Disclaimer

1. This whole document with all it's sections is to be used as guidance only.
2. All comments, suggestions, operations, and instructions, stated herein are subject to change without notice.
3. Customers should make their own risk assessments and take appropriate measures to comply with health and safety regulations prevailing in their country.
4. We or our agents take no responsibility for loss, damage or injury caused:
   - by the use of this machine,
   - by inappropriate safety measures adopted by the user,
   - as a result of work carried out by unqualified person(s),
   - by issues generally considered to be Force Majeure (e.g. acts of God).

E. & O.E.

Contents

Part A: Quick Start Guide

Part B: Mach3 Mill Guide

Part C: Technical Guide

Congratulations on purchasing a Sieg KX Series CNC Mill!

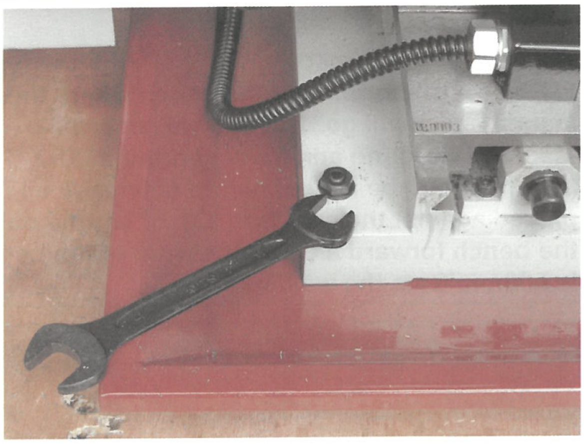
This quick start guide is designed to get you cutting as quickly as possible so please take some time at this stage to read through it before going any further.

We have tried to make the guide as "Beginner Friendly" as possible and even if you are already a Mach3 expert, we highly recommend you read through it at least once.

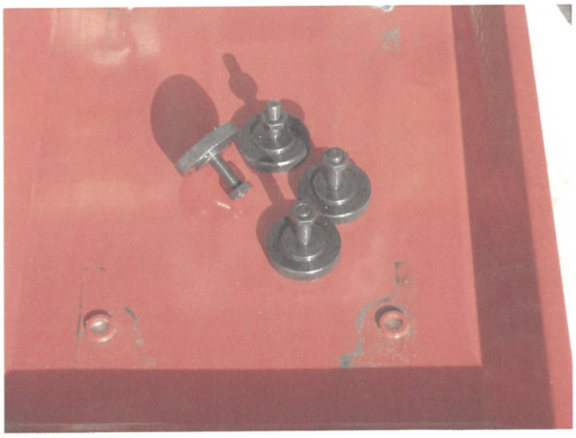
First, we'll deal briefly with the machine itself. Next we will move on to the Mach3 software installation on your computer. Finally, we will work through setting the machine up ready to make your first cut.

1. Unpacking and Setting Up Your Sieg CNC Milling Machine

1. The KX Series mill is bolted onto the base of the crate. There are four bolts, two at the front and two at the back. You can see the two front bolts either side of the main machine base in the photo above.



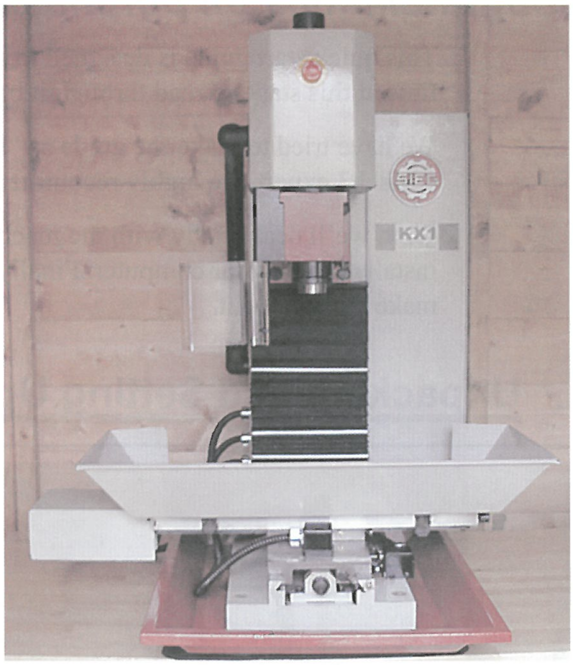
2. A close up of one of the fixing bolts.



3. There are four adjustable feet supplied

4. Fit the feet to an approximately equal height. The feet will be adjusted and finally tightened when the machine is on the bench. Tighten the nuts sufficiently to stop the tray falling off. The tray should fit into the holes in the bottom of the machine. When the machine has been lifted onto the bench, level the machine by using the adjustable feet and finally tighten the foot nuts against the bottom of the tray.

5. Place a rubber mat on the bench where the machine feet will go. The mat will stop the machine vibrating and moving around. The bench should be a minimum of 600mm deep if placed against the wall. You can of course pull the bench forward a bit so the front of the bench is 600mm from the wall.



6. Lift the machine on to the bench (the KX1 shown here is a two person lift) placing on the rubber mat.

7. Remove the four screws holding the guard so you can get at the machine to clean all the preservative off. Use WD40 or a similar product to remove the preservative. Then use SAE 30 hydraulic oil or equivalent to protect the surface. Safely dispose of any wipers used. Replace the guard after cleaning

8. Use protective clothing when cleaning the preservative of the machine.

Although these pictures show the smaller KX1 CNC Mill, the same principles apply to the larger KX3 CNC Mill - except you will probably require a hoist to lift the heavier machine!

2. Computer Set-up

Before you can start milling, you need a ***dedicated*** PC running Mach3 CNC control software. A laptop will not work correctly. The PC should have a ***surge protector*** fitted.

There should be:

- No web browser
- No email client
- No enabled network connections
- No antivirus software

running in the background. Mach3 will only run on **Windows 2000** or **Windows XP.**

2.1 Computer Specification

The specifications of this computer are as follows:-

Minimum

- 1.2GHz CPU
- 512MB Memory
- 20 GB Hard Drive
- Windows 2000 or XP, Home or Pro version
- (Mach 3 will not work correctly on Windows Vista)
- LPT Parallel port (Printer). This is essential for Mach3 to work.
- Free USB Port

Recommended

- 2GHz CPU (Dual Core CPU's will run with Mach)
- 1GB Memory
- 20 GB Hard Drive
- Windows 2000 or XP, Home or Pro version
- (Mach 3 will not work correctly on Windows Vista)
- 32 MB AGP/IDE-E Video Card
- LPT Parallel port (Printer). This is essential for Mach3 to work.
- Free USB Port
- Front USB port (for memory stick drives for moving files)

2.2 Register the Machine

Now register the machine at the Small CNC Support website - (see page 21 for full details):

http://www.smallcncsupport.com/

This is the official support site for all Sieg KX1 and KX3 CNC machines running Mach3 sold worldwide and is your first point of contact for support issues related to your machine. To register, you will need the Serial Number and Registration Number that came with your machine.

Once registered, we strongly recommended that you check the Updates page for any later versions of Mach3 files as well as updates for these guides.

3. Conventions used in this guide

To help make the software section of this Quick Start Guide easier to understand, keyboard input will be shown in a different way to screen input using a mouse or touch screen.

3.1 Keyboard Symbols:

[↵] Enter or Return key

[Tab] Tab key

[←] [→] [↑] [↓] Cursor keys

[Page Up] [Page Down] Page Up and Page Down keys

[⇧] Shift key [Ctrl] Control Key [Alt] Alt key

[Esc] Escape key [F5] F5 Function key [S] S character key

A passage of text might include: [⇧]+[←]; this means Shift+Left arrow or hold down the Shift key while pressing Left Arrow (cursor) key on the keyboard.

3.2 Mach3 Commands

3.2.1 Main Menu bar commands

You may be directed to <Config/Ports and Pins>. This indicates that you should open the Config menu at the top of the screen and click on the Ports and Pins option (which opens a dialogue box with various options)

3.2.2 Screens

Below the main menu bar, there is a row of 6 buttons which open different screens in Mach3. These will be indicated thus: **Program Run Alt-1** (Alt 1 indicates this screen can also be accessed using the [Alt]+1 keys on the keyboard)

3.2.3 Buttons

Most of the Mach3 screen buttons have appropriate text on them so when you see ***Zero X***, or ***Set Tool Offset***, click the screen button with that name.

3.2.4 Data Entry Fields

Many of the data display buttons in Mach3 will also allow you to enter data from the keyboard. For example, if you are asked to enter +30.0000 into the ***Z-Axis DRO***, click on the Z-Axis Digital Read Out field on the screen, type in +30.0000 and press [↵] to accept the data. Notice how the background colour of the button changes when you click on it, letting you know it's ready for you to type some data in and changes back again when you press [↵].

4. Install Mach3 now

> Installing the Mach3 software from the CD or the Small CNC Support website will include:
> Configuration files tailored specifically to Sieg KX Series machines.
> Customised versions of the Mach3 screens for your machine.

You do not need the CNC mill connected yet. If you are just starting it would be better not to have it connected. You can install Mach3 from the CD or from a later version you downloaded from the Small CNC Support website.

4.1 Run Setup.exe

Run the **Mach3_Setup.exe** file now and you will be guided through the usual installation steps for a Windows program such as accepting the license conditions and selecting the folder for Mach3. On the Setup Finished dialog you should ensure that **Load Mach3 Driver** and **Install English Wizards** and also **Initialise System** are checked in the dialog, and click Finish. You will now be asked to reboot before running any Mach3 software.

4.2 The vital re-boot

This reboot is vital. If you do not do it then you will get into great difficulties which can only be overcome by using the Windows Control Panel to uninstall the driver manually. So please reboot now.

4.3 Which desktop icon??

So you have rebooted! The installation wizard will have created several desktop icons for the main programs. **Mach3 Loader** is the **only** one we are interested in since it will ask which machine Profile you wish to use. KX1 Mill and KX3 Mill are shortcuts which run a defined Profile specific for those machines.

4.4 Now test the system

It is now highly recommended to test the system. Mach3 is not a simple program. It takes great liberties with Windows in order to perform its job; this means it will not work on all systems due to many factors. For example, the QuickTime system monitor (qtask.exe) running in the background can kill it and there will be other programs which you probably are not even aware are on your system that can do the same. Windows can and does start many processes in the background; some appear as icons in the system tray (bottom right of screen) and others do not show themselves in any way.

Other possible sources of erratic operation are local area network connections and it is advised to disable these. It cannot be stressed enough that you use a dedicated machine to control this Sieg product.

It is now worthwhile to run the DriverTest.exe file. Use Windows Explorer (right-click Start) and by right-clicking on the DriverTest.exe file you can drag this shortcut onto your desktop. You can find the DriverTest.exe file in the Mach3 directory. Double click the Driver Test icon that you set up. Its screen shot is in Fig 1.

You can ignore all the boxes with the exception of the Pulse Frequency. It should be fairly steady around 25,000 Hz but yours may vary, even quite wildly. This is because Mach3 uses the Windows clock to calibrate its pulse timer and, over a short time scale, the Windows clock can be affected by other processes running on the computer. So you may actually be using an "unreliable" clock (the Windows one) to check Mach3 and so get the false impression that Mach3's timer is unsteady.

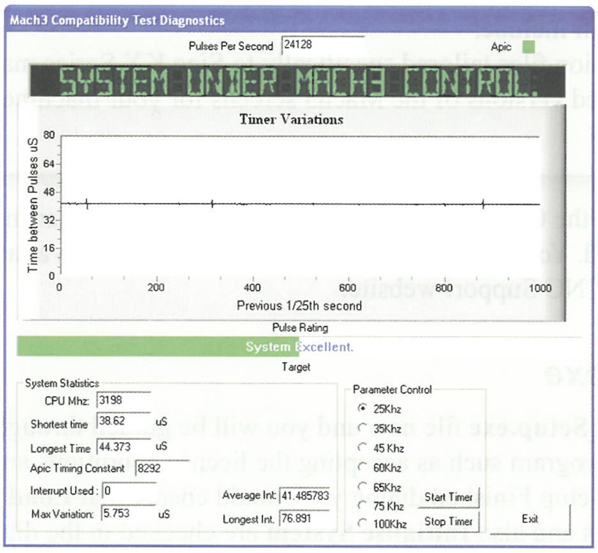


Fig. 1

Basically, if you see a similar screen to Fig. 1 with only small spikes on the Timer Variations graph and a steady Pulse Frequency, everything is working well so close the Driver Test program and skip to the section below.

You may have one of two things happen to you when running the test which may indicate a problem.

1. "Driver not found or installed, contact Art.", this means that the driver is not loaded into Windows for some reason. This can occur on XP systems which have a corruption of their driver database. Reloading Windows is the cure in this case. Or, you may be running Win2000. Win2000 has a bug/feature which may interfere with loading the driver. It may need to be loaded manually - see section 3.2.5 in the Mach3 User Guide.
2. When the system says, taking over…3…2…1.. and then reboots, one of two things has occurred. Either you didn't reboot when asked (told you!!) or the driver is corrupted or unable to be used in your system. In this case refer to the reference manual and remove the driver manually, then re-install. If the same thing happens, please notify ArtSoft using the e-mail link on www.artofcnc.ca and you will be given guidance.

A few systems have motherboards which have hardware for the APIC timer but whose BIOS code does not use it. This will confuse Mach3 install. A batch file SpecialDriver.bat is available in the Mach3 installation folder. Find it with Windows Explorer and double-click it to run it. This will make the Mach3 driver use the older i8529 interrupt controller. You will need to repeat this process whenever you download an upgraded version of Mach3 as installing the new version will replace the special driver. The file OriginalDriver.bat reverses this change. This is a rare occurrence these days with the new motherboards but it may be necessary to use this driver.

4.5 Registering the Mach3 software

The Mach3 software supplied on the CD is a demo version. It is a fully functioning program except that it's limited to 500 lines of G-code and some of the wizards are disabled. It is recommended that you purchase a Mach3 software licence via the link on:

http://www.smallcncsupport.com/

5. Connecting and Starting the KX Series CNC Mill

5.1 Connecting the Computer

Now that you are happy the driver is running correctly, it's time to connect the computer lead (supplied with your KX CNC mill) to the 25 way D connector on the back of the KX CNC Mill, see Fig. 2.

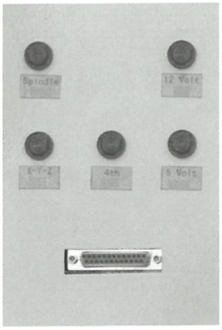


Fig. 2

Connect the other end of the lead to the printer connector on your PC. Do not connect through a printer, a scanner or other peripheral. Mach3 requires a dedicated parallel port.

5.2 Switching on

1. Switch the computer on first and allow to boot up.
2. Start Mach3: Open the **Mach3 Loader** program on the desktop and select either the KX1 or KX3 mill profile (to suit your machine), and click the OK button, Fig. 3. If you installed Mach3 from the CD or the smallcncsupport website, these profiles will be present by default.

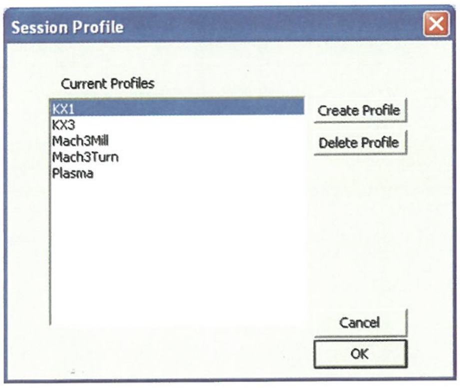


Fig. 3

Switch the KX mill on. This will now load the correct custom screen for your machine.

