the keyboard numeric keypad for controlling spindle speed, laser power, or THC target voltage. The default setting for this feature is enabled. Disabling this feature allows entering + and - characters into the WinCNC command line.

**nmode=#**

Sets the initial G90/G91 mode. By default the initial mode is saved when WinCNC exits and restored when WinCNC starts.  
# can be 90 or 91

**override=E# L# H# D# U#**

**E#** enables or disables the feed rate override. Can be set to 0 or 1. 0 is disabled, 1 is enabled. The default is enabled.  
**L#** sets the low limitation of the override percentage. Can be 1 to 200, but must be lower than the high value.  
**H#** sets the high limitation for the override percentage. Can be 1 to 200, but must be higher than the low value.  
**D#** - Aux input channel for signal to decrease feed rate override.  
**U#** - Aux input channel for signal to increase feed rate override.

**portin=#, \***  
**portout=#, \***

# is the number of the port, can be 0-7.  
\* is the hexadecimal memory address to the port.

**PCI48H=A# B#**  
**PCI24H=A#**

**A#** - Sets IO for ports A1, B1, and C1. (PCI48H and PCI24H)  
**B#** - Sets IO for ports A2, B2, and C2. (PCI48H only)  
WinCNC IO port numbers are automatically assigned to ports 8-12.  
A# or B# can be set to 0-15 from the following table.

- 0 - A=out/B=out/CU=out/CL=out
- 1 - A=out/B=out/CU=out/CL=in
- 2 - A=out/B=in/CU=out/CL=out
- 3 - A=out/B=in/CU=out/CL=in
- 4 - A=out/B=out/CU=in/CL=out
- 5 - A=out/B=out/CU=in/CL=in
- 6 - A=out/B=in/CU=in/CL=out
- 7 - A=out/B=in/CU=in/CL=in
- 8 - A=in/B=out/CU=out/CL=out
- 9 - A=in/B=out/CU=out/CL=in
- 10 - A=in/B=in/CU=out/CL=out
- 11 - A=in/B=in/CU=out/CL=in
- 12 - A=in/B=out/CU=in/CL=out
- 13 - A=in/B=out/CU=in/CL=in
- 14 - A=in/B=in/CU=in/CL=out
- 15 - A=in/B=in/CU=in/CL=in

**PCI48HREG=#**  
**PCI24HREG=#**

Allows WinCNC to support variations of the PCI24H and PCI48H add-on IO cards. Adding "PCI24HREG=#" or "PCI48HREG=#" allows the user to change the register on the card where WinCNC can access the IO pins and card configuration. Older revision PCI48H cards use the default register 2. The currently available new revision of the PCI48H requires using register 3.

**positions=P# H# T# W#**

**P#** - maximum number of stored positions.

**H#** - maximum number of stored home positions.

**T#** - maximum number of stored tool positions.

**W#** - can be used to enable/disable a pop-up warning when using the "Get Pos" button on the Position, Home, and Tool positions dialog boxes. W1 enables this warning and W0 (default) disables it.

**pumpcont=T# L# H# A# B# I# R# (Configures pump control.)**

**T#** sets the pump control type. The available types are:

T0 pulse width 200hz

T1 pulse width with auto reverse 200hz

T2 pulse width without auto reverse, double scaled 2hz

T3 frequency mode

**L#** sets the low value of the output range in units per minute.

**H#** sets the high value of the output range in units per minute.

**A#** sets the low value of the frequency range in pulses per second.

**B#** sets the high value of the frequency range in pulses per sec.

**I#** is the initial value of the pump control output (0-99).

**R#** is the initial value of the pump reverse output (0-15).

**pwm=F# M# L# H#**

**F#** is the frequency at which the pwm signal is transmitted. Default is F75.

**M#** is the maximum duty cycle percentage to use. Default is 100.

**L#** is the minimum duty cycle in percentage to use.

**H#** is the maximum duty cycle percentage to use. Default is 100. It is the same as M# and they can be used interchangeably.

**pwm da=V# I# L# H# F#**

**V#** sets the maximum D/A input voltage.

**I#** sets the initial D/A output voltage.

**L#** sets the low PWM duty cycle. Defaults to 0.

**H#** sets the high PWM duty cycle. Defaults to 100.

**F#** sets the PWM frequency. Defaults to 2000 (2kHz).

**rapid\_lock=#**

**#** can be: 0 - Unlocked status and can change G00 feed rates, or

1 - Locked status and can not change G00 feed rates.

The default setting for this feature if not specified in the INI file is 1, the locked status.

**rapidjog=#**

# can be 0 or 1. If 0, jog moves us G1 speed (this is default). If set to 1, jog moves us G0 speed.

**restart=V# S# M# W#**

**V#** is for vertical axis position restore. **V0** skips vertical axis position restore, **V1** restores vertical axis positions (default)

**S# S1** skips to next feed move. **S0** does not skip.

**M#** specifies the maximum number of files stored in the file restart dialog list.

**W# W1** forces the program to wait for to hit ENTER. **W0** will skip "wait for user".

**sawmode=T# A# R# O# Z#**

**T#** - type of saw. This setting must be 1. Reserved for future use.

**A#** - rotational axis number. 0 is X, 1 is Y, etc.

**R#** - radius of the saw blade.

**O#** - offset, in degree, from home position to 0 degree cut direction.

**Z#** - lift amount used when auto-lift is used for saw blade rotation.

**scan=T# A# C# S# H# F# P# D# (T, A & C are required)**

Configures touch-probe scanning to digitize 3D objects.

**T#** is the probe type. 1 is an up/down probe. 2 is an up/down/side to side. This setting is **required**.

**A#** is the axis # the probe is on. 2 for Z. Optional setting. Defaults to 2.

**C#** is the auxin channel the probe is connected to. This setting is **required**. An Auxin line must be added to the WinCNC.ini file.

This channel # cannot be the same as any other used channel.

**S#** is the # steps to decelerate the probe. Optional setting that defaults to 50 if not specified.

**H#** is the height to lift when the probe comes into contact with something. Optional setting that defaults to .005 if not specified.

**F#** is the feed rate to scan with. Optional setting that defaults to 30 if not specified.

**P#** is the plunge for the A#. Optional setting that defaults to 30 if not specified.

**D#** is the debounce or number of times that the input status is read to determine a state change. Default is 0.

Example: scan=T2 A2 C8 S50 H.005 F30 P30

**signal=C# M"message" T# I#**

Signal is used to set up control push buttons. First set up an auxin line for the input. Then specify signal as follows to use the input channel as a control signal.

**C#** specifies the input channel to be used for the signal.

**M"message"** denotes a custom error message. The message must be enclosed in double quotes - "".

**I#** is the initial state (1=active, 0=inactive).

**T#** specifies the signal type.

\*100 Abort

\*101 Pause/Continue

\*102 Pause

\*103 Continue

\*104 This signal will prevent all commands or attempts to run a file but will allow jog/transit moves if the signal is in an "on" state. If the signal turns "on" during a move or while running a file, WinCNC will pause, and will not allow a continue until the signal has turned back to an "off" state.

\*110 Start Cycle

\*201 Run command stored with L201 command

\*202 Run command stored with L202 command

**skipchars=C"#" N#**

A character or group of characters that indicate a block of G-Code that should be skipped or ignored.

**C"#"** - # represents the character or characters that mark the beginning of the code block to be skipped. Characters must be placed within quotes.

**N#** - # represents the number of lines to skip.

Example: skipchars=C"#%" N10

Sets WinCNC to skip 10 lines of G-Code each time that it reads a '#' or a '%' character at the beginning of a line in a G-Code file.

**softlim=T# M# B# V#**

Configures software limits for motion control.

**T#** can be one of the following:

- 0 = disabled
- 1 = enabled and active
- 2 = enabled but inactive

**M#** is to tell whether or not to show the softlim's toolbar icon. 1 will result in the toolbar icon being shown, 0 for not shown. Boundaries and Soft Limits are linked, so the state of one represents the state of the other.

**B#** can be one of the following:

- 0 = exceeding boundaries will cause an abort.
- 1 = exceeding boundaries will adjust the target position to the boundary location.

**V#** prevents soft-limit and boundary aborts from occurring on vertical axes. When this mode is active any command that exceeds the soft-limit or boundary of a vertical axis will be adjusted so that the axis move to the limit position and no error message is generated. Can be 0 or 1, 1 meaning activated and 0 is deactivated. Default is 0.

**spinchar=#**

**#** - character used for spindle speed control.

**spindle=T# R# I# A# H#**

Support for D/A card for spindle speed control.

**T#** is the type of spindle speed control.

- 1 = D/A (CN2/Pins 12-14).
- 2 = Auxout (Spin.mac).
- 3 = Frequency (CN2/Pin 11)\*.
- 4 = Pulse Width Modulation (PWM).

**R#** is the maximum revolutions per minute for the spindle.

**I#** specifies the initial RPM when WinCNC starts.

**A#** adjusts the increment size for spindle slider and MPG wheel.

**H#** used to set spindle calibration to hertz. # is set to max hertz.

\*See freq=

Once this line is in the INI file the spindle speed can be set by using S# command where # is the desired rpm of the spindle.

**steppulse=P#**

**P#** is the length of each step pulse in microseconds.

**NOTE:** This setting defaults to 5, and is generally unused.

**surfacemap=A# B# C# X# Y# W# H# M# I# T#**

This feature is similar to the axis mapping feature, but instead uses an adjustment for vertical axis based on the XY position.

**A#** is the horizontal X axis number. (0=X, 1=Y, 2=Z,...)

**B#** is the horizontal Y axis number. (0=X, 1=Y, 2=Z,...)

**C#** is the vertical Z axis number. (0=X, 1=Y, 2=Z,...)

**X#** is the map area start position for X axis.

**Y#** is the map area start position for Y axis.

**W#** is the width of map area along X axis.

**H#** is the height of map area along Y axis.

**M#** is the map increment, i.e. spacing between grid points of the map area.

**I#** is the interpolation increment, i.e. length of line segments for adding adjustments.

**T#** is the tolerance. Vertical axis adjustments of this amount or less are ignored.

**table=X# Y# W# H# B# Z# I#**

Configures table dimensions for the G-Code/BMP Viewer.

**X#** is the low X coordinate to start viewer area.

**Y#** is the low Y coordinate to start viewer area.

**W#** is the width of the table area to view in viewer.

**H#** is the height of the table area to view in viewer.

**B#** is the border size in units (keeps table outline viewable).

**Z#** is the auto-zoom for the G-Code Viewer (requires that Auto-preview be turned on in the Options menu). 1 (default) - auto zoom out only. 2 - auto zoom in and out. 3 - no auto zoom.

**I#** enables/disables GCode drawing on startup. 0 = disable, 1 = enable.

**thc=A# F# V# S# E# P# O# I# D# L# T# C# W# M#** (Torch Height Control)

**A#** is the axis number (x=0, y=1, z=2, etc.) to adjust for THC.

**F#** is the feed rate or velocity, used to adjust the specified axis up/down for THC. (Default = 100)

**V#** is the target voltage for the THC, must be between 20 and 250.

**S#** is the start, or pierce delay, in seconds, used for THC when turning the torch on.

**E#** is the end delay, in seconds, used for THC when turning the torch off.

**P#** is the percentage (1-100) of the programmed velocity that when reached will activate the THC.

**O#** is the auxiliary output channel number used to turn the torch on and off for THC.

**I#** is the input channel used to detect the “arc good” signal for THC.

**D#** specifies the compression distance of the switch used for the material touch-off.

**L#** sets the distance to raise the torch when it is turned off prior to moving to the next pierce location.

**T#** is the tolerance. Voltage changes of less than T# are ignored.

**C#** some plasma units allow for an alternate “Cold Pierce”. # is the in seconds that must expire before qualifying for a “Cold Pierce”.

**W#** W# works in conjunction with C above. If the “Cold Pierce” time has expired, # is the time in seconds that WinCNC will use for the “Cold Pierce Time” instead of the normal pierce time.

**M#** this is used for the new voltage divider THC board. **M2** is for raw voltage **divided by 50**. **M3** is for raw voltage **divided by 40**.

**thcauto=#**

Allows the user to show or hide the “Auto” THC mode screen push button and material configuration box radio button.

# can be 0 or 1, 0 hides the “Auto” buttons and 1 shows them. Default is 1.

**thcdiv=M# B#**

**M#** is the arc voltage divider (usually 40 or 50).

**B#** is the size of the read buffer used for averaging. Default is 20.

Used with THC arc voltage divider board.



**thcdisab=V# T#**

**V#** is the amount of instant voltage change that will trigger a disable.

**T#** is how long in seconds to disable.

**thclim=L# H#**

**L#** sets the low voltage. Default value is 20 volts.

**H#** sets the high voltage. Default value is 350 volts.

Allows setting the minimum and maximum THC voltages. This can be used to limit THC response to high voltage spikes or drops due to arc failure or electrical interference.

**thcvel=F# V#**

**F#** is the velocity

**V#** is the voltage

If the change in arc voltage is equal to or greater than V# the adjusting axis will move at the velocity specified F#.

**tranadj=#**

The # value can be set to any non-zero positive value. This setting controls where a transit move will stop automatically relative to the soft limit or boundary specified for an axis.

**vgroup=<F>###**

F is only used for the feed rate command group. It normally locks the X and Y axes together. It isn't used on any other vgroup line. #s are the desired alpha axis labels to group velocity.

Examples:

vgroup=FXY is the special feed rate command group; If the command F300 is given on the command line, it forces the X and Y axes to use 300 as their feed rate.

vgroup=XY would lock the X and Y axes, so that if the command G1 X15 F50 is entered, both axes would be set to 50. Otherwise, Y would remain at whatever feed rate it had previously been set to. Other than the X and Y axes, each axis should have its own vgroup; i.e. vgroup=Z.

**virtin=C# I#"**

**C#** - virtual input channel number (1-64).  
**I#** - desired initialization state, 1 for on or 0 for off.

**vmatch=P# T#**

Used to improve velocity matching when using a rotary axis together with vertical/horizontal axis. This setting is used to adjust the accuracy of vector velocity matching.

**P#** - vector velocity matching percentage. **#** - Can be set between 0.9 and 1. The default value is 1.0.  
**T#** - type setting. Use T1 for the old algorithm or T2 for the new algorithm.

**watch=T# D# C#**

The "watch" feature in WinCNC generates a square wave using an Auxout channel, CN2 Pin 11, or CN2 Pin 15.

**T#** Type of watch. Can be type 1 - 10  
**C#** Channel used for T7  
**D#** Frequency used for watch types 8,9, and 10. Minimum value 65, maximum 2000000 (65Hz to 2MHz).

**T1, T4, and T8** receive output from CN2 Pin 11. **T2, T5, and T9** receive output from CN2 Pin 15, **T3, T6, and T10** receive output from either or both Pin 11 and Pin 15.

**T1, T2, and T3** all work as an output on (hi) or off (low).  
**T4, T5, and T6** receive a software generated square wave with a variable frequency. Minimum frequency is 33 Hz and a maximum of 100KHz.

**T7** works with output channel **C#** and functions exactly like T4, 5, and 6.

**T8, T9, and T10** receive a hardware generated square wave with a **D#** frequency.

## WinCNC Controller

**xhc=T# D# S# A""**

**T#** sets the type. Enter 1 for HB04. Enter 2 for PHB04B.

**D#** sets the display refresh in milliseconds. Default is 500.

**S#** sets the MPG transit stop or timeout in milliseconds. Default is 200.

**A""** sets axis designation. Default is A"XYZ". Conflicting axis names can be resolved by replacing the axis with a space.

Example: A"XYZ A", where W is omitted.

This page is intentionally left blank.