5.3 Reset the machine

Press the red ***Reset*** button, Fig. 4. The message "Press Reset .... Emergency Mode Active" should clear.

If not, check that the emergency stop button has been released and chuck guard is closed and try again.

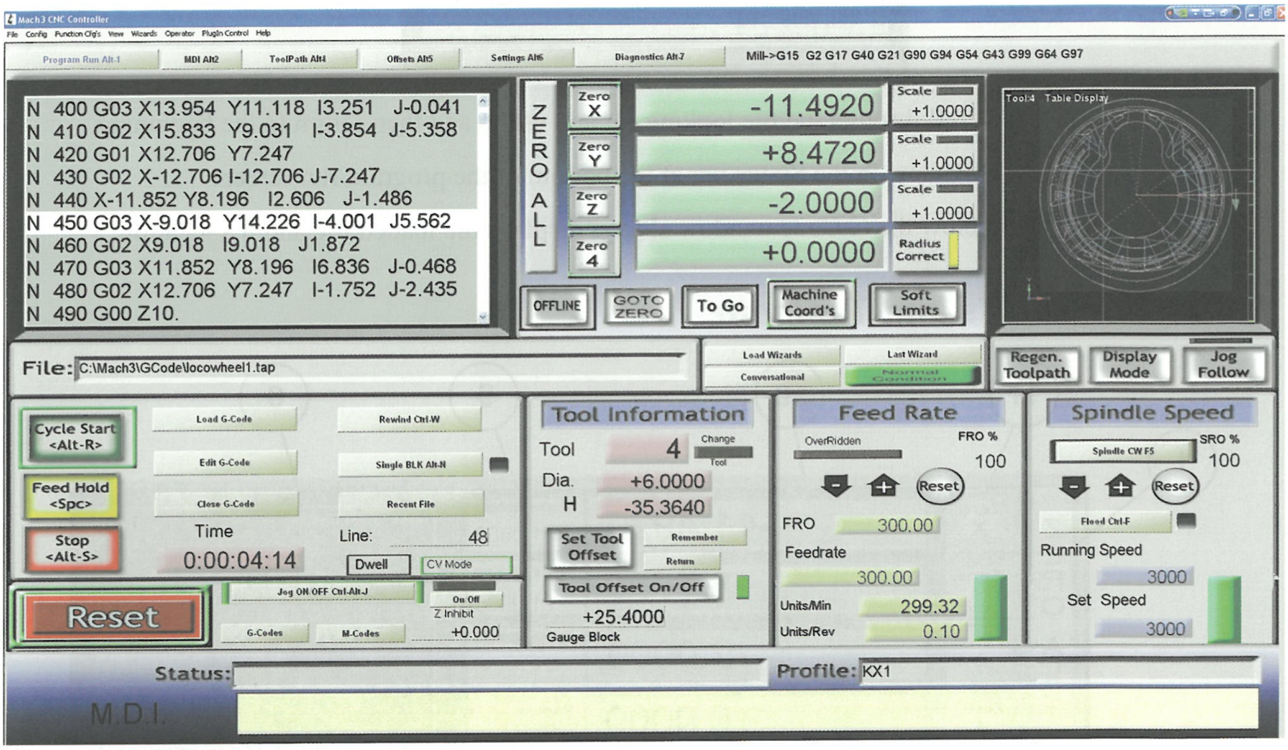
Fig. 4

6. Mach3 Screen Functions

Below is a screen shot of the main Mach3 front screen (shown with a file running) - known as the **Program Run Alt-1** screen. This is the correct custom screen you should now see (subject to updates).

Most of the information needed to setup and run appears on this screen. It may look complex but as it is designed in modules, the commands are actually grouped into separate function blocks known as Control Families.

Each control family will be taken in turn and explained in detail.



There are seven main families and two minor ones.

Main ones are:

Reading from left to right, top row:

- The G code window
- The axis Digital Readouts [DRO's]
- The toolpath screen.

Bottom row:

- The file loading and run commands
- All the tool information and settings
- Feed rates
- Spindle controls.

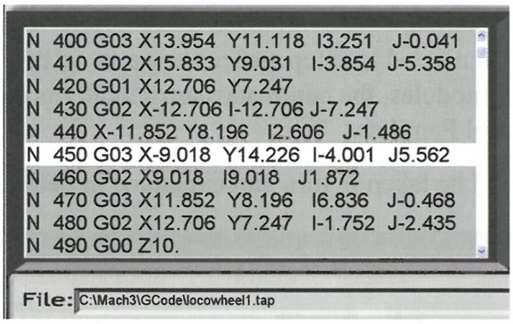
Minor families:

- All screen and wizard command buttons
- The MDI commands.

These families will be covered by separate larger screen shots and full explanations of each feature.

6.1 The G Code Window

This is the easiest family to understand - when a file is loaded it's contents appear in this window.

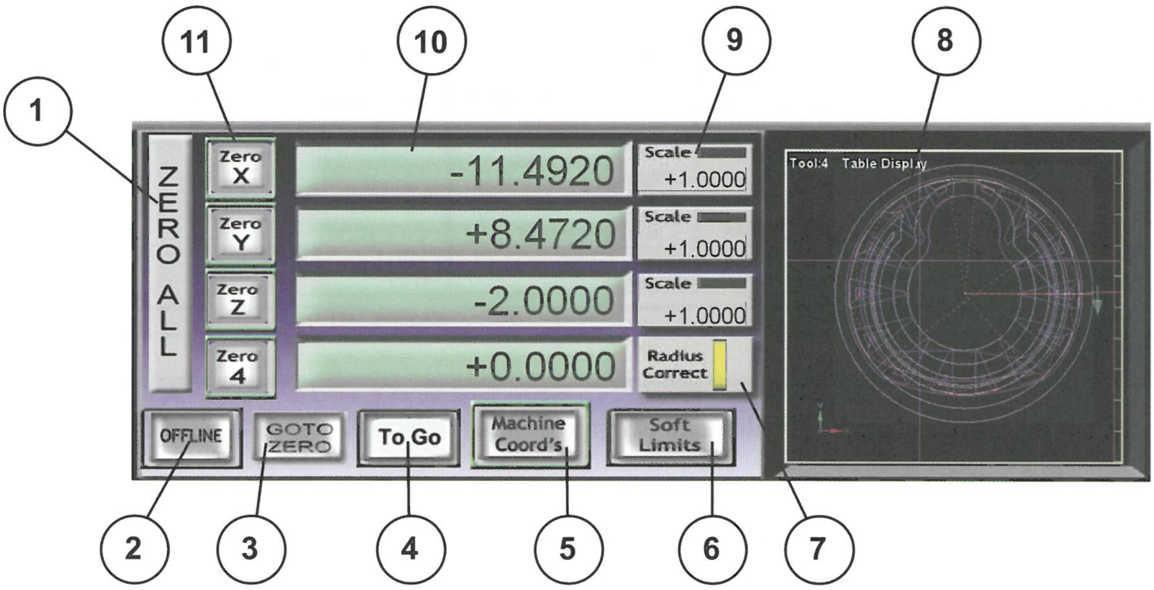


Mach3 will scroll through the file looking for any errors and then return to the beginning.

Any errors will show in the Status bar at the bottom of the program run screen.

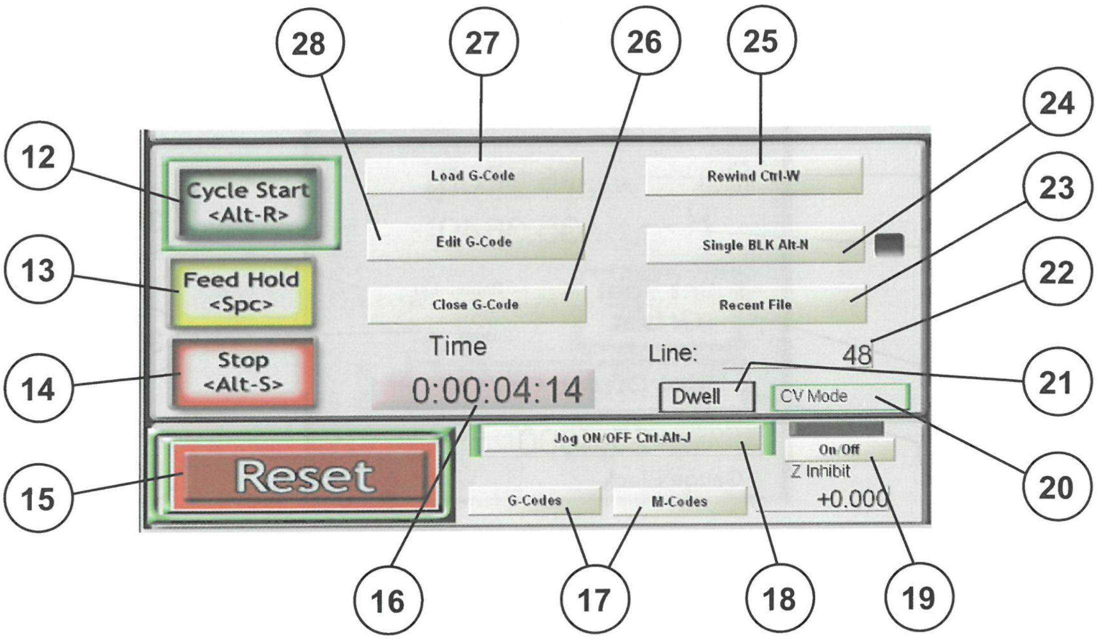
The file bar below the G Code window always displays the loaded file and it's location

6.2 The DRO's and Toolpath



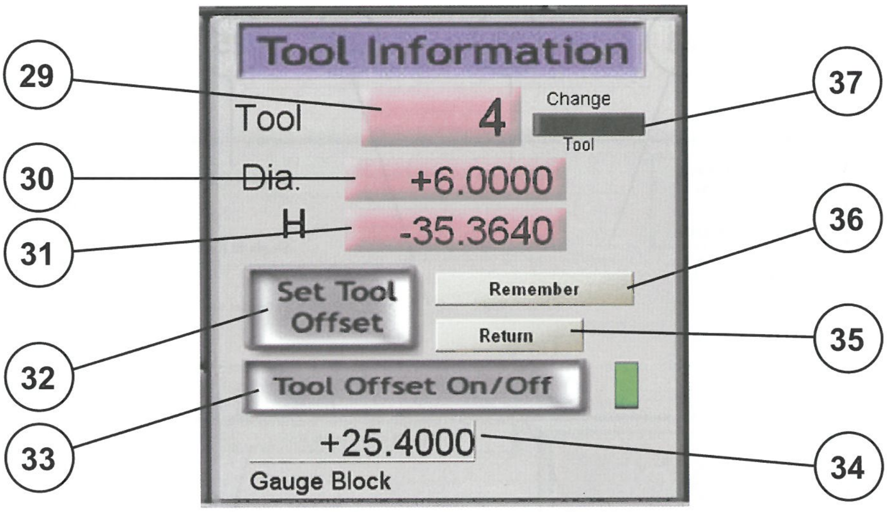
1	Zero All Axis	Zero's all the axis together when pressed.
2	Offline	Disables the parallel port to allow a file to run without the machine running.
3	Goto Zero	Returns the machine to its Zero Origin position.
4	To Go	Full title is - Distance to go - and is an inverse of what the DRO's state.
5	Machine Coord's	Switches between Machine Coords and Work Coords.
6	Soft Limits	Soft limits are a software version of limit switches and can be set under <Config/Homing/Limits>.
7	Radius Correction	For setting the radius of the work on the 4th axis so that the feed rate is correct. The actual Setting Box is on the **Settings Alt6** page.
8	Tool Path window	Complex, so see description of the Screen and Wizard Command Module.
9	Scale Window	Normally set at 1.00 but any value can be entered. If 0.5 is typed in to X and Y then the resulting file will be half size but full depth in the Z.
10	DRO's	Shows the values of each axis. You can also manually enter values into these.
11	Zero Buttons	One for each axis to zero the display.

6.3 Run and File Functions



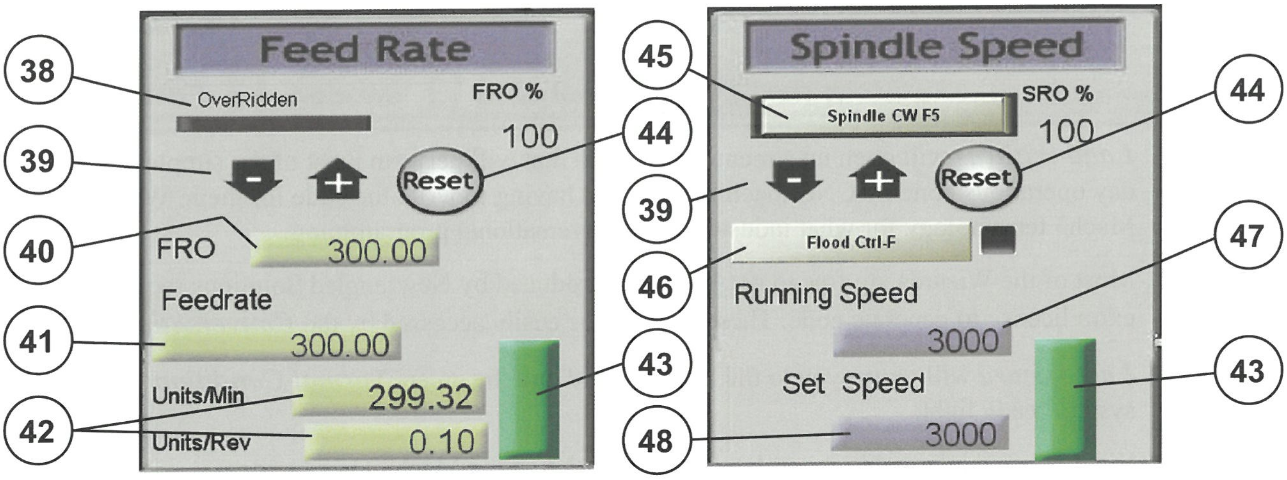
12	Cycle Start	Start button to run program, also obtained from Alt + R on the keyboard.
13	Feed Hold	Enables the program to be paused until Cycle Start is pressed again, shortcut key is the spacebar.
14	Stop Button	Stops the program, shortcut key is Alt + S.
15	Reset	Used to reset the machine at startup or from an emergency stop. Shortcut key is Esc.
16	Elapsed Time	Timer for program run time.
17	G & M Codes	Opens a new screen to list accepted G and M codes understood by Mach3.
18	Jog Control	When lit, allows the arrow keys ←↑↓→ and page up / down Page Up Page Down to be used as jog keys.
19	Z Inhibit	Type a positive number in the DRO box and switch On, then when the code is run there will be no Z moves lower than the number input.
20	CV Mode	When lit, by default, Mach3 looks-ahead in the code to smooth out the toolpath.
21	Dwell	Lights up when a dwell time [G04] is called by the program.
22	Line	Displays the current line number as it runs through the code.
23	Recent File	Brings up a list of the most recent files used.
24	Single BLK	Single block or line, when active it runs through the code one line at a time using Cycle Start.
25	Rewind	Rewinds the code back to the start point.
26	Close File	Closes the current file. Opening a new file will do the same and over write, not Append.
27	Load G Code	Opens a new file.
28	Edit G Code	Opens the current file up in Notepad for editing, also updates the file on saving and closing.