## CNCSCRN.INI Settings

### Program Icon

"Icon", "c:\wincnc\icons\msicons.ico"

### Display Sections

Syntax: "Display", "Viewer", Left, Top, Width, Height

Example: "Display", "Viewer", 370, 145, 265, 180

Syntax: "Display", "ButtonSet", StartX, StartY, EndX, EndY

Example: "Display", "ButtonSet", 0, 200, 680, 520

Syntax: "Display", "Command", StartX, StartY, EndX, EndY

Example: "Display", "Command", 0, 0, 425, 195

Syntax: "Display", "Position", Left, Top, Width, Height, G92 or ABS/LCL (0 for G92, 1 for ABS/LCL), Decimal Places in Position (can be 2, 3, or 4), Decimal Places in Feedrate (Can be 0 to 2)

Example: "Display", "Position", 430, 0, 235, 150, 0, 3, 0

Example: "Display", "StoredPosition", 5, 330

Syntax: "Display", "Main Window", Left, Top, Width, Height, Topmost

Example: "Display", "Main Window", 0, 0, 680, 435, 1

**The above line sets the left, top corner of the main window at 0, 0, sets the width to 680 and the height to 435, and makes WinCNC the topmost window.**

Example: "Display", "Title", "MicroSystems WinCNC Controller"

Syntax: "Display", "Main Color", Red, Green, Blue

Example: "Display", "Main Color", 180, 180, 180

Syntax: "Display", "Box Color", Red, Green, Blue

Example: "Display", "Box Color", 180, 180, 180

Syntax: "Display", "Text Color", Red, Green, Blue

Example: "Display", "Text Color", 180, 180, 180

Syntax: "Display", "Borders", 0

Example: "Display", "Borders", 1

Syntax: "Display", "JobWindow", Left, Top

Example: "Display", "JobWindow", 5, 350

Syntax: "Display", "THCWindow", Left, Top

Example: "Display", "THCWindow", 822, 505

Syntax: "Display", "AGasWindow", Left, Top

Example: "Display", "AGasWindow", 147, 415

Syntax	"Viewer", "<TYPE>", Red (0-255), Green (0-255), Blue (0-255), 0=Solid/1=Dash, Width (Pixels)
Defaults	"Viewer", "Table", 255, 0, 0, 0, 2 "Viewer", "Feedline", 0, 0, 255, 0, 1 "Viewer", "RapidLine", 0, 0, 0, 1, 1 "Viewer", "PreFeedLine", 180, 180, 180, 0, 1 "Viewer", "PreRapidLine", 180, 180, 180, 1, 1 "Viewer", "Grid", 210, 210, 210, 0, 1 "Viewer", "Local", 0, 0, 0, 0, 1

Syntax	"Viewer", "Background", Red (0-255), Green (0-255), Blue (0-255)
Default	"Viewer", "Background", 255, 255, 255

Syntax	"Viewer", "PosIndicator", Red (0-255), Green (0-255), Blue (0-255), 1 [Indicator Enabled]
Default	"Viewer", "PosIndicator", 255, 0, 0, 1

**Slider Controls**

Syntax: "SlideControl", Left, Top, Width, Height, Output, Unused, Unused, Unused, Unused, ScanOn, ScanOff  
 The Output value can be 0 for Spindle Speed, 1 for D/A, 2 for Laser Power, 3 for Maximum Pump Output percentage, and 4 for Pump Reverse Time percentage. If using Spindle Speed alone, or Laser Power alone, the +- (plus and minus) keys are used to adjust values up or down. If using Spindle Speed and Laser Power together, Laser Power uses the +- (plus and minus) keys and the Spindle Speed uses the *Shift* +- (Shift, plus and minus) key combinations.

"SlideControl", 425, 5, 235, 50, 0, 0, "", "", "", -1, -1

**Jog Mode Buttons**

Syntax: "JogMode", Left, Top, Width, Height, Mode, Default Speed, Label, Unused, Unused, ScanOn, ScanOff, Bitmap, Group

These buttons allow you to switch jog mode and transit speeds.

The mode number can be one of the following:

8=Custom (Uses the label value),7=Fast,6=Medium,5=Slow,4=1",  
3=0.1",2=0.01",1=0.001"

Default Speed must be 0 except for the one chosen to be default.

The default must be 1 (See page 20 for example).

"JogMode", 0, 5, 35, 25, 7, 0, "Fast", "", "", -1, -1, "fast.bmp"

"JogMode", 0, 35, 35, 25, 6, 0, "Med", "", "", -1, -1, "med.bmp"

"JogMode", 0, 65, 35, 25, 5, 0, "Slow", "", "", -1, -1, "slow.bmp"

"JogMode", 50, 95, 35, 25, 4, 0, "1", "", "", -1, -1, "1.bmp"

"JogMode", 90, 95, 35, 25, 3, 0, ".1", "", "", -1, -1, "p1.bmp"

"JogMode", 130, 95, 35, 25, 2, 0, ".01", "", "", -1, -1, "p01.bmp"

"JogMode", 170, 95, 35, 25, 8, 0, "10", "", "", -1, -1, "10.bmp"

The JogModeEx button is identical to the old JogMode button, except for the addition of a second bitmap name parameter. When using the JogModeEx button type the first specified bitmap will be displayed on the button surface when the button is on, the second bitmap is displayed when the button is off.

"JogModeEx", 5, 5, 50, 50, 1, 0, "", "0.001", "", -1, -1, "001ON.BMP",  
"001OFF.BMP", 1

**Frames**

Syntax: "Frame", Left, Top, Width, Height, FrameType, Unused, Unused, Unused, Unused, ScanOn, ScanOff, Unused, Group #

Using group frames allows the user to position buttons, labels, and indicators inside of a frame using the upper left corner of the frame as the (0, 0) location. When the frame is moved all buttons in the same group move with it. Groups can be assigned by adding a group number to the end of a "Frame" line, and then adding the group "AuxIn" line. Allows user to draw boxes around screen elements to organize them. The appearance of the frame is determined by FrameType: 0 = Invisible, 1 = Etched Frame.

Following are some examples of how to use the group frames.

"Frame", 210, 0, 115, 125, 1, 0, "", "", "", -1, -1, "", 1 [Group 1 Frame]

"UserBtn", 5, 5, 50, 25, 0, 0, "G90", "G90", "", -1, -1, "", 1 [Group 1]

"UserBtn", 5, 35, 50, 25, 1, 0, "G91", "G91", "", -1, -1, "", 1 [Group 1]

"UserBtn", 5, 65, 50, 25, 2, 0, "Home", "G28", "", -1, -1, "", 1 [Group 1]

"Frame", 325, 0, 45, 170, 1, 0, "", "", "", -1, -1, "", 2 [Group 2 Frame]

"AuxIn", 0, 5, 5, 20, 18, 1, 1, "C1", 2 [Group 2]

"AuxIn", 0, 5, 25, 20, 18, 2, 1, "C2", 2 [Group 2]

**AuxIn Indicators**

Syntax: "AuxIn", Type(0), Left, Top, Width, Height, Channel(1-32), Color(1-6), Label, Group

Auxin Indicators represent auxiliary inputs for which you want a status display. Channel corresponds to channel number in WINCNC.INI.

The following are the color schemes for the color section of the AuxIn screen line:

- 1 - On=Green, Off=Red
- 2 - On=Green, Off=Gray
- 3 - On=Red, Off=Gray
- 4 - On=Red, Off=Green
- 5 - On=Gray, Off=Green
- 6 - On=Gray, Off=Red

"AuxIn", 0, 340, 60, 18, 20, 1, 1, "Aux-In 1"



**Virtual AuxIn Indicators**

Syntax: "VirtualIn", Type(0), Left, Top, Width, Height, Channel(1-32), Color(1-6), "Label", "", Group

Virtual inputs work in similar fashion to standard aux-inputs, except that virtual input states (on/off) can only be changed by command. Virtual input states are stored and re-loaded when WinCNC closes and re-opens. There are 64 virtual input channels available for use. The following explains configuring and using virtual inputs.

L60 C# - Turn specific virtual input off, # can be 1-64

L61 C# - Turn specific virtual input on, # can be 1-64

M17.3 C# - Run next line if the specified input is on, # can be 1-64

**AuxOut Buttons** - buttons used to control auxiliary outputs.

Syntax: "AuxOut", Left, Top, Width, Height, Channel, 0, Label, AuxOn, AuxOff, ScanOn, ScanOff, Bitmap, Group

Channel should be set to the auxiliary channel you want to control. Label determines the text that is displayed on the button. AuxOn is a G-Code command that turns on your auxiliary output and may utilize macros. AuxOff is the G-Code command that turns off the auxiliary output.

**NOTE:** The 0 channel is reserved for laser enable control only! Standard auxout do not use 0.

"AuxOut", 280, 35, 50, 25, 0, 0, "Laser", "L251", "L250", -1, -1, "Laser.bmp"

"AuxOut", 280, 65, 50, 25, 1, 0, "AuxOut 1", "M11C1", "M12C1", -1, -1, "Auxout.bmp"

**AuxOutEx Buttons** - buttons used to control auxiliary outputs.

Syntax: "AuxOutEx", Left, Top, Width, Height, Channel, 0, Label, AuxOn, AuxOff, ScanOn, ScanOff, OnBitmap, OffBitmap, Group

The AuxOutEx button is identical to the old AuxOut button, except for the addition of a second bitmap name parameter. When using the AuxOutEx button type the first specified bitmap will be displayed on the button surface when the button is on, the second bitmap is displayed when the button is off.

"AuxOutEx", 5, 5, 50, 50, 1, 0, "", "M11C1", "M12C1", -1, -1, "C1ON.BMP", "C1OFF.BMP", 1

**Job Command Buttons**

“JobCmd”, Left, Top, Width, Height, Command, Unused, Label, Unused, Unused, ScanOn, ScanOff, Group

JobCmd buttons represent non G-Code commands you want to send to WinCNC. The Command parameter can be one of the following:

- |                              |                             |
|------------------------------|-----------------------------|
| 1 = Abort motion             | 7 = Manual Digitizing Rapid |
| 2 = Run current command      | 8 = Manual Digitizing Feed  |
| 3 = Pause motion             | 9 = Manual Digitizing Arc   |
| 4 = Refresh keypad file list | 10 = Digitizing Close shape |
| 5 = Restart                  | 11 = Digitizing Undo        |
| 6 = Single Step              |                             |

“JobCmd”, 0, 0, 0, 0, 1, 0, “Abort”, “”, “”, 17,0, “”, 1

“JobCmd”, 0, 0, 0, 0, 3, 0, “Pause”, “”, “”, 13, 0, “”, 1

**Job Command Ex Buttons**

Syntax: “JobCmdEx”, Left, Top, Width, Height, Command, Unused, Label, Unused, Unused, ScanOn, ScanOff, Up Bitmap, Down Bitmap, Group

JobCmdEx buttons work exactly like the JobCmd buttons but have a second bitmap that helps determine when the button is up or down.

“JobCmdEx”, 0, 0, 0, 0, 1, 0, “Abort”, “”, “”, 17,0, “”, “”, 3

“JobCmdEx”, 0, 0, 0, 0, 3, 0, “Pause”, “”, “”, 13, 0, “”, “”, 3

**Shell Command Buttons**

Syntax: “ShellCmdBtn”, Left, Top, Width, Height, ButtonNum, Unused, Label, Cmd, Directory, ScanOn, ScanOff, Bitmap, Group

Shell command buttons allow you to create buttons inside the WinCNC window, which execute outside commands or programs. Any command that can be executed from the Windows START->RUN line can also be run from a shell command button. The following example shows how to create a shell command button for the calculator application that is included in all versions of Windows.

“ShellCmdBtn”, 215, 95, 50, 25, 0, 0, “Calc.”, “calc.exe”, “”, -1, -1, “”

**User Buttons**

Syntax: "Userbtn", Left, Top, Width, Height, ButtonNum, Unused, Label, Command, Unused, ScanOn, ScanOff, Bitmap, Group

Userbtn is used to create customized push buttons that send G-Code whenever they are pressed. The ButtonNum is used internally to identify the button. It is recommended that you use sequential integers starting with 0. The Label parameter determines the text that is actually printed on the button. The Command parameter is G-Code that is executed when the button is pressed.

Userbtn", 225, 5, 50, 25, 0, 0, "G90", "G90", "", -1, -1, "g90.bmp", 2

**User Button Ex**

Syntax: "UserbtnEx", Left, Top, Width, Height, ButtonNum, Unused, Label, Command, Unused, ScanOn, ScanOff, Bitmap, Bitmap, Group

UserBtnEx is identical to the UserBtn button, but add a second bitmap that is displayed when the button or pressed.

UserbtnEx", 225, 5, 50, 25, 0, 0, "G90", "G90", "", -1, -1, "g90Up.bmp", "g90Down.bmp", 2

**Repeat Command Buttons**

Syntax: "RptCmd", Left, Top, Width, Height, Command, Delay, Label, Unused, Unused, ScanOn, ScanOff, , Bitmap, Group

RptCmd buttons represent non G-Code commands you want to send to WinCNC, using the serial keypad. These buttons are not shown on the main window. RptCmd commands should be used for JobCmds that need repeating key message. The Command parameter can be one of the following:

- 1 = Increase override
- 2 = Decrease override
- 3 = Increase laser power
- 4 = Decrease laser power
- 5 = Increase spindle speed
- 6 = Decrease spindle speed

```

"RptCmd", 0, 0, 0, 0, 1, 0, "Inc Override", "", "", 7, 0, ""
"RptCmd", 0, 0, 0, 0, 2, 0, "Dec Override", "", "", 8, 0, ""
"RptCmd", 0, 0, 0, 0, 3, 0, "Inc Laserp", "", "", 6, 0, ""
"RptCmd", 0, 0, 0, 0, 4, 0, "Dec Laserp", "", "", 5, 0, ""
"RptCmd", 0, 0, 0, 0, 5, 0, "Inc Spindle", "", "", -1, -1, ""
"RptCmd", 0, 0, 0, 0, 6, 0, "Dec Spindle", "", "", -1, -1, ""
Delay is in milliseconds - 0 = no delay, 500 = 1/2 second.
"RptCmd", 0, 0, 0, 0, 6, 500, "Dec Spindle", "", "", -1, -1, ""

```

**Repeat Command Ex Buttons**

Syntax: "RptCmdEx", Left, Top, Width, Height, Command, Unused, Label, Unused, Unused, ScanOn, ScanOff, Up Bitmap, Down Bitmap, Group

RptCmdEx buttons work exactly like the RptCmd buttons but have a second bitmap that helps determine when the button is up or down.

```

"RptCmdEx", 0, 0, 0, 0, 1, 0, "Inc Override", "", "", 17, 0, "", "", 3
"RptCmdEx", 0, 0, 0, 0, 3, 0, "Inc Laserp", "", "", 13, 0, "", "", 3

```

**Key Pad Out Buttons**

Syntax: "KeyPadOut", Left, Top, Width, Height, Command, Unused, Label, Unused, Unused, ScanOn, ScanOff, Bitmap, Group

KeyPadOut buttons represent commands you want to send to the keypad. The command parameter can be one of the following:

- 1 = Scroll keypad file list up
- 2 = Scroll keypad file list down
- 3 = Run current keypad file
- 4 = Refresh keypad file list
- 5 = Repeats last command
- 6 = Shifts the keypad axis display up
- 7 = Shifts the keypad axis display down

"KeyPadOut", 0, 0, 0, 0, 1, 0, "Up", "", "", 2, 0, ""

"KeyPadOut", 0, 0, 0, 0, 2, 0, "Down", "", "", 3, 0, ""

**Key Pad Out Ex Buttons**

Syntax: "KeyPadOutEx", Left, Top, Width, Height, Command, Unused, Label, Unused, Unused, ScanOn, ScanOff, Up Bitmap, Down Bitmap, Group

KeyPadOutEx buttons work exactly like the KeyPadOut buttons but have a second bitmap that helps determine when the button is up or down.