6.4 Tool Information



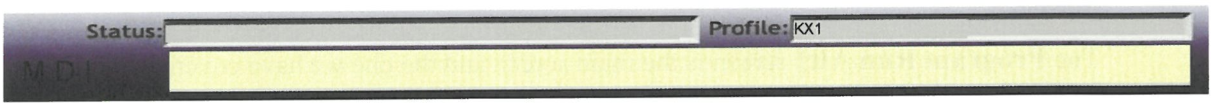
29	Tool Number	Lists the tool number when a program is running. It is also used to enter a tool number when setting the tool height offset as described later in this Quick Start guide.
30	Tool Diameter	Lists the diameter either entered directly or from the tool table.
31	Tool Height	Lists the tool height offset from the tool table.
32	Set Tool Offset	Press to enter the tool offset into the tool table when the tool is at the correct position, again described later in this Quick Start guide.
33	Tool Offset	Toggle on / off to switch on the tool offsets in the Z DRO.
34	Gauge Block	Enter the height of a gauge block or setting block if used. This will automatically added to the tool offset value.
35	Remember	Click on this function before moving off position for any reason like changing a broken tool and it will remember the position.
36	Return	Brings up a menu to return to the default position or you can type in an alternative.
37	Change Tool	An LED that lights up when it's expecting a tool change. Once changed, press ***Cycle Start*** to continue.

6.5 Feed Rate & Spindle Speed



38	OverRidden	Flashes when the Feed Rate is greater than or less than 100%.
39	+ & - Signs	Allows incremental override steps.
40	FRO	Feed Rate OverRide, shows the true feed rate.
41	Feedrate	Shows the programmed feed rate.
42	Units	Two DRO's to show Units /Min and Units / Rev.
43	Slider	Feed Rate Over Ride slider control. Drag with mouse or touch screen to alter feed rate.
44	Reset	Returns over ride rate back to 100%.
45	Spindle	Switches the spindle on / off either by clicking or using the F5 hot key.
46	Flood	Switches coolant on (if fitted)
47	Running Speed	Displays the true spindle speed.
48	Set Speed	Shows the programmed spindle speed and also a speed can be entered here.

6.6 MDI Function



The Status bar shows any errors present in Mach3 . The Profile shows what machine it is setup for.

The MDI box which stands for Manual Data Input is a simple way to make manual moves one line at a time.

Using this function it is possible to enter one line of code to make the machine perform a move or Function. You must have knowledge of G-code to use this or damage to the machine can occur.

Use of this function is described in section 3.5 in the Mach3 User Guide.

6.7 Wizards and Screen Display

Load Wizards will open up a menu of Wizards that will perform most of the simple day to day operations done by CNC machines without having to write the code for them. Wizards are Mach3 terminology for what industry term conversational programming.

Most of the Wizards are free to use but some produced by Newfangled Solutions require an extra licence to generate code. These can also be easily accessed by the ***Conversational*** button.

Last Wizard will return you to the last one used and the green ***Normal Condition*** is just a system ready light.

Wizards are explained in more detail in section 3.6 in the Mach3 User Guide.

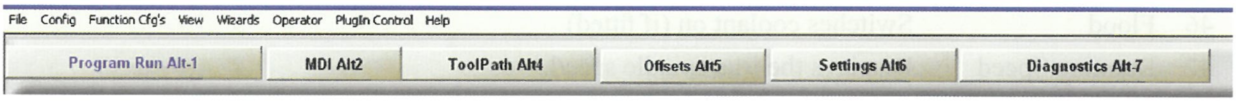
Regen. Toolpath is used when the code has been updated to force a screen redraw.

Display Mode toggles between the max size of the travels and the size of the job.

Jog Follow moves the screen to a central tool instead of the tool moving on the work, more suitable to fixed head machines.

Clicking on the toolpath screen whilst the job isn't running and holding the left mouse button down enables the screen to be rotated in 3D. The mouse wheel allows zoom in and zoom out.

6.8 Main Menu and Mach3 Screens Buttons



Across the top of the screen is the normal Windows type file layout. All these are covered in the Mach3 User Guide. The lower Screen buttons are the different screens that can be accessed in Mach3.

The **Program Run Alt1** screen is the main screen and the one we have covered here.

MDI Alt2 screen contains a duplicate MDI screen and a teach facility [see section 3.5.2]

Toolpath Alt4 screen contains a larger toolpath window for ease of use and also a set of DRO's labelled Program Limits that tell you after a file has been loaded how big the travels are in all 6 directions. It's advised you check these to make sure the job will fit the machine travels.

Offsets Alt5 screen contains all the functions to setup your work as regards the Origin.

The **Settings Alt6** page contains over ride limits so you can jog off if you hit a limit switch and the radius correction DRO for the 4th axis. The rest is more for fault finding.

The **Diagnostics Alt7** page is purely for fault finding if anything goes wrong and there will be a simple check guide on the smallcncsupport website for registered users to assist in checking machines.

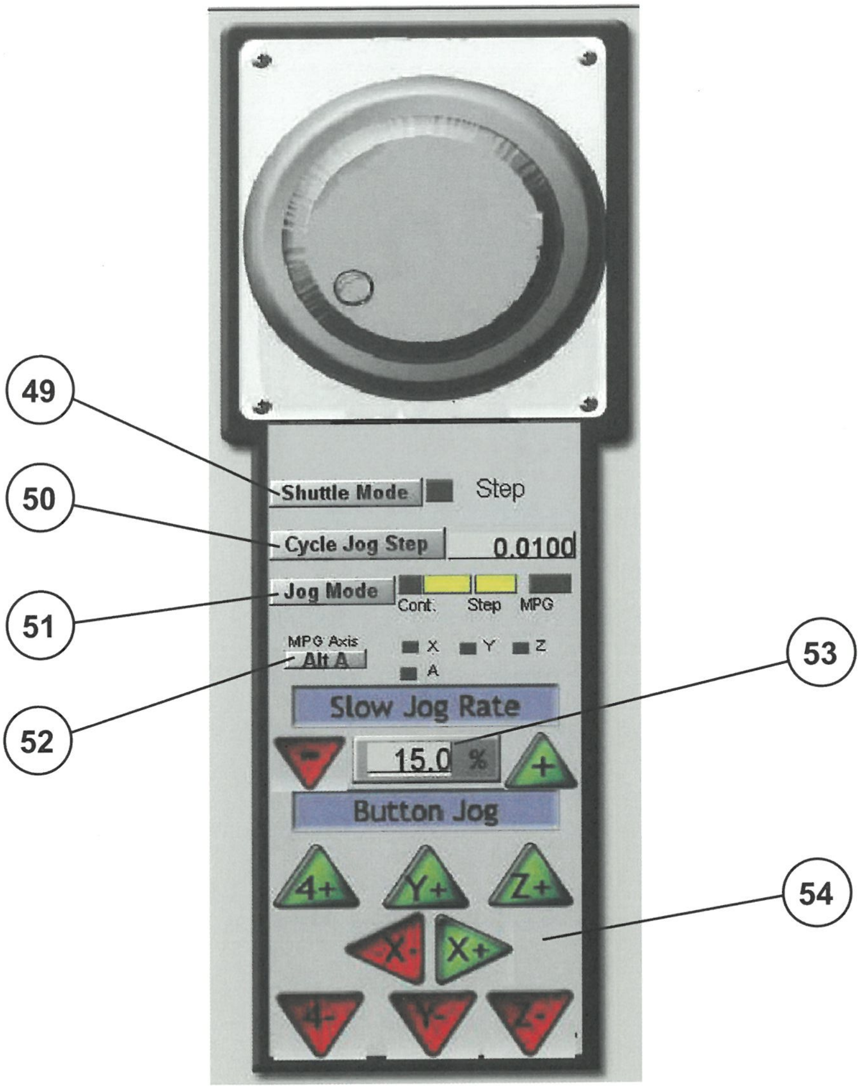
This is only a simple quick start guide and it is strongly advised you read the main Mach3 User Guide where all of what has been described here is cover in far greater depth.

6.9 Jog screen

The jog screen is available when the tab key is pressed and shows to the right of the main screen. This screen sets the feed rate for jogging as a percent of the maximum speed as set in <Config/Motor Tuning>.

In normal use, the machine, when jogged, moves at this slow jog rate.

If any jog key is pressed whilst the shift key is held then it will move at maximum speed.



49	Shuttle Mode	May be used if a Contour Design USB Shuttle is used as a handwheel.
50	Cycle Jog Step	Sets the step increments in units of 1.0000, 0.1000, 0.0100, 0.0010, 0.0001 and then back to 1.0000
51	Jog Mode	Toggles between Continuous and Step. Cont. moves the selected axis continuously whilst a jog key is pressed and Step allows jogging moves to be in steps of whatever units are selected in 50.
52	MPG Axis	Provides a visual check on what axis is selected when 49 is in operation.
53	Slow Jog Rate	Displays / sets slow jog speed described above. This can be altered with the arrow keys to either side or it can be typed in.
54	Jog Buttons	Click to jog an axis. The 4+ and 4- arrow buttons are for jogging the 4 axis.

7. Coordinates

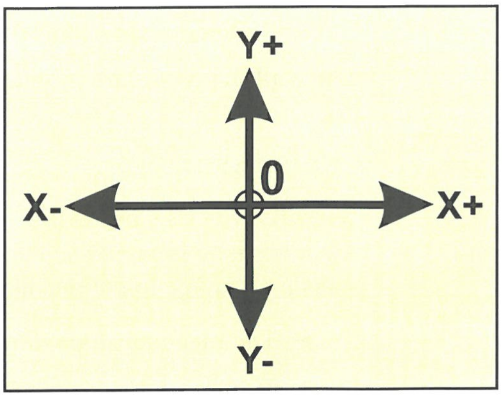


Fig. 5

Machining coordinates are normally entered into a program as X Y Z e.g. X1.0 Y-10.0 Z25.0, but what does this mean?

Imagine you are looking down on the top of the workpiece in a vice on the table of the machine - see Fig. 5. The point where the two lines cross is known as the origin, or X0.0 Y0.0.

If the tool moves to the left of the origin, the X coordinate will be negative. If the tool moves to the right of the origin, X will be positive.

If the tool moves from the origin towards the back of the machine (Up in the diagram) then Y will be positive and conversely, Y will be negative of it moves towards the front.

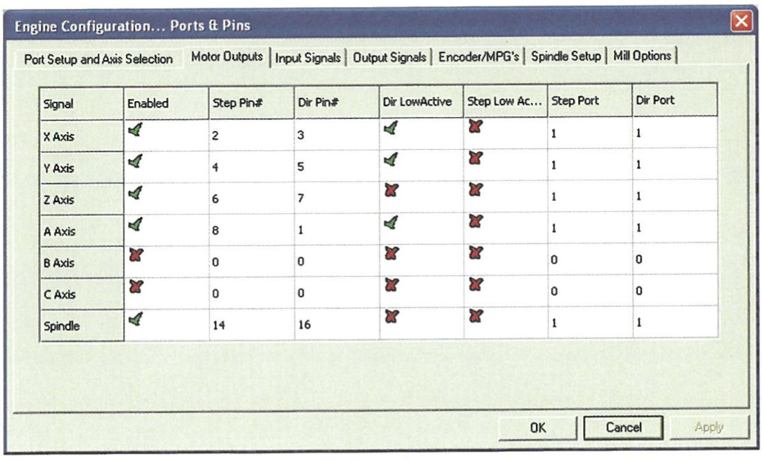
It will be helpful later on if you begin by thinking of the ***tool moving*** in the X and Y directions even though it's the table that will be moving and not the tool!

The Z zero position is usually on the top surface of the component to be milled. Any movement above the job is in the Z+ direction and any movement below the surface of the job is in the Z- direction.

7.1 Check the Axis Direction

With a new machine, it's quite possible the axes of the machine will be moving in the wrong direction since the motor direction is difficult to determine at the time the machine is built. If the direction of any axis is reversed, it's a simple job to change. Proceed as follows to check and, if necessary, reverse the axis directions:

- Press the Tab key on the keyboard and the **Jog** Flyout will appear on the right of the screen.
- Use the jog buttons ***X-***, ***X+***, ***Y-***, ***Y+***, ***Z-*** and ***Z+*** at the bottom of the flyout to move the table and head. The table should move as follows: ***X-*** = right, ***X+*** = left, ***Y-*** = in (towards the column) and ***Y+*** = out. The head should move: ***Z-*** = down and ***Z+*** = up. If all axes are moving the in right directions, there is nothing to change and you can skip to the next section. If one or more axis is moving the wrong way, continue to the next step.
- On the Mach3 menu select <Config/Ports and Pins> and select the ***Motor Outputs*** tab at the top of the window - see Fig 6.



Signal	Enabled	Step Pin#	Dir Pin#	Dir LowActive	Step Low Ac...	Step Port	Dir Port
X Axis	✓	2	3	✓	✗	1	1
Y Axis	✓	4	5	✓	✗	1	1
Z Axis	✓	6	7	✗	✗	1	1
A Axis	✓	8	1	✓	✗	1	1
B Axis	✗	0	0	✗	✗	0	0
C Axis	✗	0	0	✗	✗	0	0
Spindle	✓	14	16	✗	✗	1	1

Fig. 6

- In the column marked **Dir LowActive** change the tick to a cross (or vice versa) on the axis concerned to reverse the motor direction.
- Exit and restart Mach3 and check that the alteration has taken place.

Once all three axis are moving in the right direction, the machine is ready for use.

7.2 Limit Switches

Your KX Series CNC Mill is fitted with limit switches on all 3 axes. The purpose of these switches is to prevent each axis from travelling beyond the machine's safe limits, possibly causing damage to the machine. When a limit switch is triggered, the following happens:

- The machine stops moving
- The ***Reset*** button flashes red and green
- There will be a scrolling message reading "Press Reset - Emergency mode active"
- The ***Status*** line reads "Limit switch triggered"

Pressing ***Reset*** allows you to over-ride and jog off the limit switch. If you have driven a long way onto the limit switch, you may need to press ***Reset*** again before you can jog off the switch.

8. Setting a datum

You will need to find the exact position of two edges (or a centre) of the component you wish to machine to act as a datum (otherwise known as the Origin. There are various ways to find this; the most popular is an edge finder or wiggler. A wiggler consists of a ball jointed probe that when running spins the tip in a circle - Photo 1. As the tip touches the work the circle gradually reduces as it gets closer until at one point there is no movement then immediately after this the tip rolls along the edge of the work and 'flicks' out. This is the contact point of the edge of the wiggler and the work.

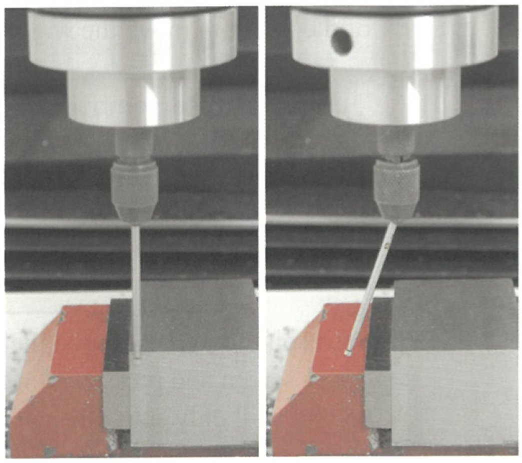


Photo 1 - A Wiggler in action

Wigglers are usually run at about 600 RPM but usually perform well over quite a large speed range.

Fit the wiggler into a holder - a standard end mill holder where the tool is held in by a grub screw is recommended. Next, the tool holder needs to be fitted to the spindle. Screw the tool into the drawbar and tighten with a spanner at the top and bottom of the spindle.

The drawbar of the KX1 CNC Mill is self extracting - Photo 2. This means there is a collar at the top of the spindle that stops the drawbar from being removed from the machine. When you undo the drawbar, it pushes against the collar and ejects the tool holder from the MT2 taper.