"KeyPadOutEx", 0, 0, 0, 0, 1, 0, "Up", "", "", 0, 0, "", "", 3

"KeyPadOutEx", 0, 0, 0, 0, 3, 0, "Run File", "", "", 0, 0, "", "", 3

**Menu Buttons**

Syntax: "MenuBtn", Left, Top, Width, Height, Function#, Unused, Label, Unused, Unused, ScanOn, ScanOff, Bitmap, Group

The menu button can be used to add buttons to the main screen and serve as shortcuts to menu options. The Function# can be set to any of the following:

- 1 = Set Local Coordinates
- 2 = Open File Dialog
- 3 = Run command/file
- 4 = Pause
- 5 = Abort
- 6 = File Restart
- 7 = View History
- 8 = Edit Current File
- 9 = Simulate GCode
- 10 = Draw GCode view
- 11 = Inch Mode
- 12 = Centimeter Mode
- 13 = Millimeter Mode

```

"MenuBtn", 5, 95, 50, 25, 1, 0, "Set Local", "", "", -1, -1, "", 1
"MenuBtn", 5, 95, 50, 25, 2, 0, "Open File", "", "", -1, -1, "", 1
"MenuBtn", 5, 95, 50, 25, 3, 0, "Run", "", "", -1, -1, "", 1
"MenuBtn", 5, 95, 50, 25, 4, 0, "Pause", "", "", -1, -1, "", 1
"MenuBtn", 5, 95, 50, 25, 5, 0, "Abort", "", "", -1, -1, "", 1
"MenuBtn", 5, 95, 50, 25, 6, 0, "File Restart", "", "", -1, -1, "", 1
"MenuBtn", 5, 95, 50, 25, 7, 0, "View History", "", "", -1, -1, "", 1
"MenuBtn", 5, 95, 50, 25, 8, 0, "Edit File", "", "", -1, -1, "", 1
"MenuBtn", 5, 95, 50, 25, 9, 0, "Simulate", "", "", -1, -1, "", 1
"MenuBtn", 5, 95, 50, 25, 10, 0, "View GCode", "", "", -1, -1, "", 1
"MenuBtn", 5, 95, 50, 25, 11, 0, "Inch Mode", "", "", -1, -1, "", 1
"MenuBtn", 5, 95, 50, 25, 12, 0, "CM Mode", "", "", -1, -1, "", 1
"MenuBtn", 5, 95, 50, 25, 13, 0, "MM Mode", "", "", -1, -1, "", 1

```

**Axis Controls**

Syntax: "AxisControl", Left, Top, Width, Height, Axis, Direction, Unused, Unused, Unused, ScanOn, ScanOff, Bitmap, Group

AxisControl sets up control of each axis. Axis number (zero based) and direction (0 or 1). It is a single command button.

"AxisControl", 121, 54, 37, 45, 0, 1, "", "", "", -1,-1, "xRIGHT.bmp", 4

Syntax: "AxisControlEx", Left, Top, Width, Height, Axis, Direction, Unused, Unused, Unused, ScanOn, ScanOff, Bitmap, Bitmap, Group

"AxisControlEx", 121, 54, 37, 45, 0, 1, "", "", "", -1,-1, ""up.bmp", "down.bmp", 4

AxisControlEx is identical to the AxisControl button, but add a second bitmap that is displayed when the button or pressed.

Axis numbers are zero-based. The letter displayed is determined by the axis given. You can change these letters in WINCNC.INI.

"AxisLabel", 50, 37, 35, 13, 0, 0, "", "", "", -1, -1, "x.bmp"

"AxisLabel", 90, 37, 35, 13, 1, 0, "", "", "", -1, -1, "y.bmp"

**Rapid Mode**

Syntax: "RapidMode", Left, Top, Width, Height, Mode, Unused, Label, Unused, Unused, ScanOn, ScanOff, Bitmap, Group

The RapidMode button allows the user to select a velocity percentage to use for rapid G0/G53 moves. Modes can be set to:

1 = 5%, 2 = 25%, 3 = 50%, 4 = 75%, or 5 = 100%.

"RapidMode", 0, 0, 0, 0, 2, "", "25%", "", "", -1,-1, "", 1

The RapidModeEx button is identical to the RapidMode button, except for the addition of a second bitmap name parameter. When using the RapidModeEx button type the first specified bitmap will be displayed on the button surface when the button is on, the second bitmap is displayed when the button is off.

"RapidModeEx", 5, 5, 50, 50, 4, 0, "", "75%", "", -1, -1, "75%ON.BMP", "75%OFF.BMP", 1

**State Buttons**

Syntax: "StateBtn", Left, Top, Width, Height, Type, Unused, Label, Command, Unused, ScanOn, ScanOff, Group

State buttons represent the state of a feature on or off. The buttons will stay down when clicked until clicked again to disable the feature. The type sets the class of the feature. The example below displays type 8 which is for the scanner feature. If an L800 or L801 command is typed into the command line, the button will change states to represent the command. The commands are represented left to right, with the left command, "L801" in the example to represent what happens when the feature is disabled, the state button is out in a normal state and can be clicked to turn on the feature. The second command "L800" is the command that is issued when the button is up and is clicked to the down state. State button types are as follows:

- 0 - Reserved
- 1 - Reserved
- 2 - Reserved
- 3 - Reserved
- 4 - Velocity mode for pump On/Off
- 5 - Reverse mode for pump indicator. Indicates if in reverse mode.
- 6 - Purge mode for pump
- 7 - Tangential Knife mode On/Off
- 8 - 3D Probe Scanning On/Off
- 9 - Smart Bitmap engraving
- 10 - AutoAuxOut

"StateBtn", 335, 25, 50, 20, 8, 0, "Scan", "L801", "L800", -1, -1, ""

StateBtnEx is identical to the StateBtn button, except for the addition of a second bitmap name parameter. When using this button type the first specified bitmap will be displayed on the button surface when it is on, the second bitmap is used when the button is off.

"StateBtnEx", 5, 5, 50, 50, 10, 0, "", "L34", "L35", -1, -1, "TON.BMP", "TOFF.BMP", 1



**Virtual Input Button**

Syntax: "VirtInBtn", Left, Top, Width, Height, Channel, 0, Label, VirtInOn, VirtInOff, ScanOn, ScanOff, Bitmap, Group

Channel should be set to the virtual input you want to control.

Label determines the text that is displayed on the button if no bitmap is used. 'VirtinOn' is a G-Code command that turns on your virtual input (L60C#) and may utilize macros. VirtInOff is the G-Code command that turns off the virtual input (L61 C#). It too may utilize macros.

"VirtInBtn", 280, 65, 50, 25, 1, 0, "Virtual Input 1", "L60 C1", "L61 C1", -1, -1, "VirtIn1.bmp", 3

The VirtInBtnEx button is identical to the old VirtInBtn button, except for the addition of a second bitmap name parameter. When using the VirtInBtn button type the first specified bitmap will be displayed on the button surface when the button is on, the second bitmap is displayed when the button is off.

"VirtInBtnEx", 280, 65, 50, 25, 1, 0, "", "L60C1", "L61C1", -1, -1, "C1ON.BMP", "C1OFF.BMP", 1

This page is intentionally left blank.

**Command Reference**

Parameters in [Brackets] are optional.

XYZWIJ . . .	Axis Specification X# Y# Z#
<p>Axis values are specified with a decimal point. A value with no decimal is read as an integer value. No value is equivalent to specifying 0.</p> <p>Example: XYZ is equivalent to X0Y0Z0</p>	
F	Feed Velocity - F#
<p>Velocity is stored separately for Linear XY, Linear ZW. Arc's Rapid and Feed Velocity is stored separately for Linear Moves. Independent velocities are stored based on the XYZW specified in the line containing the F# command. Velocity is specified in units of measure per minute.</p> <p>Example: F60 sets all axes in the feed rate override command group to a Feed Velocity of 60 (see "vgroup=").  G1 X F60 sets the X axis and all other axes grouped with X to a Feed Velocity of 60.  F100 X Y sets the X and Y axes to a Feed Velocity of 100. It also sets any other axes that are grouped with either axis to a Feed Velocity of 100. For information about axis grouping, see "vgroup=".  Feed Velocity changes affect all axes in that vgroup.</p>	
P	Stored Positions - P#
<p>Move to the selected stored position.</p>	
H	Stored Home Positions - H#
<p>Move to the selected stored Home position.</p>	
S	Spindle Speed - S#
<p>Sets the spindle speed to the given value.</p>	
[ ]	Comment
<p>Used to add comments to programs. A closing bracket is optional.</p>	
{ }	Braces
<p>Used to substitute axis or parameter values in a line of G-Code. Internally defined variables as well as constant numbers can be used within the braces. Following is a list of internally defined variables and what they represent.</p>	

**Internally Defined Variables**

TP1	Tool number, tool changer 1.
TP2	Tool number, tool changer 2.
TN1	Tool number 1 + 10.
TN2	Tool number 2 + 20.
TS1	Tool number 1 + tool changer 1 offset.
TS2	Tool number 2 + tool changer 2 offset.
TH1	Tool height for tool changer 1.
TH2	Tool height for tool changer 2.
TM1	Height of tool length measure switch 1. Set using M37.1.
TM2	Height of tool length measure switch 2. Set using M37.2.
TMX	X position of tool length measure switch 1.
TMI	X position of tool length measure switch 2. (In rare cases, all heads cannot reach the same tool measure switch.)
TMY	Y position of tool length measure switch 1.
TMJ	Y position of tool length measure switch 2.
TMD	Rapid move distance for auto tool measure.
TMF	Feed rate for auto tool measure.
LZ0	Current G92 value for axis 0.
LZ1	Current G92 value for axis 1.
LZ2	Current G92 value for axis 2.
LZ3	Current G92 value for axis 3.
LZ4	Current G92 value for axis 4.
LZ5	Current G92 value for axis 5.
AP0	Absolute position for axis 0.
AP1	Absolute position for axis 1.
AP2	Absolute position for axis 2.
AP3	Absolute position for axis 3.
AP4	Absolute position for axis 4.
AP5	Absolute position for axis 5.
THM	1 if THC Manual Voltage Mode is enabled, 0 if disabled.
THA	1 if THC Auto Voltage Mode is enabled, 0 if disabled.

THB	1 if THC mode is enabled for the active material, 0 if THC mode is disabled for the active material.
THO	The specified output channel used to turn the torch on/off for THC.
THI	The specified input channel used to monitor the “arc good” signal for THC.
THP	1 if pierce height for THC is enabled, 0 if disabled.
THC	1 if cut height for THC is enabled, 0 if disabled.
THZ	The specified pierce height for THC.
THR	The specified cut height for THC.
THS	The specified pierce delay for THC.
THE	The specified end delay for THC.
THD	The specified switch compensation amount.
THL	The specified torch lift amount.
THF	The feed rate specified for THC.
THK	THC Slow Start Time
THV	THC Slow Start Feed %
THQ	THC Early Torch Off distance
THH	THC Elevation Height Enable Status: 1 = enabled, 0 = disabled.
THT	THC Elevation Height Position
THN	THC Plunge Rate
THW	THC Activation Delay
THX	THC Move to Cut Height Delay
TD1	Provides the difference between tool height for ATC1 and the value stored in H1 of the CNC.POS file.
TD2	Provides the difference between tool height for ATC1 and the value stored in H1 of the CNC.POS file.
PH0	Provides the value stored in H0 of the CNC.POS file.
PH1	Provides the value stored in H1 of the CNC.POS file.
PH2	Provides the value stored in H2 of the CNC.POS file.
PH3	Provides the value stored in H3 of the CNC.POS file.
PH4	Provides the value stored in H4 of the CNC.POS file.
PH5	Provides the value stored in H5 of the CNC.POS file.

PH6	Provides the value stored in H6 of the CNC.POS file.
PH7	Provides the value stored in H7 of the CNC.POS file.
PH8	Provides the value stored in H8 of the CNC.POS file.
PH9	Provides the value stored in H9 of the CNC.POS file.
PH10	Provides the value stored in H10 of the CNC.POS file.
PH11	Provides the value stored in H11 of the CNC.POS file.
PH12	Provides the value stored in H12 of the CNC.POS file.
PH13	Provides the value stored in H13 of the CNC.POS file.
PH14	Provides the value stored in H14 of the CNC.POS file.
PH15	Provides the value stored in H15 of the CNC.POS file.
PH16	Provides the value stored in H16 of the CNC.POS file.
PH17	Provides the value stored in H17 of the CNC.POS file.
PH18	Provides the value stored in H18 of the CNC.POS file.
PH19	Provides the value stored in H19 of the CNC.POS file.
PH20	Provides the value stored in H20 of the CNC.POS file.
PH21	Provides the value stored in H21 of the CNC.POS file.
PH22	Provides the value stored in H22 of the CNC.POS file.
PH23	Provides the value stored in H23 of the CNC.POS file.
PH24	Provides the value stored in H24 of the CNC.POS file.
PH25	Provides the value stored in H25 of the CNC.POS file.
PH26	Provides the value stored in H26 of the CNC.POS file.
PH27	Provides the value stored in H27 of the CNC.POS file.
PH28	Provides the value stored in H28 of the CNC.POS file.
PH29	Provides the value stored in H29 of the CNC.POS file.
GR0	Storage register 0 for G5.1 command.
GR1	Storage register 1 for G5.1 command.
GR2	Storage register 2 for G5.1 command.
GR3	Storage register 3 for G5.1 command.
GR4	Storage register 4 for G5.1 command.
GR5	Storage register 5 for G5.1 command.
GR6	Storage register 6 for G5.1 command.
GR7	Storage register 7 for G5.1 command.

GR8	Storage register 8 for G5.1 command.
GR9	Storage register 9 for G5.1 command.
JC0	Holds the current value of the Job Counter.
TR1	Provides radius of current tool in tool changer 1. Must be defined for that tool in parameter "Nominal Diameter" of the Tool Library. Only works for Automatic Tool Changers.
TR2	Provides radius of current tool in tool changer 2. Must be defined for that tool in parameter "Nominal Diameter" of the Tool Library. Only works for Automatic Tool Changers.

Additional variables can be defined in the CNC.MAC file. Defined variables and constants can be used together with simple math operators. ( \* ) multiplication, ( / ) division, ( + ) addition, ( - ) subtraction.

Example: G0Z{TH1+2} moves Z to tool height 1, plus 2 units.

**NOTE:** ANY use of braces requires a value or axis spec and a number. The second parameter inside the braces must be a number.