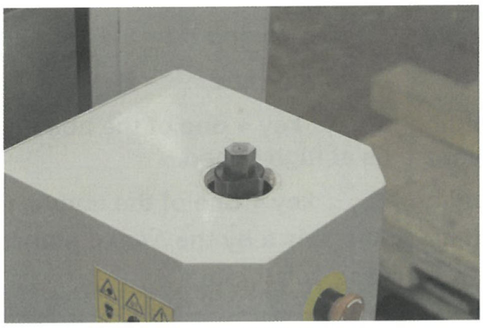


Photo 2 - The KX1 Drawbar

The KX3 CNC Mill is fitted with a standard drawbar which needs to be slackened a few threads and tapped to release the R8 or MT3 taper.

If you are using the metric system, it is recommended that you use an edge finder with a 6mm end and if using imperial, use a 0.200in end.

8.1 Zeroing the machine

Press the ***Zero X***, ***Zero Y*** and ***Zero Z*** buttons to the left of the machine DRO's (Fig. 7), this will set all the axes to read 0.0000.

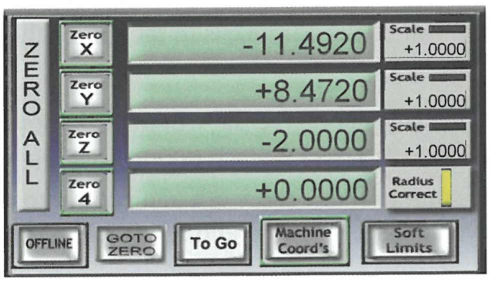


Fig 7

8.2 Using the jogging control

Press the [Tab] key on the keyboard and the **Jog** Flyout will appear on the right of the screen, Fig 8. First we'll zero the Y axis on the front of the work. Use the jog buttons ***X-***, ***X+***, ***Y-***, ***Y+***, ***Z-*** and ***Z+*** at the bottom of the flyout to move the wiggler close to the front of the work.

Fig 8

Remember:
When jogging, work as if the tool direction is moving, not the table!
If you forget, the table will move in the wrong direction!

Start spindle: type 600 into the ***set speed*** box on the **Program Run Alt-1** screen and press [Enter] on the keyboard to accept the entry. The spindle will only start when you click on the ***Spindle CW*** button or press [F5] on the keyboard. (To stop the spindle repeat the same command.) The end of the wiggler should now be running in a slightly eccentric circle.

Click the ***Y+*** button to jog towards the edge of the work.

HINT:
You can also jog using the keyboard:
[←] = X- and [→] = X+
[↑] = Y+ and [↓] = Y-
[Page Up] = Z+ and [Page Down] = Z-
Holding the [⇧] key + one of the above will jog the axis at high speed.
Holding the [Ctrl] key + one of the above will move the axis a by the *Step* distance entered in the Jog flyout.

You can alter the jogging speed (feed) by clicking the ***Slow Jog Rate*** buttons on the flyout. You can change from continuous movement (***Cont.***) to single step (***Step***) by clicking on the ***Jog Mode*** button and change the single step distance by clicking the ***Cycle Jog Step*** button or entering a distance directly into the ***Step*** field. Jog until the wobbler flicks out.

Press [F5] to stop the spindle.

At this point, click in the ***Y axis DRO*** and type in the radius of the wiggler. Since the wiggler is in front of the work, enter the wiggler **Radius** as a **Negative** value and press [Enter] to accept this value. (If you were using the wiggler at the rear of the work, you would enter a positive value)

Next we'll find zero for the X axis using the left side of the work. Start the spindle again and jog to the edge of the work until the wiggler flicks out and enter the radius of the wiggler in the ***X axis DRO*** and press [Enter]. Since the wiggler is to the left of the work, this value must also be **Negative**.

Stop the spindle and raise the wiggler above the work and remove it from the spindle. Press ***Zero Z*** at this point so the ***Z axis DRO*** reads 0.0000 above the work. Now press the ***Goto Zero*** button. This sets the spindle above the work and central to the bottom left corner you set up to. At this point we have now set the X, Y, and Z axis clear of the work.

This is known as the **Work Offset** and relates to the work offset being equal to the **Work Zero** or **Origin Point**.

The origin point is a very important point when you are drawing a part to machine; in short it's the start point.

There are another set of offsets called Machine Offsets or coordinates. These are for more experienced users who need to do multiple parts from one point. They are described in the main Mach3 User Guide but we are not going to deal with these in this Quick Start Guide but will make them equal the Work Offsets so there is no conflict.

This will be done after the tool height offsets have been set.

8.3 Setting the tool height offsets

Using the [Page Up] and [Page Down] keys on the keyboard or the ***Z+*** and ***Z-*** jogging keys on the ***Jog*** flyout, raise the spindle above the work a distance greater than the longest tool you plan to use. Clearance is needed between the bottom of the tool and the work to miss any clamps etc. This is called the **Clearance Height** or **Clearance Plane**. A good figure here is 30mm as this should be sufficient to clear the clamps - see Photo 3.

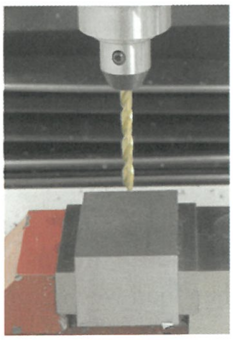


Photo 3

Once the longest tool is fitted, again ***Zero Z*** is pressed to remember this point.

Fit the tool to be used into the machine noting the tool number you are using or what the program needs. Click in the ***Tool*** number box in the Tool Information module, enter the tool number and press [Enter] to save it. Then enter the tool diameter in ***Dia.*** and again press [Enter] to save.

Jog the tool down until it just touches the work.

A time honoured method is to use a sheet of cigarette paper under the tool until it just traps it - Photo 4. At this point click the ***Set Tool Offset*** button. This enters the Z value into the Tool Table and turns the ***Tool Offset On/Off*** to On (indicated by a green LED) and shows 0.000 in the ***Z axis DRO*** as the tool is now at the working Z zero point.

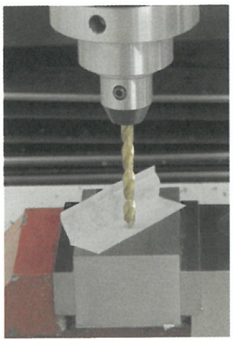


Photo 4

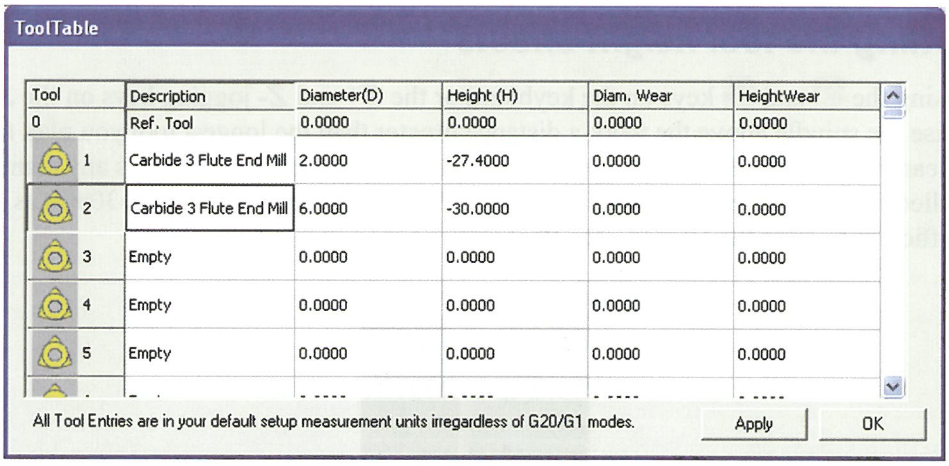
Click the ***Tool Offset On/Off*** button to turn it off. The green LED will go out and the ***Z axis DRO*** will now be reading the tool height offset.

Press the ***GOTO ZERO*** button and the tool will rise clear and go to its preset clearance plane.

CAUTION: If you fail to turn off the ***Tool Offset On/Off*** button ***before*** pressing the ***GOTO ZERO*** button, there is a risk for your tool to plough into your workpiece and cause damage.

Repeat this process for all the other tools you are using in the program.

At this point, all the work and tool offsets have been set. Go to the **Offsets Alt5** screen and click ***Save Tool Offsets*** to open the tool table - Fig 9 below.



Tool	Description	Diameter(D)	Height (H)	Diam. Wear	HeightWear
0	Ref. Tool	0.0000	0.0000	0.0000	0.0000
1	Carbide 3 Flute End Mill	2.0000	-27.4000	0.0000	0.0000
2	Carbide 3 Flute End Mill	6.0000	-30.0000	0.0000	0.0000
3	Empty	0.0000	0.0000	0.0000	0.0000
4	Empty	0.0000	0.0000	0.0000	0.0000
5	Empty	0.0000	0.0000	0.0000	0.0000

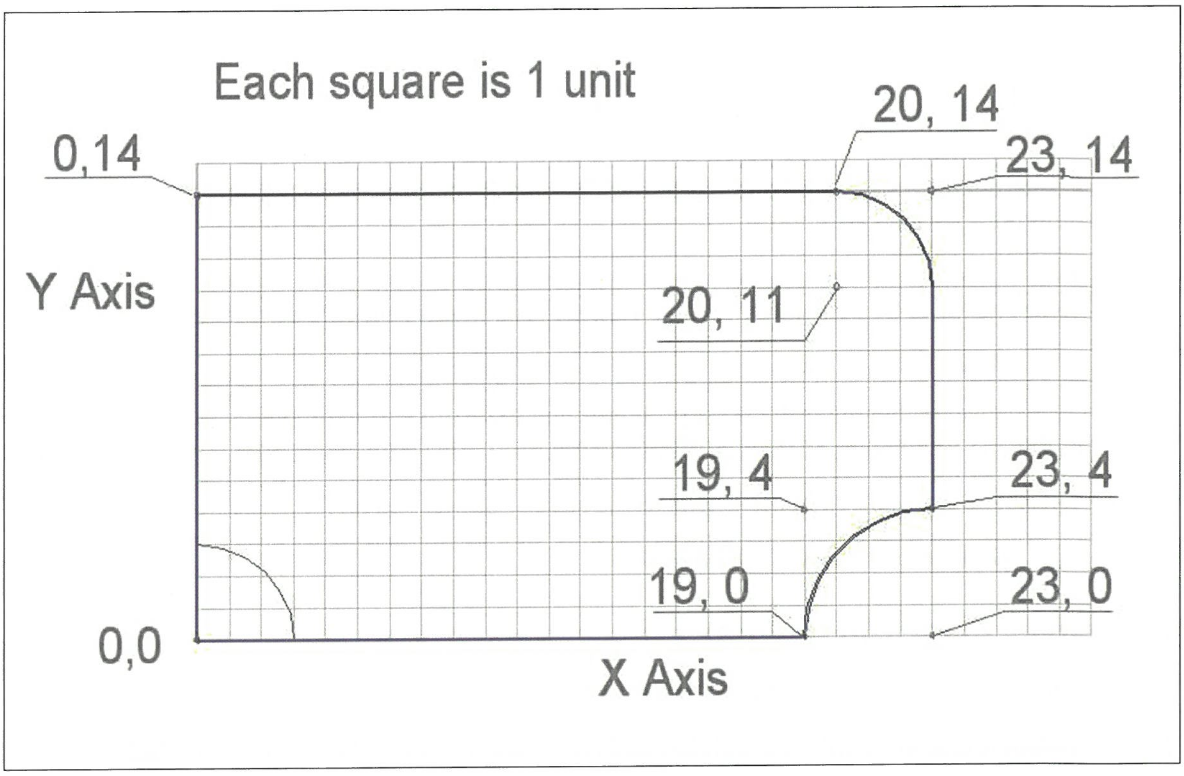
Fig. 9

It's not essential but you can enter a few words as a description. To save the tool table, click ***Apply*** and ***OK***. This table will then load when you next open Mach3.

The last job of setting up is to make the machine coordinates match the work coordinates.

Click the ***Tool Offset On/Off*** to turn it off and the green LED will go out. The ***Z axis DRO*** will now display the tool height offset so click ***ZERO ALL*** one last time to set Z back to 0.0000 and the machine is now in a full zero state.

9. Working with G Code



G-code is the name given to the instructions required by the Sieg KX series mills using Mach3 software. This code follows a loose Industry Standard in that many machines have special codes but with Mach3 the common ones are adhered to.

The four main moves are G00, G01, G02 and G03. The first two are linear moves and the latter two are arcs. Most shapes are a collection of these four moves.

G00 is a linear move at rapid rate. Rapid is defined as the maximum speed the machine can move. An example of this is: G00 X25.0. Assuming the tool is at zero then the tool will move plus 25 units in the X axis at maximum speed.

G01 is a linear move at a defined feed rate and can be expressed as: G01 X25.0 F 250.0. Again assuming the tool is at zero then this command will move the tool plus 25 units in the X axis at 250 units per minute.

G02 is a clockwise arc move and is defined as X, Y, I, J where X and Y are the end points of the arc, I is the incremental centre in the X plane and J is the incremental centre in the Y plane, both from the start point.

G03 is an anticlockwise move with the same definitions as the G02 move.

Below is the simple shape of a box with a radius on two of the corners and we will examine this shape and the code needed to trace the shape.

The origin point or 0,0 is in the bottom left and all moves will be made from this start point moving round in a clockwise direction.

9.1 G01 - Linear Move at a Defined Feed Rate

The first move will be in the Y axis to a point 14 units from 0,0 and we will move at a feed rate of 250 units per minute so out first line will look like:

```
G01 X0.0 Y14.0 F250.0
```

Next we need to move along the X axis for 20 units to get to the start of the first arc.

```
G01 X20.0 Y14.0 F250.0
```

9.2 G02 - Clockwise Arc

The next move is a G02 clockwise arc with the end point of the arc at X23.0 and Y11.0.

I relates to the centre of the arc in the X direction from the arc's start point, so since they are both the same, **I** = 0.0.

J is the centre of the arc in the Y direction from the arc's start point and since that was 14.0, the centre needs to be 11.0 and therefore **J** = -3.0. The line for the Arc then reads:

```
G02 X23.0 Y11.0 I0.0 J-3.0 F250.0
```

Arcs cause the beginner the most problems and there are two main points to remember, X and Y define the END point of the arc, I and J define the CENTRE incremental to the start point.

The shape then moves in the Y axis to the start of the second arc.

```
G01 X23.0 Y4.0 F250.0
```

9.3 G03 - Anticlockwise Arc

The last arc is anticlockwise so it's defined as G03. The end point is X19.0 and Y 0.0

The centre of the arc in X is still 0.0 and the centre in Y is -4.0 relative to the start. This gives the full line as:

```
G03 X19.0 Y0.0 I0.0 J-4.0 F250
```

To complete the shape the line then has to go back to its origin point.:

```
G01 X0.0 Y0.0 F250.0
```

This is just the basic code needed to follow the shape in the drawing, There are no Z axis moves or selecting the tools. At this point the code doesn't know what units it is working in, imperial or metric. Normally a CAM system generates these header and footer moves that define the rest of the program but as this is only a primer we will leave this to the owner to follow on.

9.4 The Safety Line

To get back to our code and putting it together with a couple of extra lines we get:

```
G21 G17 G90 G40 G49
G00 X0. Y0.
G01 X0.0 Y14.0 F250.0
G01 X20.0 Y14.0 F250.0
G02 X23.0 Y11.0 I0.0 J-3.0 F250.0
G01 X23.0 Y4.0 F250.0
G03 X19.0 Y0.0 I0.0 J-4.0 F250
G01 X0.0 Y0.0 F250.0
G28
M30
```

The first line sets the code to metric (G21), working in the X / Y plane (G17), absolute mode (G90) and G40 and G49 cancel any tool offsets that might still be running from the last program, this line is called the safety line and most programs such as dolphin and V Carve insert these lines automatically.

The next line forces the tool to go to the origin point if it's not already there. The G28 the end returns the axis to Zero and the M30 is the End of Program command.

It is important that this small example is followed through and understood as the whole method of writing and generating G-code is based on these simple moves.

With Mach3 running, load the program by clicking ***Load G-code*** on the **Program Run Alt-1** screen. Navigate to the GCode folder in Mach3 (e.g. C:\Mach3\GCode) and select the file **Demobox.tap**.

The program will now load in the G-code window and a preview will appear in the Toolpath screen. (If you like, you can view or edit the file by clicking the ***Edit G-code*** button which will open the file in Notepad)

Making sure the Z axis is clear of the table and the axis are zeroed with room to work, press cycle start and you can see it follow the shape in the drawing.

10. Ready to cut our first part

For the first time user it's advisable to do what is known as air cutting to get used to the machine and its settings.

At the side of the large red ***Reset*** button is a button called ***Z Inhibit***. It has an entry field and a two state button, ***On/Off***. If you type 6.000 into the DRO and ↵, then click the ***On/Off*** button so that the light comes on, it means that it will run a program but will ignore any Z commands with a value of less than +6.000 so the tool will be always 6.000mm above the work.

10.1 Cutting our first job

In this example we will use a file that has been previously generated in V Carve and is in the GCode folder, inside Mach3, the file is called Sieg_Logo.tap.

This is a simple engraving file that can be cut in any soft metal such as brass or aluminium or even MDF board.

The size of the part is confined into a circle 80mm in diameter and is designed to be cut with either a Vee cutter such as an engraving cutter or router cutter or even using a centre drill so no special cutters are needed for this first job. The origin point [0,0] is in the centre of the circle.

The depth of the engraving is 1mm deep so any material over 3mm will be ideal. It's advisable to use a piece larger than 80mm in diameter so that clamps can be used to hold it well out of the way of the tool.

10.2 Load the G Code

With Mach3 running, load the program by clicking ***Load G-code*** on the **Program Run Alt-1** screen. Navigate to the GCode folder in Mach3 (e.g. C:\Mach3\GCode) and select the file **Sieg_logo.tap**.

This file will run on either the KX1 or KX3 without any modifications.

10.3 Prepare the work.

The piece of material to be used should now be clamped onto the bed of the machine central to the travels of the X and Y axis. The origin or zero point is central to the work and can be found by working to a pair of diagonal crossed lines from corner to corner which will be as accurate as needed for this exercise.

Whilst learning it is also handy to mount your work on a sacrificial sheet of aluminium or MDF to give a degree of safety if you make any errors as it will give time to stop the machine before it damages the table.

10.4 Load the tool.

Fit the cutter into an appropriate collet, collet chuck or end mill holder depending on the size of the cutter and what tool holding equipment is available. Do not use a drill chuck to hold a milling cutter as these can and do work their way out, causing damage to the table and work. The rest of the tool loading is the same as covered in this Quick Start Guide earlier but we will go over it again. The tool used is tool number 20 and this has been coded into the program.

Jog the cutter over the work blank until it lines up with the centre mark you have previously made, at this point press ***Zero X*** and ***Zero Y*** at the side of the DRO's. This will set the origin point.

Because for this job we are only using one cutter we don't have to worry about setting all tools and dealing with just one tool makes it easier. Using the [Page Up] and [Page Down] keys jog the Z axis to a reasonable clearance plane above the work, for this job say 10mm and make sure all clamps are out of the cutting area.

In this position press ***Zero Z***. Then carefully jog down to the top of the work and using a strip of cigarette paper just trap the paper. This point is now the top of the work or the parts Z 0.00.

Type in 20 into the ***Tool*** number as this is the tool used in the program then press enter.

When typing values into DRO boxes it's important that you press enter and check the value is locked as if it's not pressed it remembers the previous value, often with dire results.

At this point click the ***Set Tool Offset*** button. This enters the Z value into the Tool Table and turns the ***Tool Offset On/Off*** to On (indicated by a green LED) and shows 0.000 in the ***Z axis DRO*** as the tool is now at the working Z zero point.

Click the ***Tool Offset On/Off*** button and the green LED will go out and the ***Z axis DRO*** will now be reading the tool height offset.

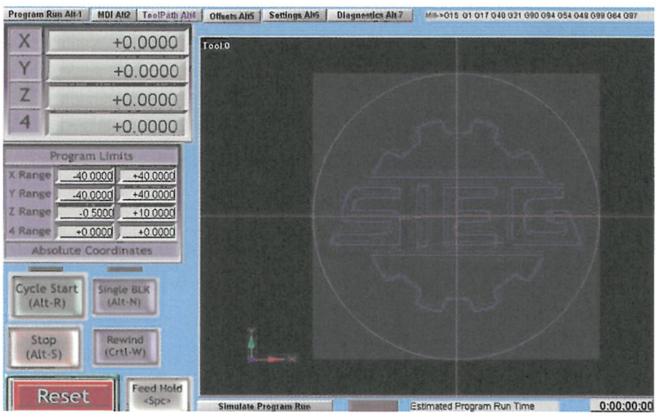
Press the ***GOTO ZERO*** button and the tool will rise clear and go to its preset clearance plane.

This now puts us in the correct position to start the job.

- The file is loaded.
- The tool is loaded.
- Work origin has been set.
- Tool offset has been set.

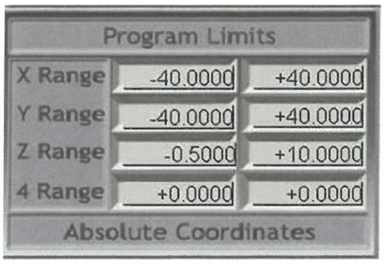
One last check we can do to satisfy ourselves that everything is correct is to see what the extents of the program are.

If we go to the toolpath screen from the top menu bar or Alt4, we get the following view.



This gives a bigger picture of what we are cutting but more importantly it shows the Program limits screen.

Looking at this shows how much room or travel is required in all axis and allows us to double check we have the room or we are not cutting too deep. In this case the maximum Z cut will be -0.5mm or 0.5mm deep INTO the part as it's a minus figure



You can now go back to the **Program Run Alt1** screen and press ***Cycle Start***.

It will load the tool then ask you to press ***Cycle Start*** again.

This is a safety measure to give you chance to check you have the right tool loaded.

The spindle will start and the tool will now feed into the work and commence to cut the Sieg logo. At the end of the cut it will stop the spindle and return to X0.0 Y0.0 Z10.0 and you will have cut your first part.

Happy Cutting !

11. CAM Software

On the install CD supplied with the machine there are two demo programs that can generate G Code. One is called Dolphin and is a full 2½D CAD and CAM package that is very powerful. The second program is called V Carve and is a sign making and engraving package but it can also do simple shapes.

Both these packages have tutorials on them and further details can be obtained from their respective web site with additional tutorials and videos.

12. Forum Based Support

12.1 SIEG CNC Machines (for machines running Mach3 only):

http://www.smallcncsupport.com/

This is the official support site. It provides on-line support on issues relating to your machine. To obtain the support, you must first be a bona fide owner of a SIEG CNC machine designed to be controlled by Mach3. Secondly, you must register your machine.

12.2 Registration Process

On the home page there is a log in box for registered users. Just underneath this is a link for new users to register. To register, you will need the Serial Number and Registration Number that came with your machine.

These two numbers ensure that only genuine owners can take advantage of the support. The support is forum based. As owners become more conversant with their machines we expect that all the owners will help each other.

As these machines get more popular we hope to add to the list with projects and ideas. Please use the feedback form to let us know what you would like to see.

12.3 Mach3 Support

Mach3 forum based support is available from:

http://www.machsupport.com/forum

For mach3 support issues relating to SIEG CNC machines, scroll down the page to the **Sieg Machines** forum in the **Third party software and hardware support forums** section. This site also has a good selection of downloadable video's and additional support.

12.4 C11G Breakout Board Support

The first point of contact for support with the C11G breakout board is via:

http://www.smallcncsupport.com/

Upon identification of a fault you will be referred to CNC4PC for guidance.

12.5 Other useful links

There are excellent Yahoo user groups for Mach3 and the C11G breakout board which can be found at:

http://groups.yahoo.com/group/mach1mach2cnc/

http://groups.yahoo.com/group/cnc4pc/

They are free to join and the members are very helpful.

E.&O.E.

13. Set-up Checklist

Following, is an abbreviated list of operations you need to carry out when setting up a job on your KX Series Mill.

Until you become more familiar with Mach3, we suggest to keep a copy of this page near the machine to act as a reminder.

- Start Computer and allow to boot
- Run the Mach3 Loader program
- Select the profile for your machine and Mach3 will then load
- Switch on the mill

13.1 Setting the Work Offset

- Click the ***Reset*** button to clear Emergency mode
- Fit edge finder or wiggler to spindle
- Click ***Zero X***, ***Zero Y*** and ***Zero Z*** to zero the DRO's
- Enter a spindle speed of 600rpm and start the spindle
- Jog the Y-Axis until the wiggler flicks out
- Enter wiggler radius into ***Y-axis DRO*** (+or -) and press ⏎ to accept (see page A-8-2 for further details)
- Jog the X-Axis until the wiggler flicks out
- Enter wiggler radius into ***X-axis DRO*** (+ or -) and press ⏎ to accept
- Raise spindle and remove wiggler and Click ***Zero Z***
- Click ***Goto Zero*** - this is the Work Offset or Origin Point

13.2 Setting the Tool Height Offsets

- Raise spindle above the work greater than longest tool
- Press ***Zero Z***
- Fit the longest tool to be used on this program
- Enter the tool number and diameter
- Jog the X and Y Axes placing the tool over the work
- Jog the Z-Axis down until it just touches the work
- Click the ***Set Tool Offset*** button
- Click the ***Tool Offset On/Off*** button to switch it off
- Click ***Goto Zero***
- Repeat the above for any other tools required in the program
- Go to the **Offsets Alt5** screen and click ***Save Tool Offset***
- Click ***Apply*** and ***OK*** to save the tool table
- Click the ***Tool Offset On/Off*** button to switch it off again
- Return to the **Program Run Alt1** screen
- Click ***Zero All*** button to set a full zero state

Index

1. Preface

Any machine tool is potentially dangerous. Computer controlled machines are potentially more dangerous than manual ones because, for example, a computer is quite prepared to rotate any size of cutter at 3000 rpm, to plunge an end mill, slot drill or drill straight into a piece of solid material, the machine table or the vice.

This manual tries to give you guidance on safety precautions and techniques but because we do not know your level of experience, we can accept no responsibility for the performance of the Sieg KX Series milling machine or any damage or injury caused by its use. Safety glasses should be work at all times while operating the Sieg KX Series CNC milling machine.

It is your responsibility to ensure that you understand the implications of how you set up and program the mill and to comply with any legislation and codes of practice applicable to your country or state.

If you are in any doubt, you must seek guidance from a professionally qualified expert rather than risk injury to yourself or to others.

This document is intended to give enough details about how the Mach3Mill software interacts with your Sieg mill, how it is configured for different axis drive methods and about the input languages and formats supported for programming to enable you to implement a powerful CNC system on your Sieg KX Series CNC milling machine.

An online wiki format document, *Customising Mach3,* explains in detail how to alter screen layouts, to design your own screens and Wizards and to interface to special hardware devices.

You are strongly advised to join one or both of the online discussion forums for Mach3. Links to join it are at *www.machsupport.com*

These forums have many engineers with a vast range of experience as participants and help and advice is freely offered.

Thanks are due to numerous people including the original team who worked at National Institute for Standards and Testing (NIST) on the EMC project and the users of Mach3 whose experience, materials and constructive comments helped greatly in the production of this manual. Credits are given for individual utilities and features as these are described in the body of the manual.

2. Introducing CNC machining systems

2.1 Parts of a machining system

This chapter will introduce you to the terminology used in the rest of this manual and allow you to understand the purpose of the different components in a numerically controlled milling system.

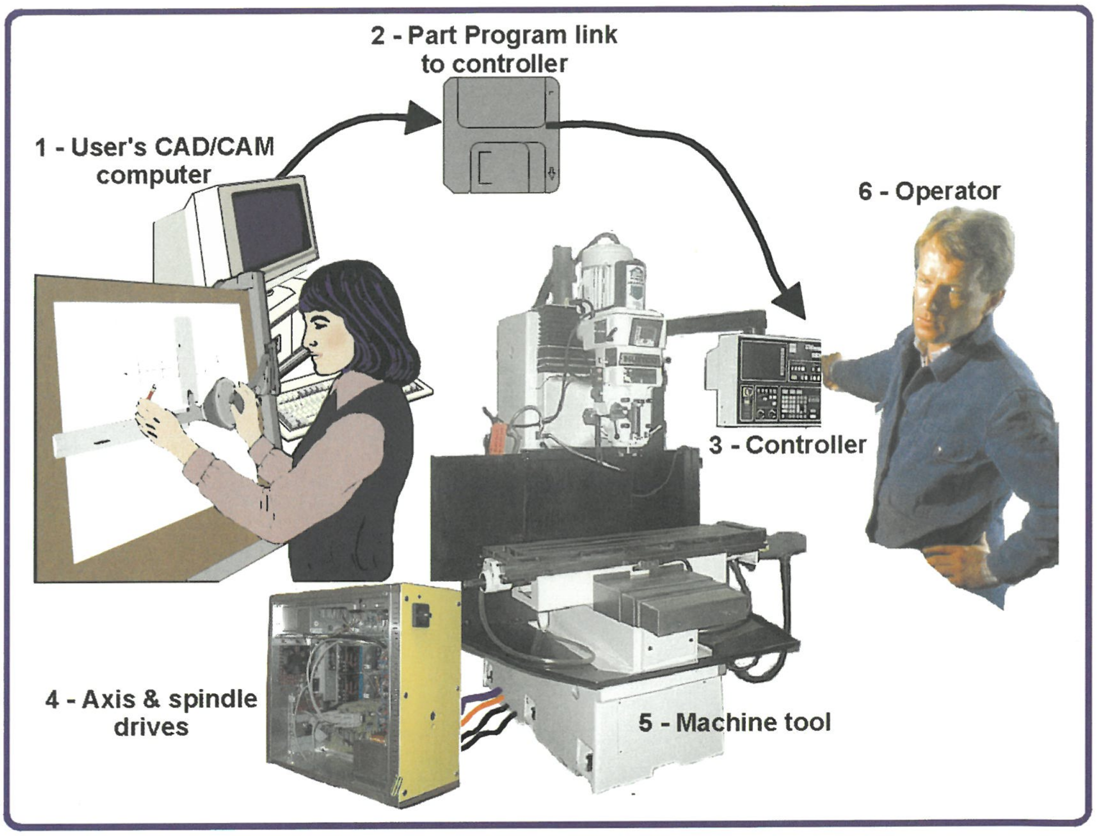


Figure 2.1 – A typical CNC machining system

The main parts of a system for numerically controlled mill are shown in figure 2.1

The designer of a part generally uses a Computer Aided Design/Computer Aided Manufacturing (CAD/CAM) program or programs on a computer (1). The output of this program, which is a part program and is often in "G-code" is transferred (by a network or perhaps floppy disc) (2) to the Machine Controller (3). The Machine Controller is responsible for interpreting the part program to control the tool which will cut the workpiece. The axes of the Machine (5) are moved by screws, racks or belts which are powered by servo motors or stepper motors. The signals from the Machine Controller are amplified by the Drives (4) so that they are powerful enough and suitably timed to operate the motors.