**G Codes**

G0	Rapid Move - G0 X# Y# Z# W#
<p>Moves to the position specified at Rapid velocity. G0 is modal. After a G0 is executed, lines with no G-Code command are executed as a G0.</p> <p>Example: X1Y1 is equivalent to G0 X1Y1 if mode is G0.</p>	
G0.1	Rapid Move with Vertical Lift - G0.1 [X#] [Y#]
<p>First lifts all vertical heads then moves the position specified at Rapid velocity then drops the vertical heads back to their previous positions.</p>	
G1	Feed Move - G1 X# Y# Z# W# L#
<p>Moves to the position specified at Feed velocity. G1 is modal. After a G1 is executed lines with no G-Code command are executed as a G1.</p> <p>Example: X1Y1 is equivalent to G1 X1Y1 if the mode is G1.</p> <p>L# is used to allow setting laser power in vector cutting (1-100.)</p>	
G2	Clockwise Arc - G2 X# Y# I# J# Z# W# K# R# L#
<p>Moves to the position specified at Feed velocity. I is the X distance to the center point. J is the Y distance to the center point. If no XY move is specified, a full circle is cut. If no I or J is specified, previous I J values are kept. Any additional axis spec that is not part of the arc itself will move that axis simultaneously throughout the arc. L# is used to allow setting laser power in vector cutting (1-100). R# is the radius, if R# is used IJK can not be used.</p>	
G3	Counter Clockwise Arc - G3 X# Y# I# J# Z# W# K# R# L#
<p>Moves to the position specified at Feed velocity. I is the X distance to the center point. J is the Y distance to the center point. If no XY move is specified, a full circle is cut. If no I or J is specified, previous I J values are kept. Any additional axis spec that is not part of the arc itself will move that axis simultaneously throughout the arc. L# is used to allow setting laser power in vector cutting (1-100). R# is the radius, if R# is used IJK can not be used.</p>	



G4	Dwell - G4 X#
<p>Stops movement for the time specified by the X value in seconds. There is no limit to delay time. If no time is specified, then the machine will be stopped until the operator pushes ENTER. Place a comment after the dwell to prompt the operator.  Example: G4 [Ready To Start Section 2]  <b>WARNING: Never use Dwell to stop the machine while changing parts!</b> Instead, program a single part and use the TAB key at the Program prompt. This will repeat the last part cut.</p>	
G5	G5 T# M"Message" I# F#
<p><b>T#</b> - Can be 0-4. Type 0 (default) is a simple OK message box displaying a message, type 1 is a Yes/No message box for abort, type 2 is a Yes/No message box to continue, type 3 is a Yes/No message box to run the next command, type 4 is a Yes/No message box to skip the next command.  <b>M"Message"</b>– Specifies the text to display in the message box enclosed in double quotes.  <b>I#</b> can be used to add integer numbers to the end of the text message specified with the M" parameter.  <b>F#</b> can be used to add floating point numbers to the end of the text message specified with the M" parameter.</p>	
G5.1	G5.1 R# H# A# M"Message"
<p>G5.1 pauses motion and creates a pop-up dialog box that will allow the user to input data into an internal variable. This variable can then be used in macros or subprograms. Ten new internal variables, GR0 through GR9, were added for this.  <b>A#</b> can be used to force a program abort if the user clicks the "Cancel" dialog box button. Use 'A1' to abort or 'A0' to continue. The default behavior is to continue without updating the specified storage register.  <b>H#</b> is the (old) 'H' register number and can be 0-29.  <b>M"Message"</b> is the message displayed on the pop-up dialog box.  <b>R#</b> is the register number and can be 0-9.</p>	

G9	Smoothing - G9 S# A#
<p>Used to set the smoothing factor. When the XY direction of motion changes, this setting can reduce the “slowdown” to improve the smoothness of motion. Increasing the S# decreases the slowdown of the machine when turning corners. Too high a factor can result in missed steps or motor stalls. Smoothing can also be set in the WINCNC.INI file using G09=S#.</p> <p>S# - smoothing factor to be used (not a specific unit of measure).  A# - used to change the angle used in the velocity matching algorithm. The A# value represents an angle. If a small angle, such as say 20, is specified, the G9=# will not apply to any angle smaller than 20.</p>	
G20	Converts unit of measure to inches.
G21	Converts unit of measure to centimeters.
G22	Converts unit of measure to millimeters.
G23	Returns to the last used unit of measure.
G28	Return to machine zero - G28 XYZW
<p>Moves specified axes to Lo Limit for XY - Hi Limit for ZW (these are the default, they can be reversed). Moves specified axes to WINCNC.INI specs from limits. Sets all axis positions to 0. All axes are moved if none are specified.</p>	
G31	Stop file parsing
<p>Must be used with the M28 command to measure machine position after touching a limit switch.  Example: L91 G1 Z-10 F20 M28 G31 M37 Z1.</p>	
G40	Cutter Compensation Off
<p>Turns off cutter compensation.</p>	
G41	Cutter Compensation Left - G41 O#
<p>Compensates for differences in nominal and actual tool diameters. Starts to the left on the first move.  O# is the compensation amount on machines without a tool changer.</p>	

G42	Cutter Compensation Right - G42 O#
Compensates for differences in nominal and actual tool diameters. Starts to the right on the first move. O# is the compensation amount on machines without a tool changer.	
G43	Tool Length Offset On - G43 ZWUV
Sets tool length offsets to the values set by the last M37 command. G43 alone turns all offsets on.	
G49	Tool Length Offset Off - G49 ZWUV
Cancels tool length offsets for axis specified. G49 alone turns all offsets off.	
G50	Scaling/Rotation Mode Off
Cancels the scaling or rotation mode.	
G51	Scaling Mode On - G51 X# Y# Z# I# J# K# C# W#
Enables the scaling factor. Scale each axis type based on the values specified by the X# Y# Z#. The object will be centered based on the I# (X center J# (Y) K# (Z). You can center all axes using the C# line	
G52	Rotation Mode On - G52 I# J# A#
Enables G-Code rotation. G-Code commands or files will be rotated by the specified number of degrees around the specified center point. I# is the X center point of rotation. J# is the Y center point of rotation. A# is the degrees to rotate.	
G52.1	Rotate G-Code moves around the current XY origin - G52.1 A# B#
Used to rotate G-Code moves around the current XY origin to the angle of a specified line. The line is defined by specifying two position numbers representing points on the line. A# - 1st point on the line. B# - 2nd point on the line.	
G53	Rapid Move - G53 X# Y# Z# W#
Moves to the position specified at rapid velocity, ignoring tool measures and local coordinates. G53 alone will lift all vertical or type 3 axes.	

G54	G54 Workspace - Rotational: G54 X# Y#
Select the Z head. If G54 is commanded from G55, G56, or G57 mode, the Z head will be moved to the current XY position.	
G55	G55 Workspace - Rotational: G55 X# Y#
Select the W head. If G55 is commanded from G54, G56, or G57 mode, the W head will be moved to the current XY position.	
G56	G56 Workspace - Rotational: G56 X# Y#
Select the U head. If G56 is commanded from G54, G55, or G57 mode, the U head will be moved to the current XY position.	
G57	G57 Workspace
Select the V head. If G57 is commanded from G54, G55, or G56 mode, the V head will be moved to the current XY position.	
G73	Chip Break Cycle - G73 X# Y# Z# R# Q# F#
Moves to XY specified at Rapid velocity. On the first peck, moves to R retract height at Rapid velocity, moves to Q peck depth at F feed velocity, lifts 0.05" at Rapid velocity to break the chip, moves to next peck depth at Feed velocity. The cycle repeats until Z depth is reached. Then moves back to retract height R at Rapid velocity. Without this cycle, some materials can produce a long "stringer" which can become a safety issue to the operator.	
G74	Left-hand tapping G74 Z# F# P#
Z# sets the depth for the tap. F# sets the vertical feed rate. P# sets the pitch of the tap.	
G80	End Drill Cycle
Used to end a drill cycle.	
G81	Drill Cycle - G81 X# Y# Z# R# F#
Moves to XY specified at Rapid velocity. Moves to R (clearing height) at Rapid velocity. Moves to Z specified at Feed velocity. Moves back to R at Rapid velocity. F allows you to set the feed rate.	

G82	Dwell Cycle - G82 X# Y# Z# R# P# F#
Moves to XY specified at Rapid velocity. Moves to R (clearing height) specified at Rapid velocity. Moves to Z specified at Feed velocity. If P is specified, bit pauses that amount of time at the bottom of the hole, then moves back to R at Rapid velocity. P is measured in milliseconds (thousandths of one second). F allows you to set the feed rate.	
G83	Peck Drill Cycle - G83 X# Y# Z# R# Q# F#
Moves to the XY specified at Rapid velocity. On the first peck, moves to R retract height at Rapid velocity, moves to peck depth Q at Feed velocity, then moves back to retract height at Rapid velocity. On each subsequent peck, moves at Rapid velocity to within 0.05" of previous peck depth, then moves at Feed velocity to next peck depth. Cycle repeats until Z depth is reached. Then moves back at Rapid velocity to retract height R. The F allows user to set the feed rate.	
G84	Right-hand tapping G84 Z# F# P#
Z# sets the depth for the tap. F# sets the vertical feed rate. P# sets the pitch of the tap.	
G90	Absolute Mode
Can be specified with other G-Codes on any line. XYZW values from the current line forward are read as absolute coordinates. IJ values are always relative to the current XY position, not absolute positions regardless of G90/G91 mode.	
G91	Relative Mode
Can be specified with other G-Codes on any line. XYZW values from the current line forward are read as relative movements from the current position.	

G92	Set Local Coordinates - G92 X# Y# Z# W#
<p>Used to specify a new coordinate system for running absolute mode programs. Use G92 alone to restore the Machine Coordinates. G92 X0 Y0 Z0 W0 sets the current position to zero. G92 then restores the Machine Coordinates values.</p> <p>**If running a .tap, .mac, .nc etc file, and there is a G92 in the file, and the user wishes to maintain that G92 after the file is done and closed, then a L92 must be added at the end of the file.</p>	
G92.1	Shift MZ Coordinates - G92.1 X# Y# Z# W#
<p>Similar to G92 except that the coordinate system produced shifts Machine Zero by the amount specified. This is useful since a given local coordinate system can be set without positioning the head to a certain position first. G92 X10 Y10 sets the absolute position X10 Y10 to X0 Y0 in local coordinates. G92.1 may also be used to restore a single axis to absolute coordinates. G92.1 X0 leaves YZW local coordinates but sets X back to absolute coordinates.</p>	
G92.2	Shift LZ Coordinates - G92.2 X# Y# Z# W#
<p>Similar to G92.1 except that the coordinate system shift is added to the current local coordinates instead of replacing them. This is useful for 'jogging' a local position. If the Z head has been set for running a G90 mode file but then needs to set to cut .010" deeper, G92.2 Z-.01 will accomplish this with a single command. Can be useful as a macro.</p>	
G92.3	Stores and Restores G92 values
<p>Allows the user to switch back and forth between local and absolute coordinates. Use the G92.3 command without any parameters or axis characters. If G92 positions are set on any axis then they will be stored and cleared. The next G92.3 call will restore the previous G92 values even if they have been cleared by using G92.</p>	
G92.4	Sets G92 XY offset positions to intersection of two lines - G92.4 A# B# C# D# O#
<p>G92.4 sets the G92 XY offset positions to the intersection of two lines. The lines are defined by specifying four stored position numbers representing two points on each of the two lines.</p> <p>A# - 1st point on first line. B# - 2nd point on first line. C# - 1st point on second line. D# - 2nd point on second line. O# - Amount to shift all points for compensation. Can be positive or negative.</p>	

G93	Inverse Time Feed Mode
<p>In inverse time feed rate mode, an F word means the move should be completed in [one divided by F number] minutes. For example, if the F number is 2.0, the move should be completed in half a minute. When the inverse time feed rate mode is active, an F word must appear on every line which has a G1 motion, and an F word on a line that does not have G1 is ignored. Being in inverse time feed rate mode does not affect G0 motions. It is an error if: inverse time feed rate mode is active and a line with G1 does not have an F word.</p>	
G94	<p>Cancel the Inverse Time Feed Mode</p> <p>Turns off the Inverse Time Feed Mode.</p>
G95	<p>Enable and Disable the Auto Rotary Axis Feed Mode</p> <p>Used to enable and disable the auto rotary axis feed mode. When the G95 mode is enabled the feed rate for all rotary (type 4) axis is automatically set, using the Z axis position as a radius value, to the specified linear feed rate.</p> <p>G95 F# - F# is the desired linear feed rate for rotary moves  G95 - A call with no parameters ends the auto rotary axis feed mode.</p>

## L Codes

L1	Save Positions - L1 N# V#
<p>Store values in the saved positions file. Can store a value in one of the 30 existing 'H' registers.  N# is the desired 'H' register number, can be 0 to 29.  V# is the desired value. Can be the result of a math operation or a constant.</p>	
L1.1	Save 'H' Register
<p>Save the 'H' register value to a file.</p>	
L1.2	Load 'H' Register
<p>Load the 'H' register value from a file.</p>	

L3	Set Home Position - L3 XYZW N#
Stores current position of each axis specified. Values specified are ignored. WinCNC provides 30 locations for storing positions. If L3 is used without the N#, the XY and Z values would be stored in position 0.	
L3.1	Store Local (G92) Offset - L3.1 XYZW N#
WinCNC provides 30 positions for storing local offsets. Those axes specified with G92 (green box) positions set will be stored in the N# position. If no N# is listed, the values will be stored at position 0. If there is an N# on the line, the G92 values will be stored at position N#. Example: L3.1XYZN4 Would store any local offsets for XY and Z in internal position 4.	
L3.2	Store Workspace (M37) Offsets - L3.2 XYZW N#
WinCNC provides 30 positions for storing workspace offsets. Those axes specified with M37 (blue box) positions set will be stored in the N# position. If no N# is listed, the values will be stored at position 0. If there is an N# on the line, the M37 values will be stored at position N#. Example: L3.2XYZN1 Would store any workspace offsets for XY and Z in internal position 1.	
L3.3	Calculate Distance Between two points - L3.3 A#B#H#
Used to calculate the distance between two points (stored with L3) and store the result in an 'H' register. A# specifies the first stored position. B# specifies the second stored position. H# specifies the 'H' register to store the distance from A to B in.	
L4	Return to Home
Moves each axis specified to the last L3 position stored. Values specified are ignored.	
L4.1	Recall Local (G92) Offset - L4.1 XYZWN#
Recall and set local (G92) values for axes specified from N# position. If no N# was listed on the line, values will be loaded from local storage position 0.	



L4.2	Recall Workspace (M37) Offset - L4.2 XYZW N#
Recall and set workspace (M37) values for axes specified from N# position. If no N# was listed on the line, values will be loaded from workspace storage position 0.	
L5	Set Contouring Velocity - L5 XYZ
Sets the contouring velocity in units per minute.	
L6	Set Rapid and Feed Acceleration - L6 XYZ
Sets the acceleration in units per minute per second. L6 alone restores the original values.	
L6.1	Set Rapid Accelerations L6 XYZ
Sets the acceleration in units per minute per second. L6 alone restores the original values.	
L6.2	Set Feed Acceleration - L6 XYZ
Sets the acceleration in units per minute per second. L6 alone restores the original values.	
L7	Set Acceleration Percentage L7 P#
Sets the acceleration to a percentage of the current acceleration. P# is the percentage (1-100) to set. L7 alone restores the original.	
L8	Turns Backlash Compensation On, Restore INI values for all axes.
Examples: L8XY - Turn Backlash Compensation On, Restores INI values for specified axes. L8X.001 - Turns Backlash Compensation On, Set new value for specified axis.	
L9	Turns Backlash Compensation Off.



L12	Set Axis Mapping - L12 ZW
Allows re-mapping of axis inputs. Specify the axes that should receive the input from ZW in that order. This command works like L11 except that the XY inputs are not used. Specify L12 alone to return to normal parsing. Example: L12 WZ swaps the W and Z axis, L12 ZZ moves Z and W together.	
L13	Set Axis Mapping - L13 XYZW
Allows re-mapping of axis on a pulse basis. This means when the controlling axis takes a step, the slave will follow in that direction. Specify L13 alone to return to normal mapping. Examples: L13 ZW (Z is controlling axis, W follows Z.) L13 WZ (W is controlling axis, Z follows W.) L13 XY (X is controlling axis, Y follows X.)	
L20	Enable Soft Limits - L20 [X1, X2]
Enable Soft Limit and Boundary Checking: L20 X1 enables Boundaries only. L20 X2 enables Soft Limits only. L20 alone enables both.	
L21	Disable Soft Limits
Disable Soft Limit and Boundary Checking. Soft Limits and Boundaries are used to define a cutting area which is checked during parsing of a file or command. This effectively keeps the machine from moving out of a defined area. L21 disables Soft Limit and Boundary checking.	
L22	Set Low Boundaries - L22 X# Y# Z# W#
Set Low Boundaries to values specified. If no values are given then all Low Boundaries are set to WINCNC.INI defaults.	
L22.1	Restore Low Boundaries - L22.1 XYZW
Restores default Low Boundaries to the specified axes.	
L23	Set High Boundaries - L23 X# Y# Z# W#
Set High Boundaries to values specified. If no values are given then all High Boundaries are set to WINCNC.INI defaults.	
L23.1	Restore High Boundaries - L23.1 XYZW
Restores default High boundaries to the specified axes.	