Frequently the Machine Controller can control starting and stopping of the spindle motor (or even control its speed) and will check that a part program or Machine Operator (6) are not trying to move any axis beyond its limits.

The Machine Controller usually has controls like buttons or a keyboard, so that the Operator can control the machine manually and start and stop the running of the part program. The Machine Controller has a display so that the Operator knows what is happening.

Because the commands of a G-code program can request complicated co-ordinated movements of the machine axes, the Machine Controller has to be able to perform a lot of calculations in "real-time" (e.g. cutting a helix requires a lot of trigonometrical calculation). Historically this made it an expensive piece of equipment.

2.2 How Mach3 fits in

Mach3 is a software package which runs on a PC and turns it into a very powerful and economical Machine Controller to replace (3) in figure 2.1.

To run Mach3 you need Windows XP (or Windows 2000) ideally running on a 1GHz processor with a 1024 x 768 pixel resolution screen. A desktop machine will give much better performance than most laptops and will be considerably cheaper. You can of course use this computer for any other functions in the workshop (such as (1) in figure 2.1 - running a CAD/ CAM package) when it is not controlling your machine.

Mach3 communicates principally via one (or optionally two) parallel (printer) ports and, if desired, a serial (COM) port.

3. An overview of Mach3 Machine Controller software

3.1 Installation

Mach3 is distributed by ArtSoft Corp. via the Internet. You download the package as one self installing file (which, in the present release, is about 8 megabytes). This will run for an unlimited period as a demonstration version with a few limitations on the speed, the size of job that can be undertaken and the specialist features supported. When you purchase a licence this will "unlock" the demonstration version you have already installed and configured. Full details of pricing and options are on the ArtSoft Corporation website www.artofcnc.ca

3.1.1 Downloading

Download the package from www.artofcnc.ca using the right mouse button and *Save Target as*... to put the self-installing file in any convenient working directory (perhaps Windows\Temp). You should be logged in to Windows as an Administrator.

When the file has downloaded it can be immediately run by using the *Open* button on the download dialog or this dialog can be closed for later installation. When you want to do the installation you merely run the downloaded file. For example you could run Windows Explorer (right click *Start* button), and double-click on the downloaded file in the working directory.

3.2 Installing

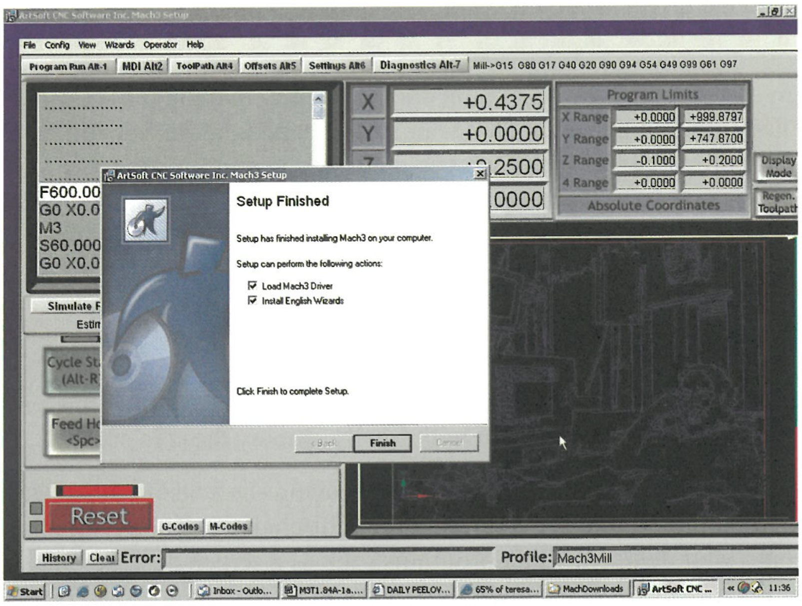


Figure 3.1 – The installer screen

You do not need the Sieg KX Series CNC milling machine connected yet. It would be better not to have one connected at this stage. Note if the cable or cables from the Sieg KX Series CNC milling machine are plugged into your PC, Switch off the PC, the machine tool and its drives and unplug the 25 pin connector(s) from the back of the PC. Now switch the PC back on.

When you run the downloaded file, you will be guided through the usual installation steps for a Windows program such as accepting the license conditions and selecting the folder for Mach3. On the Setup Finished dialog you should ensure that *Initialise System* is checked and click *Finish*. You will now be told to reboot before running any Mach3 software.

The background image during installation is the standard Mach3Mill screen.

On the Setup Finished dialog you should ensure that *Load Mach3 Driver* and *Install English Wizards* are checked and then click *Finish*. You will now be told to reboot before running any Mach3 software.

3.2.1 The vital re-boot

This reboot is **vital**. If you do not do it then you will get into great difficulties which can only be overcome by using the Windows Control Panel to uninstall the driver manually. **So please reboot now.**

If you are interested in knowing why the reboot is required then read on, otherwise skip to the next section.

Although Mach3 will appear to be a single program when you are using it, it actually consists of two parts: a driver which is installed as part of Windows like a printer or network driver and a graphical user interface (GUI).

The driver is the most important and ingenious part. Mach3 must be able to send very accurately timed signals to control the axes of the machine tool. Windows likes to be in charge and runs normal user programs when it has nothing better to do itself. So Mach3 cannot be a "normal user program"; it must be at the lowest level inside Windows (that is, it handles interrupts). Furthermore, to do this at the high speeds possibly required (each axis can be given attention 45,000 times per second), the driver needs to tune its own code. Windows does not approve of this, (it's a trick that viruses play) so it has to be asked to give special permission. This process requires the reboot. So if you have not done the re-boot then Windows will give the Blue Screen of Death and the driver will be corrupt. The only way out of this will be to manually remove the driver.

Having given these dire warnings, it is only fair to say that the reboot is only required when the driver is first installed. If you update your system with a newer version then the reboot is not vital. The install sequence does however, still ask you to do it. Windows XP boots reasonably quickly so it is not much hardship to do it every time.

3.2.2 Convenient desktop icons

So you **have** rebooted! The installation wizard will have created desktop icons for the main programs. Mach3.exe is the actual user interface code. If you run it, it will ask which Profile you wish to use. Mach3Mill, Mach3Turn etc. are shortcuts which run with a Profile defined by a "/p" argument in the shortcut target. You will usually employ these to start the required system.

It is now worthwhile to setup some icons for desktop shortcuts to other Mach3 programs. Use Windows Explorer (right-click *Start*) and by right-clicking on the DriverTest.exe file. Drag this shortcut onto your desktop. Other programs such as a screen designer and a manipulator for screen set files are available as a separate download.

3.2.3 Testing the installation

It is now highly recommended to test the system. As mentioned above, Mach3 is not a simple program. It takes great liberties with Windows in order to perform its job; this means it will not work on all systems due to many factors. For example, the QuickTime system monitor (qtask.exe) running in the background can kill it and there will be other programs which you probably are not even aware are on your system that can do the same. Windows can and does start many processes in the background; some appear as icons in the system tray (bottom right of screen) and others do not show themselves in any way.

Other possible sources of erratic operation are local area network connections which may be configured to automatically speed detect. You should configure these to the actual speed 10 Mbps or 100 Mbps of your network.

Finally a machine that has been surfing the Internet may have gained one or more of a host of "robot" type programs which spy on what you are doing and send data over the 'net to their originators. This traffic can interfere with Mach3 and is not something you want anyway. Use a search engine for terms like "Spybot" to locate software to tidy up your machine.

Because of these factors, it is important, though not mandatory, that you test your system when you suspect something is wrong or you just want to check that an install went well.

Double click the DriverTest icon that you set up. Its screen shot is in figure 3.2.

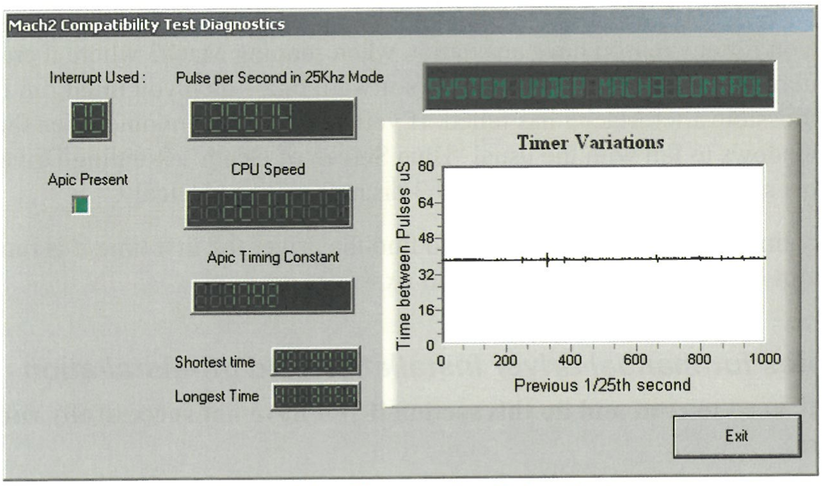


Figure 3.2 – The running DriverTest

You can ignore all the boxes with the exception of the Pulse Frequency. It should be fairly steady around 25,000 Hz but yours may vary, even quite wildly. This is because Mach3 uses the Windows clock to calibrate its pulse timer and, over a short time scale, the Windows clock can be affected by other processes loading the computer. So you may actually be using an "unreliable" clock (the Windows one) to check Mach3 and so get the false impression that Mach3's timer is unsteady.

Basically, if you see a similar screen to figure 3.2 with only small spikes on the Timer Variations graph and a steady Pulse Frequency, everything is working well so **close the DriverTest program and skip to the section *Screens*** below.

Windows "experts" might be interested to see a few other things. The white rectangular window is a type of timing analyzer. When it is running it displays a line with small variations indicated. These variations are the changes in timing from one interrupt cycle to another. There should be no lines longer than ¼ inch or so on a 17" screen on most systems. Even if there are variations, it's possible they are below the threshold necessary to create timing jitters so when your machine tool is connected, you should perform a movement test to see if jogging and G0/G1 moves are smooth.

You may have one of two things happen to you when running the test which may indicate a problem.

1. "Driver not found or installed, contact Art.", this means that the driver is not loaded into Windows for some reason. This can occur on XP systems which have a corruption of their driver database. Reloading Windows is the cure in this case. Or, you may be running Win2000. Win2000 has a bug/"feature" which interferes with loading the driver. It may need to be loaded manually ,see the next section

2. When the system says, taking over…3…2…1.. and then reboots, one of two things has occurred. Either you didn't reboot when asked (told you!!) or the driver is corrupted or unable to be used in your system. In this case, follow the next section and remove the driver manually, then re-install. If the same thing happens again, please notify ArtSoft using the e-mail link on www.artofcnc.ca and you will be given guidance.

A few systems have motherboards which have hardware for the APIC timer but whose BIOS code does not use it. This will confuse Mach3 install. A batch file SpecialDriver.bat is available in the Mach3 installation folder. Find it with Windows Explorer and double-click it to run it. This will make the Mach3 driver use the older i8529 interrupt controller. You will need to repeat this process whenever you download an upgraded version of Mach3 as installing the new version will replace the special driver. The file OriginalDriver.bat reverses this change.

3.2.4 Driver Test after a Mach3 crash

Should you for any reason have a situation when running Mach3 where it crashes - this might be an intermittent hardware problem or a software bug – then you **must** run DriverTest.exe as soon as possible after Mach3 has failed. If you delay for two minutes then the Mach3 driver will cause Windows to fail with the usual "Blue Screen of Death". Running DriverTest resets the driver to a stable condition even if Mach3 disappears unexpectedly.

You may find, after a crash that it fails to find the driver the first time it is run. In this case merely run it again as the first run should fix things up.

3.2.5 Notes for manual driver installation and un-installation

You only need to read and do this section if you have not successfully run the DriverTest program.

The driver (Mach3.sys) can be installed and uninstalled manually using the Windows control panel. The dialog boxes differ slightly between Windows 2000 and Windows XP but the steps are identical.

- Open the Control panel and double-click on the icon or line for *System*.
- Select *Hardware* and click *Add Hardware wizard*. (As mentioned before Mach3's driver works at the lowest level in Windows). Windows will look for any new actual hardware (and find none).
- Tell the wizard you have already installed it and then proceed to the next screen.
- You will be shown a list of hardware. Scroll to the bottom of this and select *Add a new hardware device* and move to the next screen.
- On the next screen, you do not want Windows to search for the driver so select *Install the hardware that I manually select from a list (Advanced)*
- The list you are shown will include an entry for *Mach1/2 pulsing engine*. Select this and go to the next screen.
- Click *Have disc* and on the next screen point the file selector to your Mach3 directory (C:\Mach3 by default). Windows should find the file *Mach3.inf*. Select this file and click *Open*. Windows will install the driver.

The driver can be uninstalled rather more simply.

- Open the Control panel and double-click on the icon or line for System.
- Select *Hardware* and click *Device Manager.*
- You will be shown a list of devices and their drivers. *Mach1 Pulsing Engine* has the driver *Mach3 Driver* under it. Use the + to expand the tree if necessary. Right-click on Mach3 Driver gives the option to uninstall it. This will remove the file Mach3.sys from the Windows folder. The copy in the Mach3 will still be there.

There is one final point to note. Windows remembers all the information about the way you have configured Mach3 in a Profile file. This information is not deleted by un-installing the driver and deleting other Mach3 files so it will remain whenever you upgrade the system. However in the very unlikely event that you need a totally clean installation from scratch then you need to delete the .XML profile file or files.

3.3 Screens

You are now ready to try out a "dry run" Mach3. It will be much easier to show you how to set up your actual machine tool when you have experimented with Mach3 like this. You can "pretend" to machine and learn a lot before connecting the Sieg KX Series CNC milling machine to the PC.

Mach3 is designed so that it is very easy to customize its screens to suit the way you work. This means that the screens you see may not look exactly like those in Appendix 1.

Double-click the Mach3Mill icon to run the program. You should see the Mill Program Run screen similar to that in Appendix 1 (but with the various DROs set to zero, no program loaded etc.).

Notice the red Reset button. It will have a flashing Red/Green LED (simulation of a light emitting diode) above it and some yellow LEDs lit. If you click the button then the yellow LEDs go out and the flashing LED turns to solid green. Mach3 is ready for action!

If you cannot reset, then the problem is probably something plugged into your parallel port or ports (a "dongle" perhaps) or the PC has previously had Mach3 installed on it with an unusual allocation of port pins to the Emergency Stop (EStop signal). By clicking on the *Offline* button you should be able to reset the system. **Most of the tests and demonstrations in this chapter will not work unless Mach3 is reset out of the EStop mode.**

3.3.1 Types of object on screens

You will see that the Program Run screen is made up of the following types of object:

- Buttons (e.g. Reset, Stop Alt-S, etc.)
- DROs or Digital Readouts. Anything with a number displayed will be a DRO. The main ones are, of course the current positions of the X, Y, Z, A, B & C axes.
- LED's (in various sizes and shapes)
- The G-code display window (with its own scroll bars)
- The Tool path display (blank square on your screen at the moment)

There is one further important type of control that is not on the Program Run screen:

- MDI (Manual Data Input) line

Buttons and the MDI line are your inputs to Mach3.

DROs can be displays by Mach3 or can be used as inputs by you. The background colour changes when you are inputting.