L28	Set Position - L28 X# Y# Z# W#
Sets Machine Coordinates. Machine Coordinates specify the fixed machine zero point for your machine. L28 is normally used only to write a custom homing program. L28 does not move the machine, but sets the current position to the values specified. Use G92 to set a local coordinate system for running absolute mode programs from any table position.	
L28.1	Reset Rotational Axis Between 0° and 360°
Automatically resets any rotational axis (type 4) to corrected angle between 0° and 360°.	
L29	Clear Home Flags - L29 XYZ
Clears all require home flags. Axes can be used with the command to clear require home flags only on the specified axes.	
L29.1	Ignore Home Flags
Ignores require home flags for all axes during execution of the currently running file.	
L29.2	Set G28 Requirement Flags
If the L29.2 command is run with no axis characters specified then the G28 requirement is set for all configured axis. L29.2 can also be used only for specific axis, for example L29.2 XY would set the G28 requirement flag for only the X and Y axis.	
L30	Knife Mode On
Enables support for the Knife Mode.	

L30.1	Knife Parameters L30.1 A# Z# P# C# D# R# V# B# X# Y# S#
<p>Set options for knife. When amount of change in the angle of the knife exceeds A# the knife will lift to make the turn. A# is also used for angle correction in Target Mode: see Knifemode=.</p> <p>Z# is the position to lift to.</p> <p>P# is the position that the knife will plunge to.</p> <p>C# is the auxiliary channel used to raise and lower the knife if controlled by that method.</p> <p>D# is the dwell time for the knife if controlled by an auxiliary output.</p> <p>R# is the axis number of the rotary axis (Zero Based).</p> <p>V# is the vertical axis number (Zero Based). B# is the number of degrees in an angled blade.</p> <p>X# is the X offset from center for an angled blade.</p> <p>Y# is the Y offset from the center for an angled blade.</p> <p>S# is used to create a separation offset between cuts in opposite directions.</p>	
L31	Knife Mode Off
Disables support for the Knife Mode.	
L32	Saw Mode On
Enables support for the Saw Mode.	
L33	Saw Mode Off
Disables support for the Saw Mode.	
L34	AutoAuxOut Enable - L34 D#
<p>Enables an auxiliary output that will turn on during G1/G2/G3 moves, and will turn off for all other moves. L34 D# sets the dwell time in seconds used when the channel is set to on and L35 D# sets the dwell time in seconds when the channel is set to off.</p> <p>When the 'D' parameter is used with the L34 command the "AutoAuxOut" mode does not change. L34 parameters are set in the WINCNC.INI file by the "autoaux=" line.</p>	
L35	AutoAuxOut Disable - L35 D#
<p>Disables the auxiliary output that is turned on by the L34 command. L35 D# sets the dwell time in seconds when the channel is set to off. When the 'D' parameter is used with the L35 command the "AutoAuxOut" mode does not change.</p>	

L36	Torch Height Control On.
L36.1	Auto Voltage Set Mode On.
L36.2	Reset Auto Voltage Set Status.
L37	Torch Height Control Off.
L37.1	Auto Voltage Set Mode Off.
L38	Torch On. L38 T# P#
<p>T# sets the slow start time.                      P# sets the feed rate percentage used for the slow start.                      When both T# and P# parameters are used the cutting speed of the torch is reduced by P# % for T# seconds after the initial pierce. Once T# seconds have expired the cutting speed returns to the originally specified feed rate.                      D# sets the active delay.                      C# sets cut height delay.                      A# sets the torch bevel angle.</p>	
L39	Torch Off.
L39.1	Distance for Early Off - L39.1 T#
<p>T# sets the distance (in units of measure) from to the end of the last cut where the torch will turn off. Must be used prior to the L38 command to turn the torch on, and will result in the torch turning off prior to the end of the last cut move by the distance specified. Referenced from the internal variable THQ.</p>	
L39.2	Reinitialize "AutoGas" - L39.2
<p>Reinitializes serial communication for the "AUTOGAS" function included with the torch height control feature.</p>	
L40	Feed Rate Override - L40 P#
<p>Allows user to set the feed rate override by command. P must be within the bounds specified in the INI file. If not, the command is ignored.</p>	
L41	Shell Command - L41 C:\Path\Program.exe
<p>Allows the user to run an executable program from within WinCNC.  <b>NOTE:</b> There is no error reporting for this command. If the path or filename are incorrect the command will fail without aborting.</p>	

L42	EOL Feedrate Reduction - L42 D# P#
Automatically reduce programmed feed rates for the XY axis near the end of each G1/G2/G3 move. D# is the distance from the end of each move where the feed rate will be reduced. P# is the percentage by which the feed rate is reduced.  L42 without parameters disables feedrate reduction.	
L48	Laser Height Control
Sets the target voltage to the current input voltage.	
L51	Set D/A - L51 V# P# S#
Sets the output voltage for DA. The V parameter is used to set the DA by voltage, the P to set the DA by a Percentage, and the S is used to set the DA by spindle speed. ONLY 1 PARAMETER CAN BE USED.	
L53	Set Vertical Head Height
Raises all vertical heads as high as possible while keeping the bits at equal height.	
L55	Pump Velocity Mode On
Enables velocity matched pump control.	
L56	Pump Purge Mode On
Sets pump to purge mode.	
L57	Pump Off
Stops the pump.	
L58	Pump Off with Auto Reverse
Stops the pump, using auto reverse.	
L59	Set Up Pump Max - L59 M#
Sets the pump max to the specified M parameter.	
L60	Turn Virtual Input Off - L60 C#
Turn specified virtual input off, # can be 1-64.	
L61	Turn Virtual Input On - L61 C#
Turn specified virtual input on, # can be 1-64.	

L70	Re-Enter Last Command
When L70 is used in a file, the last used command will automatically be re-entered on the command line after file completion.	
L82	Measures the Surface - L82 Z#
L82 Z# measures the surface with the probe and sets the measured point to the Z value given. L82 alone will set the measured point to Z0.	
L82.1	Sets the Surface for Probe Digitization
Sets a G92 for vertical axis based on the L82.3 calibration.	
L82.2	Moves the Surface for Probe Digitization
Adjusts G92 positions set by the last L82.1 based on a new measurement.	
L82.3	Calibrate Digitizing Probe
Calibrates, or stores, the distance from the probe contact point to the table surface.	
L83	Measures Z Depth - L83 X# Y#
L83 command will do a one-shot Z probe at the specified location, and record the XYZ position values in a comma delimited text file named SCAN.CSV. Where X# and Y# specify the (X,Y) coordinates to move to before doing the depth probe.	
L90	One Time Absolute
L90 is used in a command to specify G90 coordinates on one line without changing program mode to G90.	
L91	One Time Relative
L91 is used in a command to specify G91 coordinates on one line without changing program to G91.	



L92	Cancel G92 Restore
<p>The current G92 Local Coordinate system is saved every time a program is started or a subprogram is called using M98. When the program is finished, the original G92 system is restored. This is not desirable if the program is being used to preset a system configuration for other programs. L92 causes the normal G92 coordinate restore to be skipped after the program containing the L92 is finished.</p>	
L101	Tool Change - L101 T# R#
<p>Sets the tool number and tool measure offset for ATC1. T# specifies the tool number to load. R# specifies the tool measure storage register to use.</p>	
L102	Tool Change - L102 T# R#
<p>Sets the tool number and tool measure offset for ATC2. T# specifies the tool number to load. R# specifies the tool measure storage register to use.</p>	
L110	Tool Change - L110 T#
<p>Sets current tool number for machines with an automatic tool changer. The tool number is available as {TP1} in subsequent commands, or {TN1} and {TN2} for dual tool changer machines. The stored tool measure is reloaded for the new tool. (See M37 command for storing tool measures)</p>	
L111	Test Tool Number - L111 T# A#
<p>Run next line if tool number is equal to T#. A# - command used for checking the current tool number loaded in the ATC. 'A0' should be used to check the tools in ATC1 and ATC2. 'A1' should be used to check just the ATC1. 'A2' should be used to check ATC2.</p>	
L111.1	Test Tool Number with Dual Tool Changers - L111.1 T#
<p>Run next line if both tool numbers are equal to T#.</p>	
L112	Test Tool Number - L112 T#.
<p>Skip next line if tool number is equal to T#.</p>	

L113	Test to see if V# in range - L113 L# H# V#
Skip the next line if the V# is not greater than the L# and less than the H#.	
L120	Tool Change - L120 T#
Sets current tool number for machines with an automatic tool changer. The tool number is available as {TP2} in subsequent commands.	
L201	Set 201 Signal Command - L201 FileName T# C# S#
Sets the command used with the 201 signal type.	
L202	Set 202 Signal Command - L202 [Command]
Sets the command used with the 202 signal type.	
L203.2	Job Counter - L203.2 T# A#
T# the job counter is set to the specified value. A# the specified value is added to (or subtracted from) the job counter.	
L204	Activate Signals and Enables - L204 C#
L204 C# activates signals and enables the specified channel. L204 activates all signals and enables.	
L205	Deactivate Signals and Enables - L205 C#
L205 C# deactivates signals and enables the specified channel. L205 deactivates all signals and enables.	
L210	Select Alternate Low Limits - L210 XYZW
Selects axis to use alternate low limits.	
L210.1	Select Alternate Low Limit Pin - L210.1 XYC# *
Selects alternate low limit pin from the specified C# channel. If no channel is specified, the original settings are restored. *See <b>exaltaxislo=</b>	
L211	Select Alternate High Limits - L211 XYZW
Selects axis to use alternate high limits.	

L211.1	Select Alternate High Limit Pin - L211.1 XYC#
Selects alternate high limit pin from the specified C# channel. If no channel is specified, the original settings are restored. *See <b>exaltaxishi=</b>	
L212	Select Primary Limits
Selects the primary limits for all axes.	
L213	Dual Axis, Select Main - L213 XYZW
Selects the main motor on a dual motor axis.	
L214	Dual Axis, Select Secondary - L214 XYZW
Selects the secondary motor on a dual motor axis.	
L215	Dual Axis, Select Both Axes - L215 XYZW
Selects both motors on a dual motor axis.	
L249	Laser Power Setting - L249 P# I#
Using the L249 command allows you to specify the laser power. P# - Sets laser power. Example: L249 P50 (sets laser power to 50). I# - reverses color polarity. Can fire on black or white.	
L250	Vector Laser Off
Disables vector laser cutting.	
L251	Vector Laser On - L251 P# D# L#
Enables vector laser cutting. P# - laser power (1-100). D# - sets the distance for the distance based vector laser firing. L# - sets the distance for the off signal when perforating.	
L251.1	Laser Dwell - L251.1 D#
D# - time in seconds for dwell.	
L252	Bitmap Engraving
L252.1 Stores rotation for future use.	
L253	Bitmap Engraving rotated 90 degrees clockwise.
L253.1 Stores rotation for future use.	

L254	Bitmap Engraving rotated 180 degrees clockwise.
L254.1 Stores rotation for future use.	
L255	Bitmap Engraving rotated 270 degrees clockwise.
L255.1 Stores rotation for future use.	
L256	Offset for bitmap engraving row alignment L256 R#
Offset on positive motions to compensate for laser firing delays to line up bitmap rows during dual motion engraving. R# sets the radius used for rotary engraving in the current unit of measure.	
L300	Lock Axis - L300 XYZW
Locks specified axis motion. Any motion commanded to a locked axis will be ignored.	
L301	Unlock Axis - L301 XYZW
Unlocks specified axis motion.	
L400	Create Home File
Creates a home file at the machine's current position.	
L401	Recall Positions
Moves all axes to the positions where they were at previous to the last file ran.	
L402	Save current positions for all axes to a specified text file - L402 FILENAME.TXT
Moves all axes to the positions where they were at previous to the last file ran.	
L500	Store Absolute Positions - L500 [X, Y, Z...] [P H T] L# C# N"Name"
<p>[X, Y, Z...] - axes to set positions for, if none specified then all axes positions are stored.</p> <p>[P H T] - type of position (P)osition, (H)ome Position, (T)ool Position. Can only use one value.</p> <p>L# - 0 no vertical head lift, 1 include head lift.</p> <p>C# - Clearance for vertical heads if L1.</p> <p>N"Name" - Name for specified position. &lt; 16 characters long.</p>	

L500.2	- L500.2 T# N# A# C# O# H# M#
<p>T# specifies the tool number.  N# specifies the nominal diameter for the tool.  A# specifies the actual diameter for the tool.  C# specifies the diameter compensation for the tool.  O# specifies the height offset for the tool.  H# specifies the tool height.  M# specifies the maximum current allowed for the tool.  This command sets up a tool library instead of using the dialog boxes.</p>	
L501	Store Local Positions - L501 [X, Y, Z...] [P H T] L# C# N"Name"
<p>[X, Y, Z...] - axes to set positions for, if none specified then all axes positions are stored.  [P H T] - type of position (P)osition, (H)ome Position, (T)ool Position. Can only use one value.  L# - 0 no vertical head lift, 1 include head lift. Default L1.  C# - Clearance for vertical heads if L1.  N"Name" - Name for specified position. &lt; 16 characters long.</p>	
L502	Set material size and position on the G-Code viewer - L502 X# Y# W# H#
<p>X# sets the material origin X position.  Y# sets the material origin Y position.  W# sets the material width (X axis).  H# sets the material height (Y axis).  NOTE: Calling L502 with no parameters can be used to toggle the material view on/off.</p>	
L600	Disable Mapping Adjustments
<p>Disable mapping adjustment for the specified axis. If no axis is specified, mapping adjustments for all axes are disabled.</p>	
L601	Enable Mapping Adjustments
<p>Enable mapping adjustment for the specified axis. If no axis is specified, mapping adjustments for all axes are enabled.</p>	

L602	Start Mapping Setup {XX} {XY} ... M#
<p>Starts mapping setup for specified axis. First axis specifies the axis to map and the second axis specifies the axis to adjust. The axis map will automatically move to the start map position. No axis spec stops the mapping setup.</p> <p>M# - M1 (default) initiates auto-move to first mapping point. M0 disables auto-move.</p>	

L603	Store Current Adjustment and Move - L603 {NPSE}
<p>Stores the current adjustment and moves the map axis to the specified map increment. N is next, P is previous, S is start, and E is end. The default move is to the next map increment position.</p>	

L604	Move Map Axis By Increment - L604 {NPSE}
<p>Moves the map axis to the specified map increment. N is next, P is previous, S is start, and E is end. The default move is to the next map increment position.</p>	

L605	Store Current Adjustment
<p>Stores the current adjustment.</p>	

L606	Store a adjustment for a specific mapping point of current mapping axis - L606 N# V#
<p>N# is the mapping point (must be between start and end for the mapping axis)</p> <p>V# is the mapping adjustment to store at the specified mapping point.</p> <p>*Note: Requires a previous L602 call.</p>	

<b>Surface mapping</b>	
<p>L610 - L615 are codes used to create a surface map of your cnc table surface. Before running these commands, a local zero (G92Z) must be set as a reference Z0. See also <b>surfacemap=</b></p>	

L610	Disable surface mapping - L610
<p>L610 Disables the surface mapping.</p>	

L611	Enable surface mapping - L611
<p>Enables surface mapping.</p>	

L612	Move to first XY map point - L612
Move XY to the X and Y coordinates specified by the X# Y# on the surfacemap= line in the wincnc.ini file.	
L613	Store vertical axis adjustment - L613
Store vertical axis adjustment from preset G92Z.	
L614	Move to the next XY map point - L614
Move XY to the next incremental surface map coordinate.	
L615	Store offset for specified map point - L615 X# Y# A#
Store an offset X axis position X#, Y axis position Y#, offset amount A#	
L700.1	Laser Scan Setup - L700.1 A# R# X# Y# F#
Sets the axis numbers (scan/rotary), scan resolutions (dpi), and scan velocity for laser probe digitizing. A# - Axis number (0=X, 1=Y, ..) for primary scanning axis. R# - Axis number (0=X, 1=Y, ..) for rotary scanning axis. X# - Points per unit for scanning along the 1st dimension. Y# - Points per unit for scanning along the 2nd dimension. F# - Units per minute for scanning passes.	
L700.2	Laser Scan Type - L700.2 T# S# A#
T# - Type. 1 for surface scanning and 2 for rotary scanning. S# - If 1 then the current positions of both scan axis are stored for start positions. E# - If 1 then the current positions of both scan axis are stored for end positions.	
L700.3	Optimet Laser Scan Dialog - L700.3 D# Z#
Calls the Optimet laser probe dialog and/or to set the Z-zero position for data output by the laser probe digitizing. Syntax is as follows: D# - If non-zero the Optimet laser probe dialog box is displayed. Z# - If non-zero a one-shot measurement is taken and used for Z-zero.	

L700.4	Set Active Optimet Lens Index - L700.4 L#
<p>Changes the active lens for the Optimet laser digitizing hardware.                  L# - The index (zero based) of the lens to use, in reference to the lens choices available from the drop down list in the Optimet Laser Scan software.</p>	

L800	Disable Probe
<p>Disables the touch probe.</p>	

L801	Enable Probe
<p>Enables the touch probe.</p>	

When probe is enabled (L801) it acts similarly to a limit switch. When probe comes in contact with an object, it will stop motion. If it is disabled (L800) probe no longer functions.

L802	Scan Given Rectangle - L802 X# Y# I#
<p>Scans a given rectangle XY with an increment of I in and up/down, and side-to-side style scan. (Use automatic scan)</p>	

L803	Scan Given Rectangle - L803 X# Y# I#
<p>Scans a given rectangle XY with an increment of I in an up/down style scan.</p>	

L810	Scan Slice (Drop-off Detection) - L810 FILENAME. EXT X# Y# I# K#
<p>Digitizes a slice (line), either in X direction, or Y direction and saves the points to a text file. If a drop-off is encountered, or distance stated in X or Y is reached, the slice is saved and the scan is ended. <b>Only one axis can be scanned at a time: X or Y.</b> Uses up/down, and side-to-side probe type.                  FILENAME.EXT - replace with your filename. <b>Do not surround with quotation marks.</b>                  X# - is the actual X axis distance that you want the probe to travel looking for an edge. If it finds one, the file is saved and the scan ends.                  Y# - works like X#, except on the Y axis.                  I# - is the maximum increment for the horizontal axis.                  K# - is the vertical increment (how high the probe raises).</p>	

L820	Disable axisadj
<p>Disables automatic axis position adjustment.</p>	



L821	Enable axisadj
Enables automatic axis position adjustment.	
L901	Limit Switch Debounce - L901 D#
Changes the limit switch debounce. D# can be 1-30.	
L910	Disable G-Code Line Display
Disabling G-Code display when running large G-Code files can reduce CPU and memory usage resulting in better overall computer system performance.	
L911	Enable G-Code Line Display
Enabling G-Code display when running large G-Code files can increase CPU and memory usage resulting in worse overall computer system performance.	

**M Codes**

M11	Auxiliary Output On - M11 C#
Controls auxiliary outputs. Channel specified by # is turned on. Port and bit addresses are set up in WINCNC.INI file. <b>Note: For use with a Drill Bank</b> A d# can be added to the M11 command for use with a drill bank. The d# is the binary number of channels up to the n# specified in the drill bank line to turn on. Example: M11D7 would turn on the first 3 channels of a ten channel drill bank.	
M12	Auxiliary Output Off - M12 C#
Controls auxiliary outputs. Channel specified by # is turned off. Port and bit addresses are set up in WINCNC.INI.	
M13	Auxiliary Output On - M13 C#
Same as M11, except the output is turned on without pausing motion.	
M14	Auxiliary Output Off - M14 C#
Same as M12, except the output is turned off without pausing motion.	

M17	Pause Until Input On - M17 C#
Wait for channel specified by # to come on. Port and bit addresses are set up in WINCNC.INI.	
M17.1	Run Next if Input On - M17.1 D# C#
Run next command only if channel specified by # is on. Port and bit addresses are set up in WINCNC.INI. D# can be 0 or 1. D0 sets the simulate logic to false. D1 sets the simulate logic to true.	
M17.2	Run Next if AuxOut is On - M17.2 D# C#
Run the next command only if channel specified by the # is on. Port and bit addresses are set up in WINCNC.INI file. D# can be 0 or 1. D0 sets the simulate logic to false. D1 sets the simulate logic to true.	
M17.3	Run Next if Virtual Input On - M17.3 C#
C# is a "virtual input" channel, will run the next line if the specified "virtual input" channel is on and skip the next line if the channel is off. # can be 1-64.	
M17.4	Run Next if C# is 1, Otherwise Skip. M17.4 C#
C# can be 0 or 1. 1 will run the next line, 0 will skip the next line.	
M18	Pause Until Input Off - M18 C#
Wait for channel specified by # to go off. Port and bit addresses are set up in WINCNC.INI.	
M18.1	Run Next if Input Off - M18.1 D# C#
Run next command only if channel specified by # is off. Port and bit addresses are set up in WINCNC.INI. D# can be 0 or 1. D0 sets the simulate logic to false. D1 sets the simulate logic to true.	
M18.2	Run Next if AuxOut is off - M18.2 C#
C# is an "auxiliary output" channel, will run the next line if the specified output channel is off and skip the next line if the channel is on. # can be 1-64.	

M18.3	Run Next if Virtual Input is Off - M18.3 C#
C# is a “virtual input” channel, will run the next line if the specified “virtual input” channel is off and skip the next line if the channel is on. # can be 1-64.	
M18.4	Run Next Line If C# is 0, Otherwise Skip - M18.4 C#
C# can be 0 or 1. 0 will run the next line, 1 will skip the next line.	
M28	Disable Limit Abort
M28 is used to write your own home program instead of using G28 to home the machine. WinCNC normally aborts a program when a limit switch is encountered. Adding M28 to a G0 or G1 command disables the program abort and instead stops the limited axis while allowing other axes to continue. Limit Cushions are disabled while using M28 so any move done with M28 should be done slowly to avoid machine damage. M28 must be specified on each line separately.	
M29	Disable Limit Abort
M29 is used to write your own home program instead of using G28 to home the machine. Adding M29 to a G0 or G1 command disables the program abort and instead stops the current G-Code line but continues on the following line. M29 must be specified on each line separately.	

M37	Measure Tool - M37 O# Z# W# H# T#
<p>M37 is used to set Tool Length Offsets. The end of the tool is first set to a known position. M37 is then specified to set the tool length offset.</p> <p>O# manually specifies the tool height offset.</p> <p>H# is the tool number of the currently loaded tool. Use this with mode 0 only, M0 on the ATC line in the WinCNC.ini file.</p> <p>T# is the tool number of the currently loaded tool. Use this with mode 1 only, M1 on the ATC line in the WinCNC.ini file.</p> <p>Example: Jog the Z head to the spoil board. Enter M37Z0 to the Z work coordinate to 0. Once tool length offsets are set, Boundaries may be used to insure that the table is not routed by accident. Normally zero values are set to the top of the spoil board. Use G92 to set material thickness if programs reference zero from the top of material. This eliminates re-measuring tool for different thickness materials. G43 and G49 are used to enable and disable offsets.</p>	
M37.1	Measure Tool for Calibrating G37 - M37.1 Z# W#
<p>M37.1 is used to store a tool measure in the TM1 variable. This command is used primarily with calibrating G37 for automatic tool measuring.</p>	
M37.2	Measure Tool for Calibrating G37 - M37.2 Z# W#
<p>M37.2 is used to store a tool measure in the TM2 variable. This command is used primarily with calibrating G37 for automatic tool measuring.</p>	
M37.3	Shift Workspace Offsets M37.3 S# H# T# M# X# Y# Z# ...

M37.3 is used to shift workspace offsets for tool measures, tool measure switch heights, or individual axis.  
S# is the shift amount to be used for tool measures and tool measure switch positions (usually a negative #).  
H# is the H register number to be shifted by the shift amount. H0 will shift all H registers.  
T# is the tool number height to be shifted by the shift amount. T0 will shift all tool numbers.  
M# is the measure switch height to be shifted by the shift amount. M0 will shift both switches.  
X# Y# Z# ... shifts each axis workspace offsets by the specified value.

M98	Subprogram Call - M98 <filename>
-----	----------------------------------

Run a subprogram. A subprogram may be any G-Code file. Control is passed back to the calling program when finished. To run a G90 mode subprogram from any position on the table use G92X#Y#Z#W# to set the local coordinates at the beginning of the file.

M98.1	Subprogram Call - M98.1 <filename>
-------	------------------------------------

Same as M98, except line numbers in the subprogram are not counted. This is most useful for macros that call subprograms.

M99	End Subprogram
-----	----------------

Ends a subprogram.

M99.1	Abort
-------	-------

Initiates a commanded program abort.

This page is intentionally left blank.

**Macro File Configuration**

The CNC.MAC file allows the user to create his or her own commands already built into WinCNC. If macros are not used, a blank text file should be created in the installation directory, and named CNC.MAC.

If the user wanted to create shortcut commands to turn auxiliary output channel 2, which controls the spindle power, on and off the entries to the CNC.MAC file might be as follows:

M1 = M11C2	[SPINDLE ON]
M2=M12C2	[SPINDLE OFF]

Macro commands defined in the CNC.MAC file can only contain a single command. If multiple commands are needed within a macro, you can define a macro that calls a subroutine. Using a subroutine, you can call multiple commands. The following example shows how to set a macro to call a subroutine and an example of what that subroutine would contain. This example demonstrates a macro that calls a subroutine that will turn on a spindle and wait for it to get up to speed before moving.

Macro Define:

```
M3=M98 SPIN.MAC [SPINDLE ON]
```

Subroutine:

```
M11C1 [SPINDLE ON]
```

```
M17C10 [WAIT FOR SPEED UP]
```

Another useful feature of the CNC.MAC file is to disable unwanted M-Codes output by some CAD/CAM packages. WinCNC will ignore disabled codes. Following is an example of how to disable an unwanted M2 code.

```
M2=[
```

Also the CNC.MAC file allows the user to override command names which have already been built into WinCNC. Once you override a built in command you will have to use a tilde (~) any time you want to use the built in function. One example of this would be if a user wanted to use the command G28 to return to 0 on axis XYZ in that order. The CNC.MAC file entry to accomplish this would be: Macro Define:

## G28=M98 HOME.MAC

This sets the WinCNC G28 command to call the macro instead of its built in function.

In the HOME.MAC file the tilde (~) tells the controller that the built in command has been renamed, and must override the renamed assignment and run its built in function. Keep in mind that by declaring this macro the built in G28 command will be disabled and cannot use its built in function without calling it with a tilde (~). The HOME.MAC file would look similar to this.

Subroutine:

~G28Z [Return to 0 on Z axis]

~G28X [Return to 0 on X axis]

~G28Y [Return to 0 on Y axis]

Note: The .MAC extension of the subroutine file names is NOT required, except for the CNC.MAC file. That extension can be used if desired, but was used here only as an example.

There is command that can be set in the CNC.MAC file called “**define=**”. It is used to define variables and the syntax is as follows:

<b>define=</b>	AAA, #
<p>The first portion of the define (<b>AAA</b>) can be any combination of alpha or numeric characters. The second portion (after the comma) <b>#</b> defines the value and must be an integer or decimal numeric value. This allows a custom variable to be defined for use as a parameter in a subprogram using braces {}. For example, the following define could be added to the CNC.MAC.</p> <pre>DEFINE=ZU1, 3.25</pre> <p>The following statement could be used in a subprogram to set the Z axis to a specific position.</p> <pre>G53 Z{ZU1}</pre>	
<p>In order to do math with the define parentheses must be used around the variable.</p> <pre>G53 Z{{(ZU1-2)}}</pre> <p>In the example above, 2 will be subtracted from the variable value to equal a final value of 1.25.</p>	



# **WinCNC Controller      Messages and Error Codes**

## **Messages and Error Codes**

### **Error - Line Too Long**

Input line more than 256 characters.

### **Error - Unsupported G Code**

G Code in input line which is not supported.

### **Error - Unsupported L Code**

L Code in input line that is not supported or ignored.

### **Error - Unsupported M Code**

M Code in input line which is not supported.

### **Error - Multiple Commands**

Input line contains more than one command.

### **Error - Syntax - Command or File Not Found**

Non-supported code or invalid file name entered.

### **Error - Arc Radius**

Distance from start point to center is not equal to distance from endpoint to center. Arc Radius errors can occur from running a G2 or G3 intended for G91 mode in G90 mode and vice versa.

### **Error - Soft Limit Exceeded**

Move specified would result in the head being moved outside the soft limits.

### **Error - Boundary Exceeded**

Move specified would result in the head being moved outside the workspace boundaries.

### **Error - Acceleration Out of Bounds**

Acceleration rate too high for resolution. Maximum (acc\*res) in steps is 65535.

### **Error - Illegal Value**

The value specified is invalid.

### **Error - Unknown Position Specified**

The position specified is invalid.

## **WinCNC Controller      Messages and Error Codes**

<b>Error - Subprogram Nesting Too Deep</b>
Too many nested subprogram calls.
<b>Error - L10 in Subprogram</b>
L10 command found in subprogram.
<b>Error - File Not Found</b>
The specified file name cannot be found.
<b>Error - Parameter Not Specified</b>
A required parameter is not specified.
<b>Error - Unknown</b>
An unknown error has occurred.
<b>Error - Memory Error</b>
Insufficient or bad memory.
<b>Error - User Level 5 Required</b>
A Level 5 user license is required.
<b>Error - Invalid L10 Command</b>
L10 command has invalid parameters.
<b>Error - Invalid P1 Command</b>
P1 command has invalid parameters.
<b>Error - Command Not Implemented</b>
This command has not yet been fully implemented.
<b>Error - Tool Changer Not Enabled</b>
Tool changer is not set up in the WINCNC.INI file.
<b>Error - Parameter Out of Range</b>
Specified parameter is out of range.
<b>Error - D/A Not Enabled</b>
D/A channel is not set up in the WINCNC.INI file.
<b>Error - Arc Too Small</b>
Specified arc move is too small.