The G-code window and Tool path displays are for information from Mach3 to you. You can, however, manipulate both of them (e.g. scrolling the G-code window, zooming, rotating and panning the Tool path display)

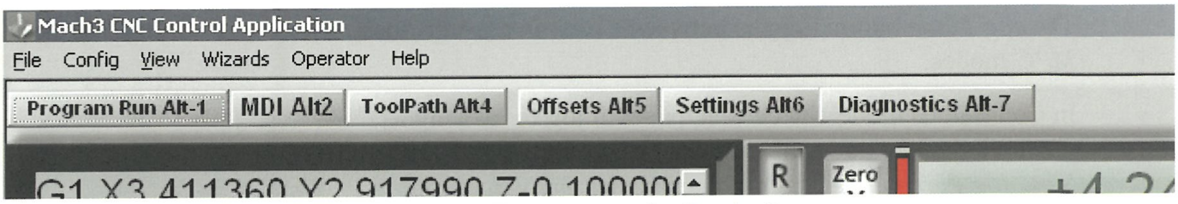


Figure 3.3 - The screen selection buttons

3.3.2 Using buttons and shortcuts

On the standard screens, most buttons have a keyboard hot key. This will be shown after the name on the button itself or in a label near it. Pressing the named key when the screen is displayed is the same as clicking the button with the mouse. You might like to try using the mouse and keyboard shortcuts to turn on and off the spindle and to switch to the MDI screen. Notice that letters are sometimes combined with the *Control* or *Alt* keys. Although letters are shown as uppercase (for ease of reading) you do **not** use the shift key when using the shortcuts.

In a workshop it is convenient to minimise the times when you need to use a mouse. Physical switches on a control panel can be used to control Mach3 by use of a keyboard emulator board (e.g. Ultimarc IPAC). This plugs in series with your keyboard and sends Mach3 "pretend" key presses which activate buttons with shortcuts.

If a button does not appear on the current screen then its keyboard shortcut is not active.

There are certain special keyboard shortcuts which are global across all screens. Chapter 5 shows how these are set up.

3.3.3 Data entry to DRO

You can enter new data into any DRO by clicking in it with the mouse, clicking its hot key (where set) or by using the global hot key to select DROs and moving to the one that you want with the arrow keys)

Try entering a feed rate like 45.6 on the Program Run screen. You **must** press the *Enter* key to accept the new value or the *Esc* key to revert to the previous one. *Backspace* and *Delete* are not used when inputting to DROs.

Caution: It is not always sensible to put your own data into a DRO. For example the display of your actual spindle speed is computed by Mach3. Any value you enter will be overwritten. You can put values into the axis DROs but you should not do it until you have read this entire manual in detail. This is **not** a way of moving the tool!

3.4 Jogging

You can move the tool relative to any place on your work manually by using various types of Jogging. The jogging controls are on a special "fly-out" screen. This is shown and hidden by using the *Tab* key on the keyboard. Figure 3.4 gives a view of the fly-out.

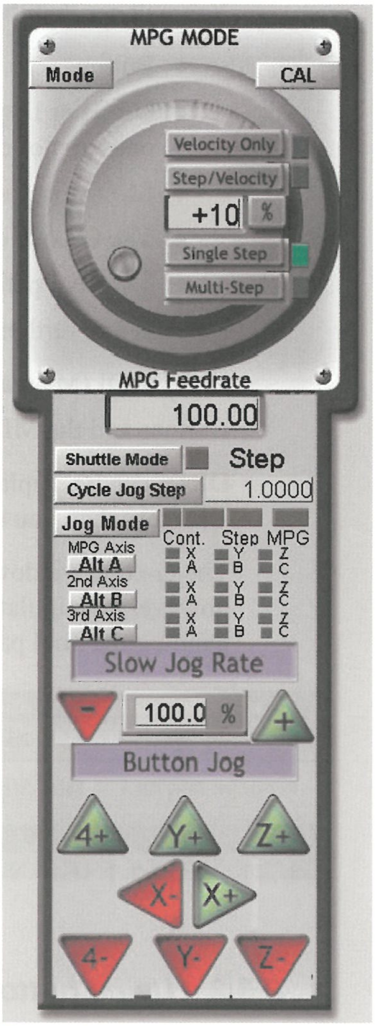


Figure 3.4 - Jog controls (use the Tab key to show and hide this).

You can use the keyboard for jogging. The arrow keys are set by default to give you jogging on the X and Y axes and Pg Up/ Pg Dn jogs the Z axis. You can re-configure these keys to suit your own preferences. You can use the jogging keys on any screen with the *Jog ON/OFF* button on it.

In figure 3.4 you will see that the Step LED is shown lit. The *Jog Mode* button toggles between *Continuous, Step* and *MPG* modes,

In Continuous mode, the chosen axis will jog for as long as you hold the key down. The speed of jogging is set by the *Slow Jog Percentage* DRO. You can enter any value from 0.1% to 100% to get whatever speed you want. The Up and Down screen buttons beside this DRO will alter its value in 5% steps. If you depress the *Shift* key then the jogging will occur at 100% speed whatever the override setting. This allows you to quickly jog to near your destination and the position accurately.

In Step mode, each press of a jog key will move the axis by the distance indicated in the *Step* DRO. You can set this to whatever value you like. Movement will be at the current Feed rate. You can cycle through a list of predefined Step sizes with the *Cycle Jog Step* button.

Another option for jogging is a joystick connected to the PC games port or USB. Mach3 will work with any Windows compatible "analogue joystick" (so you could even control your X axis by a Ferrari steering wheel!). The appropriate Windows driver will be needed for the joystick device. The 'stick is enabled by the *Joystick* button and, for safety, must be in the central position when it is enabled.

If you have an actual joystick and it has a throttle control then this can be configured either to control the jog override speed or the control the feed rate override (see Chapter 5). Such a joystick is a cheap way of providing very flexible manual control of your machine tool. In addition, you can use multiple joysticks (strictly Axes on Human Interface Devices) by installing manufacturer's profiler software or, even better, the Key Grabber utility supplied with Mach.

Now would be a good time to try all the jogging options on your system. Don't forget that there are keyboard shortcuts for the buttons, so why not identify them and try them. You should soon find a way of working that feels comfortable.

3.5 Manual Data Input (MDI) and teaching

3.5.1 MDI

Use the mouse or keyboard shortcut to display the MDI (Manual Data Input) screen.

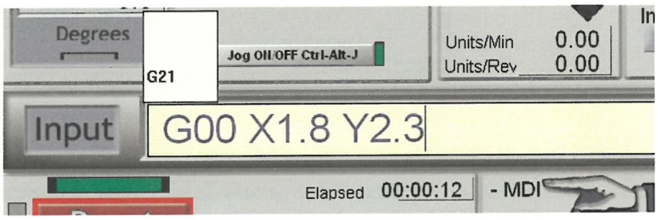


Figure 3.5 – MDI data being typed

This has a single line for data entry. You can click in it to select it or use press *Enter* which will automatically select it. You can type any valid line that could appear in a part program and it will be executed when you press *Enter*. You can discard the line by pressing *Esc*. The *Backspace* key can be used for correcting mistakes in your typing.

If you know some G-code commands then you could try them out. If not then try:

```
G00 X1.6 Y2.3
```

Which will move the tool to coordinates X = 1.6 units and Y = 2.3 units. (It is G zero not G letter O). You will see the axis DROs move to the new coordinates.

Try several different commands (or G00 to different places). If you use the up or down arrow keys while in the MDI line, you will see that Mach3 scrolls you backwards and forwards through the history of commands you have used. This makes it easy to repeat a command without having to re-type it. When you select the MDI line you will have noticed a fly out box giving you a preview of this remembered text.

An MDI line (or block as a line of G-code is sometimes called) can have several commands on it and they will be executed in the "sensible" order as defined in Chapter 9 - not necessarily from left to right. For example, setting a feed speed by something like F2.5 will take effect before any feed speed movements even if the F2.5 appears in the middle or even at the end of the line (block). If in doubt about the order that will be used then type several separate MDI commands in one by one.

3.5.2 Teaching

Mach3 can remember a sequence of lines that you enter using MDI and write them to a file. This can then be run again and again as a G-code program.

On the MDI screen (figure 3.6), click the *Start Teach* button. The LED next to it will light to remind you that you are teaching. Type in a series of MDI lines and Mach3 will execute them as you press return after each line and store them in a conventionally named Teach file. When you have finished, click *Stop Teach.*

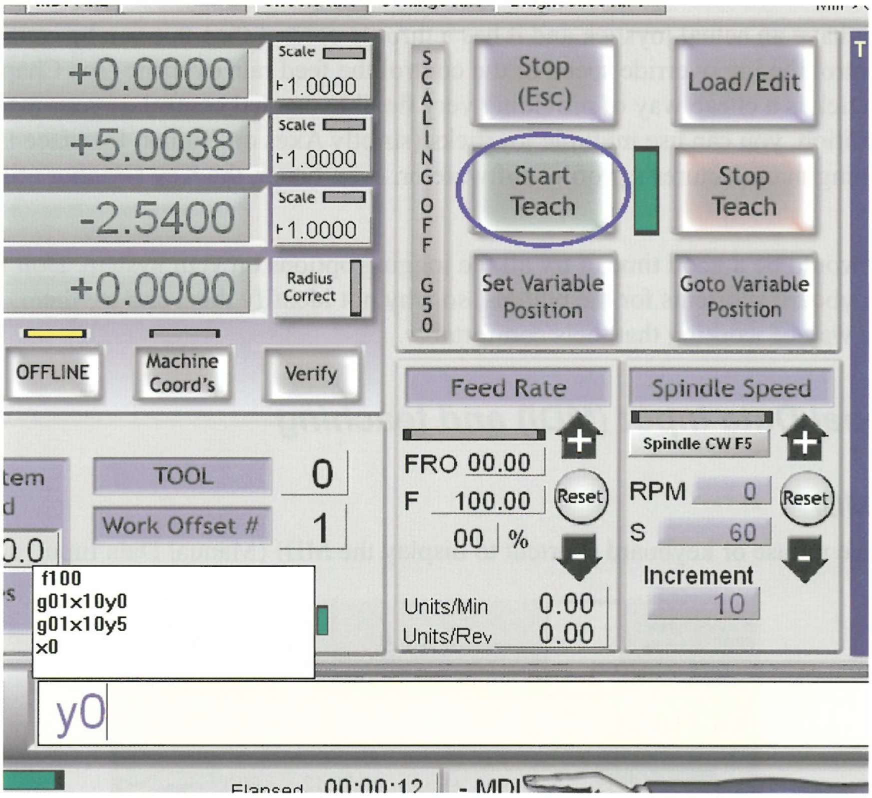


Figure 3.6 – Taught program running

You can type your own code or try:

```
g21
f100
g1 x10 y0
g1 x10 y5
x0
y0
```

All the 0 are zeros in this. Next click *Load/Edit* and go to the Program Run screen. You will see the lines you have typed are displayed in the G-code window (figure 3.7). If you click Cycle Start then Mach3 will execute your program.

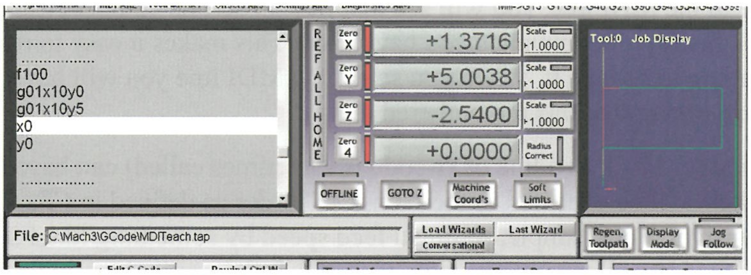
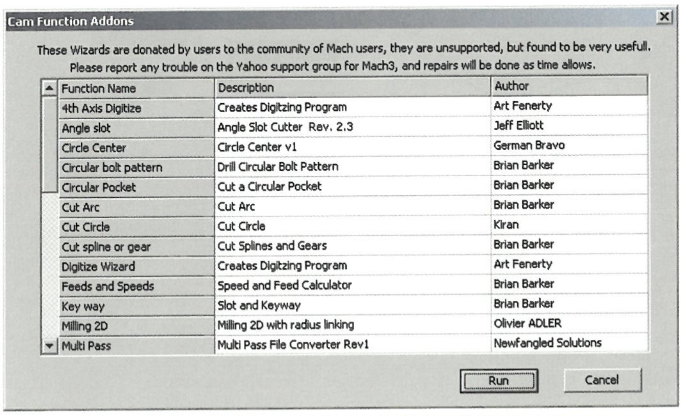


Figure 3.7 – In the middle of teaching a rectangle

When you have used the editor you will be able to correct any mistakes and save the program in a file of your own choosing.

3.6 Wizards – CAM without a dedicated CAM software

Mach3 allows the use of add-on screens which allow the automation of quite complex tasks by prompting the user to provide the relevant information. In this sense they are rather like the so-called Wizards in much Windows software that guides you through the information required for a task. The classic Windows Wizard will handle tasks like importing a file to a database or spreadsheet. In Mach3, examples of Wizards include cutting a circular pocket, drilling a grid of holes or digitising the surface of a model part.



Function Name	Description	Author
4th Axis Digitize	Creates Digitzing Program	Art Fenerty
Angle slot	Angle Slot Cutter Rev. 2.3	Jeff Elliott
Circle Center	Circle Center v1	German Bravo
Circular bolt pattern	Drill Circular Bolt Pattern	Brian Barker
Circular Pocket	Cut a Circular Pocket	Brian Barker
Cut Arc	Cut Arc	Brian Barker
Cut Circle	Cut Circle	Kiran
Cut spline or gear	Cut Splines and Gears	Brian Barker
Digitize Wizard	Creates Digitzing Program	Art Fenerty
Feeds and Speeds	Speed and Feed Calculator	Brian Barker
Key way	Slot and Keyway	Brian Barker
Milling 2D	Milling 2D with radius linking	Olivier ADLER
Multi Pass	Multi Pass File Converter Rev1	Newfangled Solutions

Figure 3.8 – Table of Wizards from Wizard menu

It is easy to try one out. In the Program Run screen, click *Load Wizards*. A table of the Wizards installed on your system will be displayed (figure 3.8). As an example click on the line for *Circular pocket*, (this is in the standard Mach3 release), and click Run.

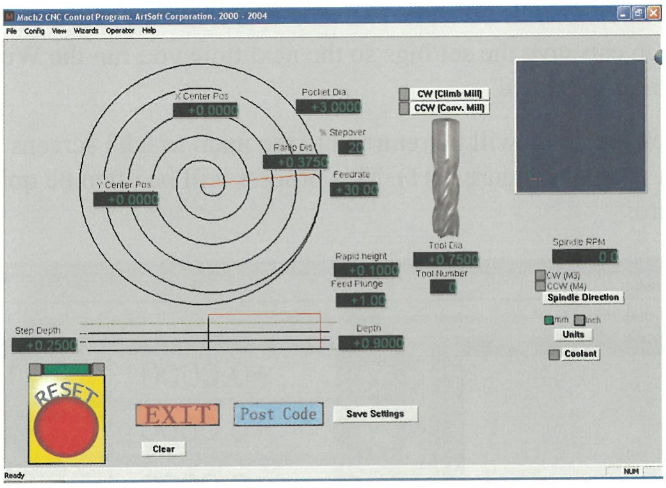


Figure 3.9 – Circular pocket with defaults

The Mach3 screen currently displayed will be replaced by the one shown in figure 3.9. This shows the screen with some default options. Notice that you can choose the units to work in, the position of the centre of the pocket, how the tool is to enter the material and so on. Not all the options might be relevant to the Sieg KX Series CNC milling machine. In this case you can ignore the controls on the Wizard screen.