## **WinCNC Controller      Messages and Error Codes**

<b>Error - Device Not enabled</b>
The requested device is not enabled.
<b>Error - Could Not Open File</b>
Requested file could be opened. May be in use by another program.
<b>Error - Pump Control Not Enabled</b>
The pump control feature has not been enabled.
<b>Error - Soft Limits Not Enabled</b>
Softlimits are set as disabled.
<b>Error - Tangential Knife Not Enabled</b>
The tangential knife feature is not enabled.
<b>Error - Forced Limit Exceeded</b>
Move specified would result in the head being moved outside the forced limits.
<b>Error - Digitizer Probe Not Enabled</b>
The digitizer probe feature has not been enabled.
<b>Error - Arcs Disabled When X or Y scaled</b>
When scaling is enabled, arcs are not supported.
<b>Error - Rotating Saw Not Enabled</b>
The rotating saw feature is not enabled.
<b>Error - Macro Expansion Error</b>
Occurs when macro commands exceed 256 characters.
<b>Aborted - Limit Switch</b>
Limit switch encountered while running.
<b>Aborted - Aborted By User</b>
Escape button pushed while running.
<b>Key Not Found</b>
The software key cannot be found.

# WinCNC Controller Messages and Error Codes

<b>Wrong Key Found</b>
The software key found is incorrect.
<b>Key Error #</b>
A software key read error has occurred.
<b>Errors Reading WINCNC.INI</b>
The WINCNC.INI file contains errors or is not found.
<b>Errors Reading CNCSCRN.INI</b>
The CNCSCRN.INI file contains errors or is not found.
<b>CNC.MAC Not Found</b>
The CNC.MAC file is not found.
<b>Upgrade needed for Key Version 5, or 6 Axis card</b>
And upgrade of the daughter board is needed.
<b>Pausing.....</b>
Motion is pausing.
<b>Continuing.....</b>
Motion is continuing.
<b>Finished Running "Filename"</b>
The file "Filename" is complete or has been aborted.
<b>Runtime: #d Minute(s) and #f Second(s)</b>
Run time for the last file in minutes and seconds.
<b>File Not Found</b>
A specified file cannot be found.
<b>Error - Demo Count Exceeded</b>
Demo mode only. 25 commands have been completed.
<b>Error - Helical Interpolation Not Supported</b>
No Z or W moves during an arc.
<b>Escape Button Pushed While Running</b>
User has aborted motion with the ESC key.

## WinCNC Controller Messages and Error Codes

### **Error - Not a BMP File**

Tried to open a non-bmp file for bitmap engraving.

### **Error - BMP File Invalid Header**

Tried to open an incompatible bmp file.

### **Error - 2-Bit Bitmap Engrave Not Enabled**

Tried to open a 2-Bit bitmap without the feature being enabled in the software.

### **Error - 8-Bit Bitmap Engrave Not Enabled**

Tried to open an 8-bit bitmap without the feature being enabled in the software.

### **Error - Vector Laser Not Enabled**

Vector laser feature not enabled in the software.

### **Error - Auto-AuxOut Not Enabled**

User has tried to use an Auto-AuxOut without the "autoaux=" line being set in the WINCNC.INI file.

### **Error - Depth Must Be Lower Than Retract**

Tried to retract a tool to a height that was below current depth.

### **Error - Unknown**

An error has occurred that is undefined in WinCNC.

This page is intentionally left blank.

**Keypad**

There are four types of keypads that can be used with the WinCNC Controller. Three are connected by serial cable, one is wireless.

Type one is an input only keypad which is a serial number pad, just like the one found on the far right of a standard keyboard. Each key can be configured individually in the CNCSCRN.INI file, using the ScanOn and ScanOff parameters. The type two keypad is an input/output keypad with a 20x4 LCD screen display. When this type of keypad is defined X, Y, and Z axis information is displayed on the keypad screen. This keypad can also be configured to browse and select files from the directory last used by the WinCNC Controller. Configuration for this keypad is done in the CNCSCRN.INI file, just as for keypad type one. To use the file browsing capabilities of keypad type two you will need to set up three "KeyPadOut" buttons, as shown in the previous CNCSCRN.INI Settings section. The type 3 is the 2Tech brand keypad with LCD display and optional emergency stop. This keypad would be configured the same way as the type 2 keypads. The type 4 is the XHC wireless keypad with MPG wheel. These keypads feature configurable buttons and the ability to transit and jog the machine, adjust feed rate and spindle speed, and repeat jobs. Configuration for this keypad is also done in the CNCSCRN.INI file.

**NOTE:** When defining keypad buttons it is important to set the ScanOff code to zero, only for buttons that you wish to use with the keypad. If this is not done properly a button can be pressed, but will not be released. When configuring the keypad for use with jog or transit keys, this is particularly important because the motion will not stop when the key is released. It is recommended that the keypad setup is done after you have verified that all limit switches are working properly.

Below is a list of scan codes for all types of compatible keypads.

### Keypad Type 1: without LCD

WINCNC.INI Setting: keypad=c[com port #] b1200 p1

Key	OnCode	OffCode
1	96	64
2	97	65
3	98	66
4	99	67
5	100	68
6	101	69
7	102	70
8	103	71
9	104	72
ESC	105	73
/	106	74
*	107	75
-	108	76
+	109	77
.	110	78
0	111	79
ENTER	116	84
NUMLOCK	121	89



## Keypad Type 2: with LCD

WINCNC.INI Setting: keypad=c[com port #] b9600 p2

Key	OnCode	Shift+Key OnCode
T1	1	51
T2	2	52
T3	3	53
T4	4	54
Left Arrow	5	55
Right Arrow	6	56
Up Arrow	7	57
Down Arrow	8	58
ESC	9	59
1	10	60
2	11	61
3	12	62
SPACE	13	63
4	14	64
5	15	65
6	16	66
.	17	67
7	18	68
8	19	69
9	20	70
DEL	21	71
0	22	72
ENTER	23	73

**Note:** OffCodes on Keypad Type 2 with LCD are all 0. The “Shift+Key OnCode” refers to holding the Shift Key on the keypad down while pressing the corresponding key.

**Keypad Type 3: with LCD**

Below is the keypad layout for a Type 3, 2Tech keypad.



The following table represents the number codes associated with the appropriate keys on the above keypad without holding CTRL or SHIFT.

17	18	19	20	21
65	66	67	68	69
70	71	72	73	74
75	76	77	78	79
80	81	55	56	57
82	83	52	53	54
84	85	49	50	51
86	87	88	48	CTRL/NA
89	90	32	13	SHFT/NA

**Keypad Type 3: with LCD continued**

The following table represents the number codes associated with the appropriate keys on the keypad on the previous page while holding CTRL.

65	66	67	68	NA
1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	NA	NA	NA
18	19	NA	NA	NA
20	21	NA	NA	NA
22	23	24	NA	CTRL/NA
25	26	NA	NA	SHFT/NA

The following table represents the number codes associated with the appropriate keys on the keypad on the previous page while holding SHIFT.

6	7	8	9	10
43	40	41	60	62
45	33	95	91	93
47	38	58	59	63
42	64	55	56	57
61	35	52	53	54
94	36	49	50	51
44	37	46	48	CTRL/NA
34	27	8	13	SHFT/NA

## Keypad Type 3: with LCD continued

In most cases, your 2Tech keypad will already be configured properly. In rare cases, it will need to be set up.

If the keypad isn't working, first check the "keypad=" line in the WINCNC.INI file.

```
keypad=c[com port #] b9600 p3
```

**[com port#]** is the com port number assigned to it in the "Ports" setting of the Device Manager.

If the line is correct and the keypad is still not functioning, it must be configured.

**CAUTION:** Only do this if all else fails. If you lose configuration and can't get this to work, you will have to return the keypad to Microsystems to reset the configuration.

To make changes to the keypad...

Hold CTRL and SHIFT while pressing XPOS. This will put it into configuration mode. In configuration mode, the display addresses F1 - F5. These are the buttons across the top of the keypad: F1 = XPOS, F2 = YPOS, F3 = ZPOS, F4 = WPOS, F5 = UPOS.

To cycle through things that can be set, press F2 (YPOS) to cycle to the next item. When you get to the item you need to change, click F1 (XPOS) to change the value. Cycle to the next item that must be changed, click F1 to change the value, etc. Do not change anything that is not listed below. When KNP, ESC MODE, and BREAK CMD are set to their correct values, click F5(UPOS) to save. The keypad is set up and ready to use.

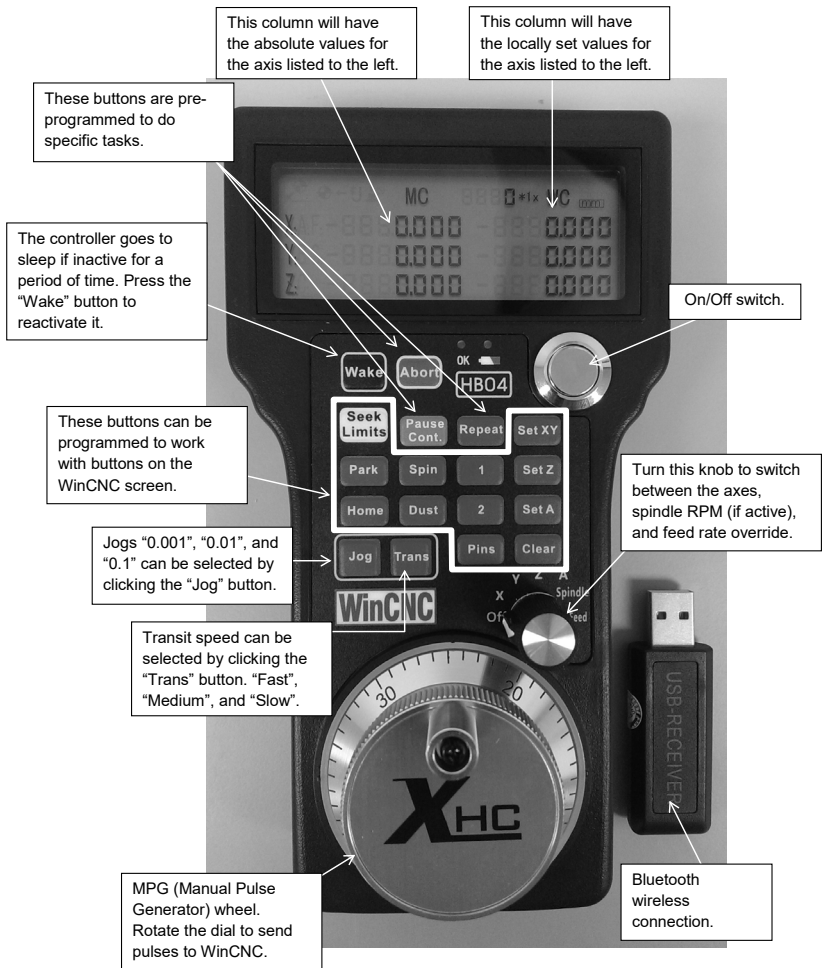
Things that need to be set, and the correct settings for each:

Set **KNP = ENABLED**

Set **ESC MODE = PRIVATE**

Set **BREAK CMD = ENABLED**

## Keypad Type 4: XHC HB04 Wireless with LCD



The XHC HB04 is a wireless pendant with a range up to 40 meters. It has twelve programmable keys that can be set to operate functions and/or run macros in WinCNC. With the MPG (Manual Pulse Generator) wheel, you can control any of up to four axes, either in Jog or Transit mode. You can also use the MPG wheel to control feedrate override and spindle speed (if the machine is so equipped). Rotate the selection knob to the desired axis, spindle speed, or feed rate and use the wheel to change the values.

Below is the button layout and numbering for setup of the custom buttons on the XHC keypad in WinCNC.



The large numbers (1 - 12) represent the codes to add to the scanon and scanoff variables in the CNCSCRN.INI file. Here is an example:

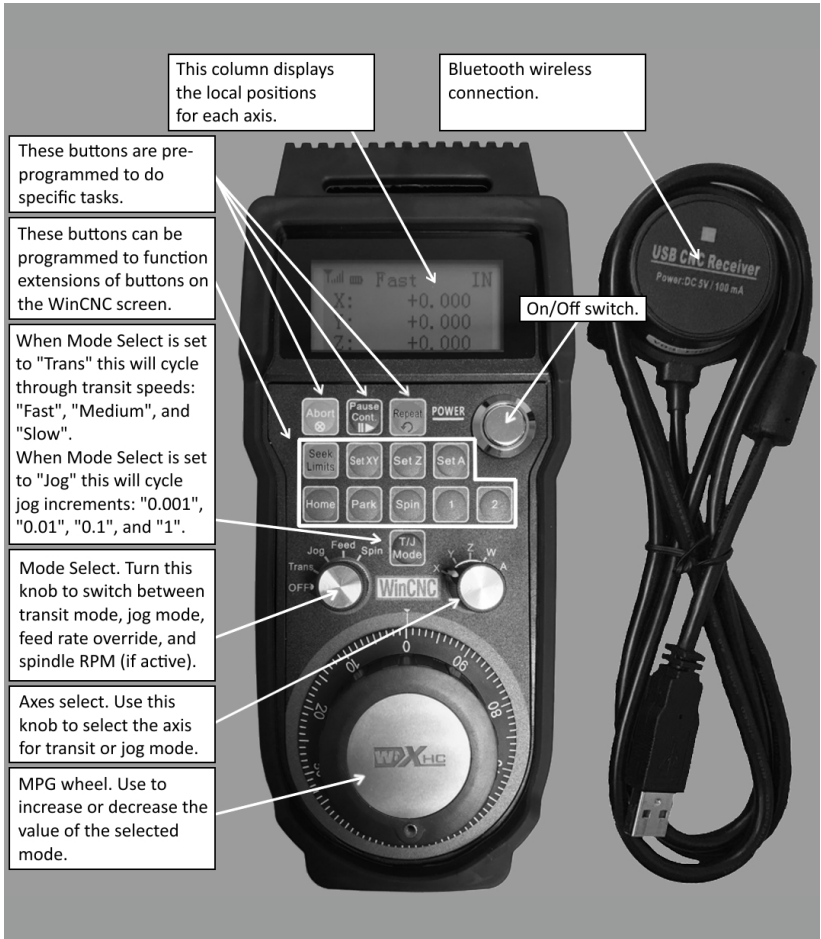
```
"UserBtn",7,4,115,39,1,0 ,"Seek Home ","G28","",1,0,"Home.bmp",2
```

The gray number "1" is the button number and the "0" is necessary to turn it off after the button is pressed.

The XHC wireless pendant requires a feature enable by your machine manufacturer or by Microsystems World CNC LLC.

Also, an "XHC=" entry must be added to the WINCNC.INI. See the "XHC=" entry in the System Configuration section of this manual for more information.

## Keypad Type 4: XHC PHB04B Wireless with LCD



The XHC PHB04B is a wireless pendant with a range up to 40 meters. It has nine programmable keys that can be set to operate functions and/or run macros in WinCNC. With the MPG wheel, you can control any of up to five axes, either in Jog or Transit mode. You can also use the MPG wheel to control federate override and spindle speed (if the machine is so equipped). Rotate the mode selection knob and axes selection knob to the desired mode and axis, and use the wheel to change the values.



Below is the button layout and numbering for setup of the custom buttons on the XHC keypad in WinCNC.



The large numbers (1 - 12) represent the codes to add to the scanon and scanoff variables in the CNCSCRN.INI file. Here is an example:

```
"UserBtn",7,4,115,39,1,0,"Seek Home","G28","",1,0,"Home.bmp",2
```

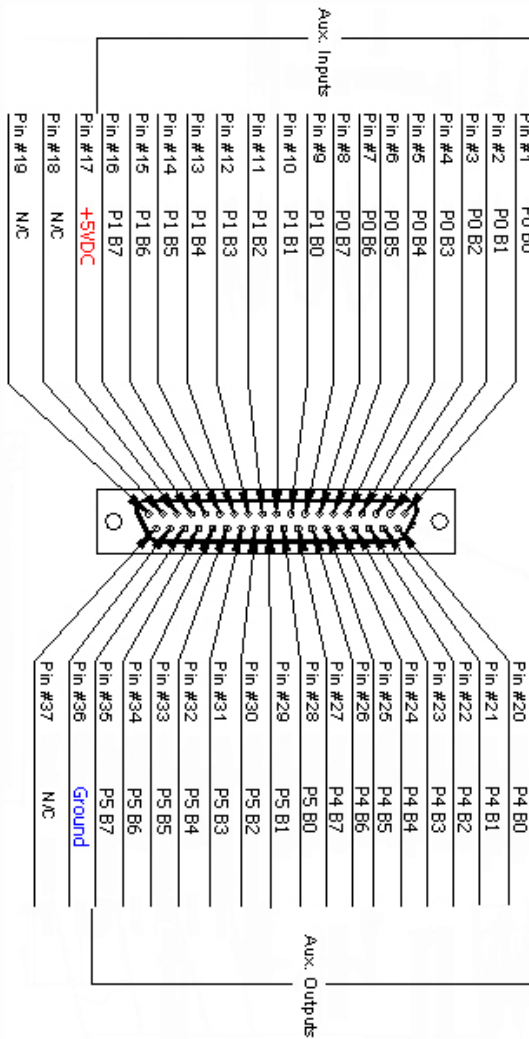
The gray number "1" is the button number and the "0" is necessary to turn it off after the button is pressed.

The XHC wireless pendant requires a feature enable by your machine manufacturer or by Microsystems World CNC LLC.

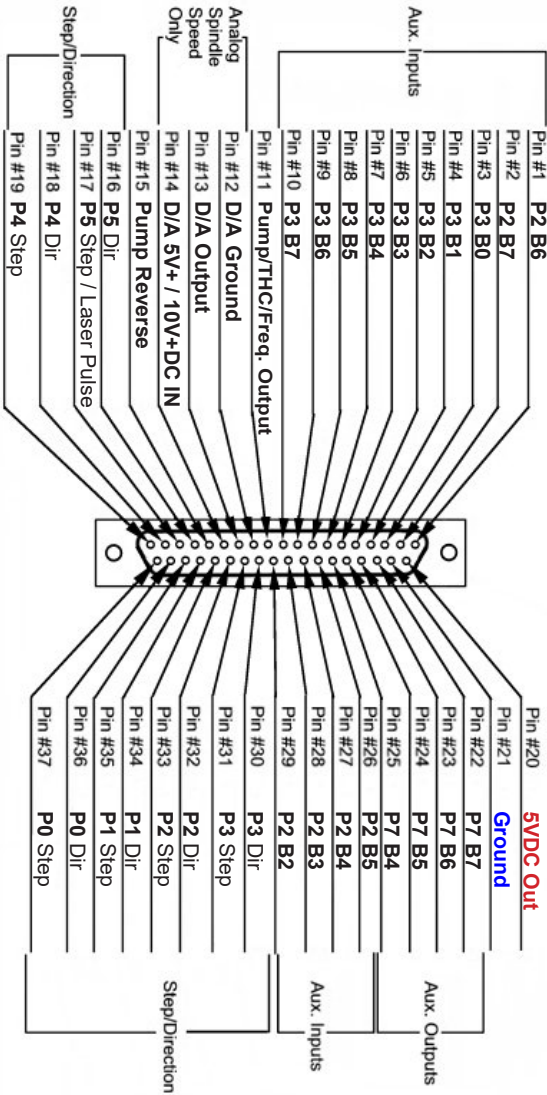
Also, an "XHC=" entry must be added to the WINCNC.INI. See the "XHC=" entry in the System Configuration section of this manual for more information.



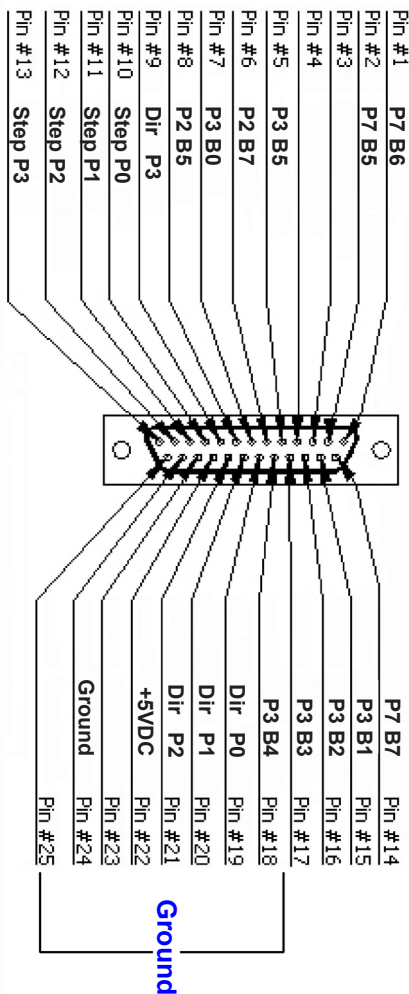
## PCI 7200 37 Pin Connector CN1:



**6-Axis 37-Pin Daughter Board Connector CN2:**



## 4-Axis 25-Pin Daughter Board Connector CN2:



6 Axis Dual DB25 C2 - 3 Axis Connector:

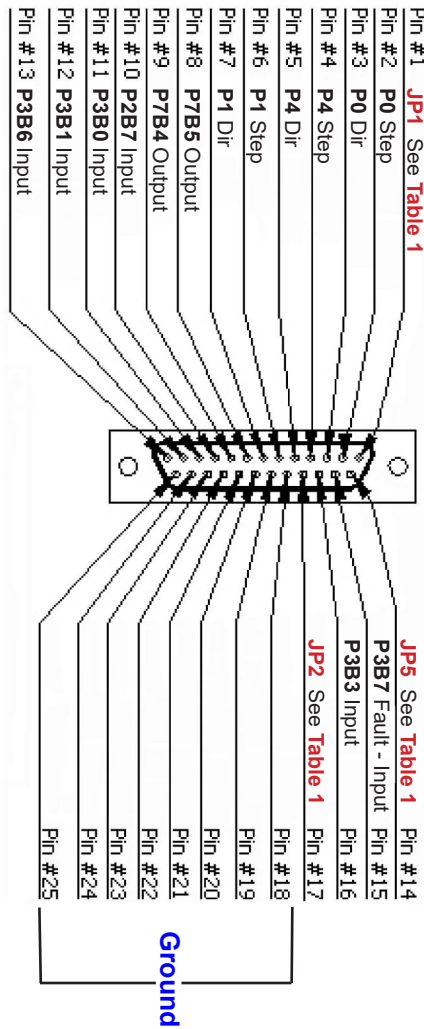


Table 1		Jumper Position			
Header	Jumper	1&2	Description	2&3	Description
C2	JP1	P7B7	OUTPUT	P2B6	INPUT
C2	JP2	P7B6	OUTPUT	P3B4	INPUT
C2	JP5	P3B2	INPUT	NA	PWM

## 6 Axis Dual DB25 C3 - 4 Axis Connector:

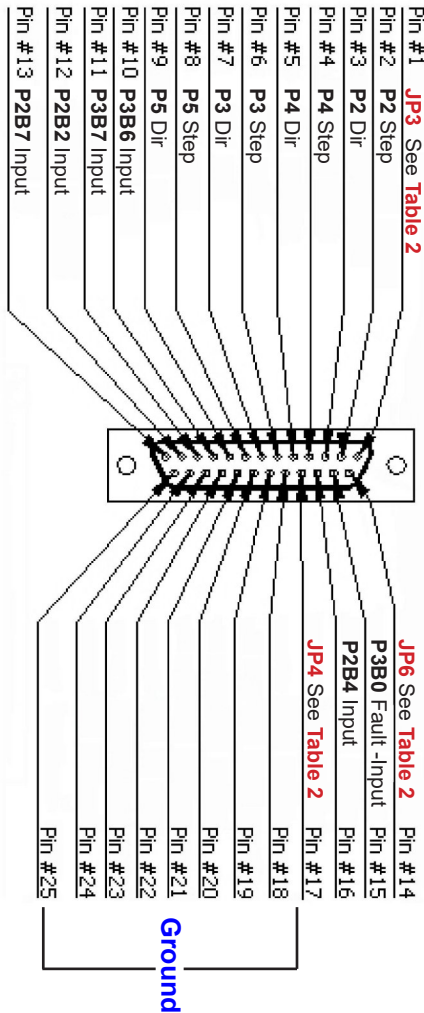


Table 2		Jumper Position			
Header	Jumper	1&2	Description	2&3	Description
C3	JP3	P7B7	OUTPUT	P3B5	INPUT
C3	JP4	P7B6	OUTPUT	P2B5	INPUT
C3	JP6	P2B3	INPUT	NA	PWM

PCI 24H:

24/CTR3	24 & 24H	
CLK 0	GND	19
GATE 0	+5V	18
OUT 0	GND	17
CLK 1	+12V	16
GATE 1	GND	15
OUT 1	-12V	14
CLK 2	GND	13
GATE 2	-5V	12
OUT 2	GND	11
	PORT B 0	10
	PORT B 1	9
	PORT B 2	8
	PORT B 3	7
	PORT B 4	6
	PORT B 5	5
	PORT B 6	4
	PORT B 7	3
	IR ENABLE	2
	IR INPUT	1



37	PORT A 0
36	PORT A 1
35	PORT A 2
34	PORT A 3
33	PORT A 4
32	PORT A 5
31	PORT A 6
30	PORT A 7
29	PORT C 0
28	PORT C 1
27	PORT C 2
26	PORT C 3
25	PORT C 4
24	PORT C 5
23	PORT C 6
22	PORT C 7
21	GND
20	+5V



# **WinCNC Controller**

---

This page is intentionally left blank.



# Index

## Illustrations

PCI 7200 37 Pin Connector  
CN1: 165

6-Axis 37-Pin Daughter Board  
Connector CN2: 166

4-Axis 25-Pin Daughter Board  
Connector CN2: 167

6 Axis Dual DB25 C2 - 3 Axis  
Connector: 168

6 Axis Dual DB25 C3 - 4 Axis  
Connector: 169

PCI 24H: 170

PCI 48H: 171

## A

accel= 63  
ad= 63  
altaxishi= 63  
altaxislo= 63  
altaxispec= 63  
anykey= 64  
Appendix 165  
arc\_err= 64  
arc\_min= 64  
arcs= 64  
arctype= 65  
atc1= 65  
atc2= 65  
autoarcfeed= 65  
autoaux= 66  
Auto Digitizing 42

autodir= 66  
autofdir= 66  
autogas= 66  
automode= 67  
auxin= 67  
AuxIn Indicators 100  
auxout= 68  
AuxOut Buttons 101  
AuxOutEx Buttons 101  
auxoutvel= 68  
axisadj= 62  
axischar= 60  
Axis Controls 107  
axishi= 63  
axislo= 63  
axismap= 62  
Axis Settings 60  
axispec= 61  
axisvel= 62  
Axis Window 21

## B

backlash= 69  
bitmapdir= 69  
bmpeditor= 69  
bmpend= 69  
bmpoffset= 70  
bmpstart= 69  
btnoptions= 71

## C

cmdabort= 71  
cmdend= 71  
cmderror= 71  
cmdrestart= 71  
cmdrestartfeed= 71  
cmdrestartrapid= 71  
cmdstart= 71  
CNCSCRN.INI Settings 97  
comchars= 72  
comdll= 72

Command Reference 111  
commatodec= 72  
Configuration Menu 17  
contdelay= 72  
Contents 2  
Coordinate System 28  
ctrlf#= 72  
Custom Screen Buttons 23  
cutcomp= 72

**D**

da= 73  
define= 148  
Display Sections 97  
drillbank= 73  
Driver Installation for PCI-7200  
Card: 8

**E**

editor= 73  
enab= 74  
errorabort= 74  
esc\_step= 74  
exaltaxishi= 75  
exaltaxislo= 75

**F**

Feed Rate Override 27  
File Menu 15  
filetype= 75  
forcelim= 75  
Frames 100  
Frame Scan Digitizing 38  
freq= 76  
fslock= 76

**G**

g2modal= 77  
g08= 76  
g09= 77

g28altmove= 77  
g28check= 77  
g28checktol= 77  
g28coord= 77  
g28\_lim\_cnt= 77  
g28move= 77  
g28req= 78  
g37= 78  
g54= 79  
g55= 79  
g56= 79  
g57= 79  
g58= 79  
g59= 79  
g81dwell= 79  
g81set= 79  
g84set= 79  
G90 Mode 29  
G91 Mode 28  
G Codes 115–122

**H**

helpfile= 80  
Help Menu 19  
hibound= 80  
hilim= 80  
holddir= 80  
homedir= 80  
Homing the Machine 27

**I**

Index 173, 174, 175, 176  
Installation 3  
Internally Defined Variables  
112–113

**J**

Job Command Buttons 102  
Job Command Ex Buttons 102  
Jog Increment Buttons 22

Jog Menu 18  
Jog Mode Buttons 99

**K**

keypad= 80  
Keypad 155  
Keypad Configuration 155  
keypaddir= 81  
Key Pad Out Buttons 105  
Keypad Type 1: without LCD 156  
Keypad Type 2: with LCD 157  
Keypad Type 3: with LCD 158  
Keypad Type 4: XHC HB04  
Wireless with LCD 161  
Keypad Type 4: XHC PHB04B  
Wireless with LCD 163  
knifeangle= 81  
knifemode= 82

**L**

L Codes 123–141  
laser= 82  
laserscan= 83  
Laser Scanning/Digitizing 48  
lim\_buf= 83  
lim\_cnt= 83  
Limit Switches 26  
lim\_mode= 83  
lim\_step= 84  
lobound= 84  
logdir= 84  
logfile= 84  
lolim= 84

**M**

Macro Configuration 148  
Macro File Configuration 147  
Manual Digitize 36  
mapping= 84  
maxparam= 85

maxstepv= 85  
maxtrans= 85  
M Codes 141–145  
Menu Bar 14  
Menu Buttons 106  
Message Display Window 21  
Messages and Error Codes 150

**N**

nmode= 85  
numkp= 85

**O**

Options Menu 18  
Outline Digitizing 45  
override= 85

**P**

PCI24H= 86  
PCI24HREG= 86  
PCI48H= 86  
PCI48HREG= 86  
Plasma Torch Height Control 54  
portin= 86  
portout= 86  
positions= 86, 87  
Program Operation 13  
pumpcont= 87  
Pump Control 34  
pwm= 87

**R**

rapidjog= 88  
rapid\_lock= 87  
Rapid Mode 107  
Repeat Command 33  
Repeat Command Buttons 104  
  
Repeat Command Ex Buttons  
104

Requirements 3  
restart= 88

**S**

sawmode= 88  
scan= 88  
Screen Configuration 97  
Screen Display 14  
Settings Menu 17  
Shell Command Buttons 102  
Shortcut Keys 24  
signal= 89  
Simulate 33  
skipchars= 89  
Slider Controls 98  
softlim= 90  
Soft Limit / Boundaries 32  
spinchar= 90  
spindle= 90  
State Buttons 108  
Status Bar 23  
steppulse= 91  
surfacemap= 91  
System Configuration 59

**T**

table= 91  
thc= 92  
thcauto= 92  
thcdisab= 93  
thcdiv= 92  
thclim= 93  
thcvel= 93  
timertype= 59  
Tool Length Offset 32  
tranadj= 93  
Transit Menu 18  
Transit Speed Buttons 22

**U**

unit= 59  
User Buttons 103

**V**

vgroup= 93  
Viewer 33  
View Menu 17  
virtin= 94  
Virtual AuxIn Indicators 101  
Virtual Input Button 109  
vmatch= 94

**W**

watch= 94  
WINCNC.INI Settings 59  
Workspace 31

**X**

xhc= 95