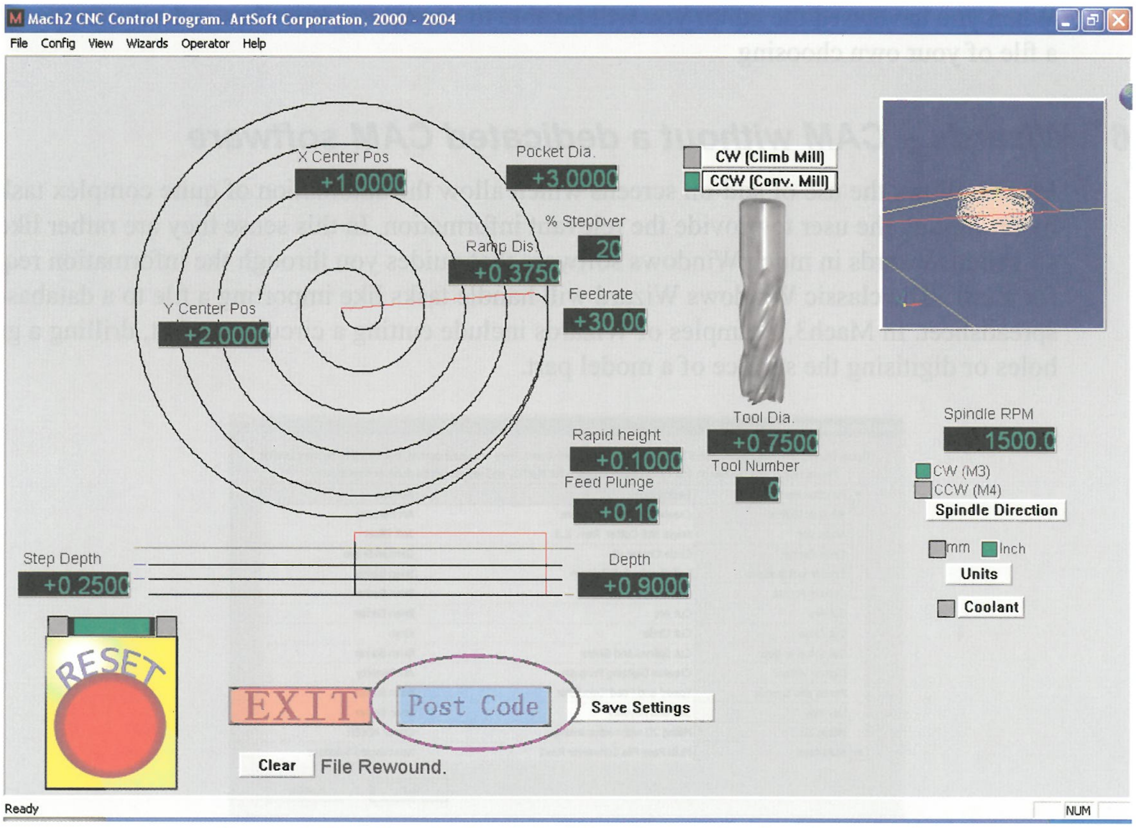


Figure 3.10 – Circular Pocket with values set and code posted

When you are satisfied with the pocket, click the *Post Code* button (figure 3.10). This writes a G-code part program and loads it into Mach3. This is just an automation of what you did in the example on Teaching. The tool path display shows the cuts that will be made. You can revise your parameters to take smaller cuts or whatever and re-post the code.

If you wish, you can save the settings so the next time you run the Wizard, the initial data will be what is currently defined.

When you click *Exit*, you will be returned to the main Mach3 screens and can run the Wizard-generated part program (figure 3.11). This process will be often be quicker than reading the description here.

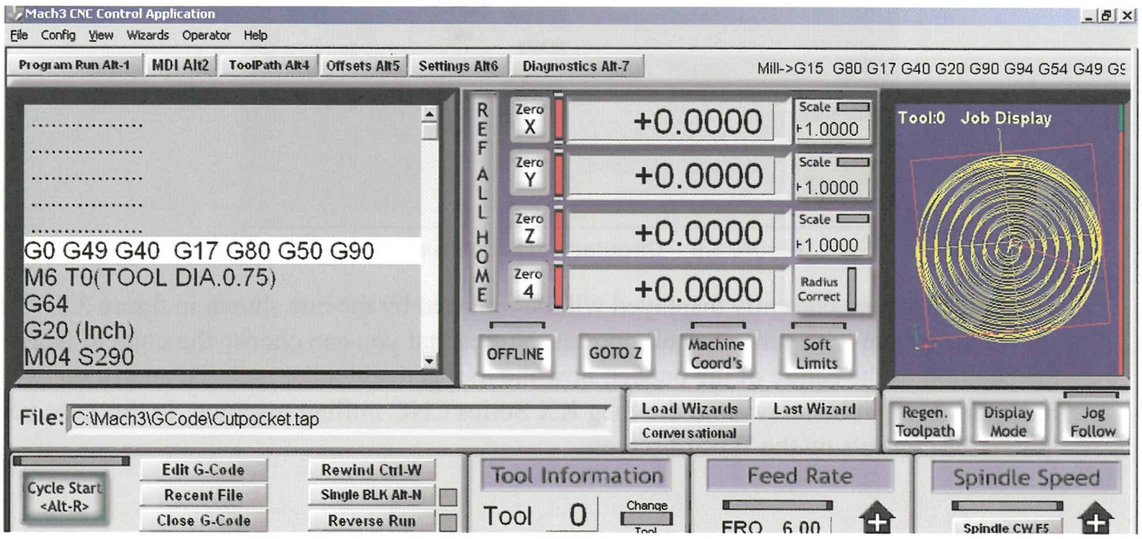


Figure 3.11 – The result of Circular Pocket ready to run

3.7 Running a G-code program

Now it is time to input and edit a Part Program. You will normally be able to edit programs without leaving Mach3 but as we have not yet configured it to know which editor to use, it is easiest to set up the program outside Mach3.

Use Windows Notepad to enter the following lines into a text file and save it in a convenient folder (My Documents perhaps) as `spiral.tap`

You must choose *All Files* in the *Save As Type* drop-down or Notepad will append .TXT to your filename and Mach3 will not be able to find it.

```
g20 f100
g00 x1 y0 z0
g03 x1 y0 z-0.2 i-1 j0
g03 x1 y0 z-0.4 i-1 j0
g03 x1 y0 z-0.6 i-1 j0
g03 x1 y0 z-0.8 i-1 j0
g03 x1 y0 z-1.0 i-1 j0
g03 x1 y0 z-1.2 i-1 j0
m00
```

Again all the "0" are zeros in this. Don't forget to press the *Enter* key after the m00. Use the File>Load G-code menu to load this program. You will notice that it is displayed in the G-code window.

On the *Program Run* screen, you can try the effect of the *Start Cycle, Pause, Stop,* and *Rewind* buttons and their shortcuts.

As you run the program, you may notice that the highlighted line moves in a peculiar way in the G-code window. Mach3 reads ahead and plans its moves to avoid the tool path having to slow down more than in necessary. This look ahead is reflected in the display and when you pause.

You can go to any line of code scrolling the display so the line is highlighted. You can then use *Run from here.*

Note: You should always run your programs from a hard drive, not a floppy drive or USB "key". Mach3 needs high-speed access to the file, which it maps into memory. The program file must not be read-only.

3.8 Tool path display

3.8.1 Viewing the tool path

The Program Run screen has a blank square on it when Mach3 is first loaded. When the Spiral program is loaded you will see it change to a circle inside a square. You are looking straight down on the tool path for the programmed part, i.e. in Mach3Mill you are looking perpendicular to the X-Y plane.

The display is like a wire model of the path the tool will follow placed inside a clear sphere. By dragging the mouse over the window you can rotate the "sphere" and so see the model from different angles. The set of axes in the top left hand corner show you what directions are X, Y and Z. So if you drag the mouse from the centre in an upwards direction, the "sphere" will turn showing you the Z axis and you will be able to see that the circle is actually a spiral cut downwards (in the negative Z direction). Each of

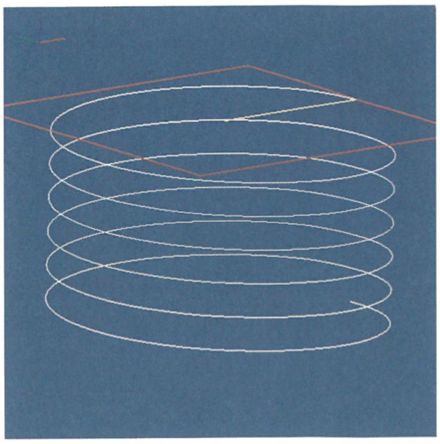


Figure 3.12 Toolpath from Spiral.txt

the G3 lines in the Spiral program above draws a circle while simultaneously lowering the tool 0.2 in the Z direction. You can also see the initial G00 move which is a straight line.

You can if you wish, produce a display like the conventional isometric view of the tool path.

A few minutes of "play" will soon give you confidence in what can be done. Your display may be a different colour to that shown in figure 3.12. The colours can be configured. See chapter 5.

3.8.2 Panning and Zooming the tool path display

The tool path display can be zoomed by dragging the cursor in its window with the Shift key depressed.

The tool path display can be panned in its window by dragging the cursor in the window with the Right mouse button held.

Double-clicking the tool path window restores the display to the original perpendicular view with no zoom applied.

Note: You cannot Pan or Zoom while the machine tool is running.

3.9 Other screen features

Finally it is worth browsing through some of the other Wizards and all the screens.

As a small challenge you might like to see if you can identify the following useful features:

- A button for estimating the time that a part program will take to run on the actual machine tool
- The controls for overriding the feed rate selected in the part program
- DROs which give the extent of movement of the tool in all axes for the loaded part program
- A screen that lets you set up information like where you want the Z axis to be put to make X and Y moves safe from hitting clamps etc.
- A screen that lets you monitor the logic levels (zero and one) on all Mach3s inputs and outputs.

3.10 The EStop control

The Sieg KX Series CNC milling machine has a big red Emergency Stop (EStop) button. It is fitted right at the front so that you can easily reach it from wherever you might be when you are operating the machine.

The EStop button will stop all activity in the machine as quickly as is safely possible. The spindle will stop rotating and the axes will stop moving. This will happen **without** relying on software. The circuit will tell Mach3 what you have done. It will generally **not** be good enough to turn off the AC power for an EStop because the energy stored in DC smoothing capacitors can allow motors to run on for some considerable time.

The machine will not be able to run again until the "reset" button has been pressed. If the EStop button locks when pushed then the machine will not start when you release it by turning its head.

It will not generally be possible to continue machining a part after an EStop but you and the machine will at least be safe.

3.11 The PC parallel port

3.11.1 The parallel port and its history

When IBM designed the original PC (160k floppy disc drive, 64kbytes of RAM!) they provided an interface for connecting printers using a 25 conductor cable. This is the foundation of the Parallel port we have on most PCs today. As it is a very simple way of transferring data, it has been used for many things other than connecting printers. You can transfer files between PC's, attach copy protection "dongles", connect peripherals like scanners and Zip drives and of course control machine tools using it. USB is taking over many of these functions and this conveniently leaves the parallel port free for Mach3.The connector on the PC is a 25 way female "D" connector.

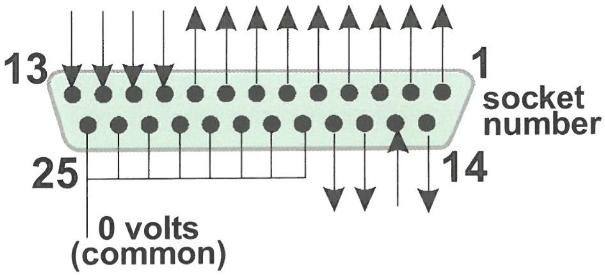


Figure 3.13 - Parallel port female connector
(seen from back of PC)

Its sockets seen from the back of the PC are shown in figure 3.13. The arrows give the direction of information flow relative to the PC. Thus, for example, pin 15 is an input to the PC.

Note: Convertors which plug into a USB port and have a 25 pin connector will not drive a machine even though they are perfectly suitable for the simpler task of connecting a printer.

3.11.2 Logic signals

All the signals output by Mach3 and input to it are binary digital (i.e. zeros and ones). These signals are voltages supplied by the output pins or supplied to the input pins of the parallel port. These voltages are measured relative to the computer's 0 volt line (which is connected to pins 18 to 25 of the port connector).

The first successful family (74xx series) of integrated circuits used TTL (transistor-transistor logic). In TTL circuits, any voltage between 0 and 0.8 volts is called "lo" and any voltage between 2.4 and 5 volts is called "hi". Connecting a negative voltage or anything above 5 volts to a TTL input will produce smoke. The parallel port was originally built using TTL and to this day these voltages define its "lo" and "hi" signals. Notice that in the worst case, there is only 1.6 volts difference between them.

It is, of course, arbitrary whether we say that a "lo" represents a logic one or a logic zero. However, as is explained below, "lo" = one is actually better in most practical interface circuits.

For an output signal to do anything, some current will have to flow in the circuit connected to it. When it is "hi" current will flow **out** of the computer. When it is "lo" current will flow **into** the computer. The more current you have flowing in, the harder it is to keep the voltage near zero so the nearer to the permitted limit of 0.8 volts "lo" will become. Similarly, current flowing out of a "hi" will make the voltage be lower and nearer to the 2.4 volts lower limit. So with **too** much current the difference between "lo" and "hi" will be even less than 1.6 volts and things will become unreliable. Finally, it's worth noting you are allowed roughly 20 times more current flowing into a "lo" than you are allowed flowing out of a "hi".

So this means that it is best to assign logic 1 to be a “lo” signal. Fairly obviously this is called **active lo** logic. The main practical **disadvantage** of it is that the device connected to the parallel port has to have a 5 volt supply to it. This is sometimes taken from the PC game port socket or from a power supply in the device that is connected.

Turning to input signals, the computer will need to be supplied with some current (less than 40 microamps) for “hi” inputs and will supply some (less than 0.4 milliamps) for “lo” inputs.

Because modern computer motherboards combine many functions, including the parallel port, into one chip we have experienced systems where the voltages only just obey the “hi” and “lo” rules. You might find that a machine tool that ran on an old system becomes temperamental when you upgrade the computer. Pins 2 to 9 are likely to have similar properties (they are the data pins when printing). Pin 1 is also vital in printing but the other output pins are little used and may be less powerful in a carefully “optimised” design.

3.12 Limit and Home switches

3.12.1 Reference signals

Limit switches are used to prevent any linear axis moving too far (over travel) and so causing damage to the structure of the machine. The Sieg KX Series milling machine has limit switches built in.

Over travel is the movement of the switch that occurs after it has operated. With a limit switch it can be caused by the inertia of the drive.

Home switches The Sieg KX Series milling machine also has home switches. Mach3 can be commanded to move one (or all) axes to the home position. This will need to be done whenever the system is switched on so that it knows where the axes are currently positioned.

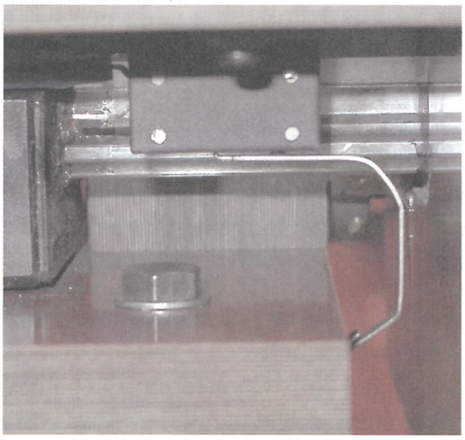


Figure 3.14 – A typical limit switch – a microswitch mounted on the table is tripped by bed of machine.

3.12.2 Referencing in action

When you request referencing (by button or G-code) the axis (or axes) which have home switches defined will travel (at a selectable low speed) in the defined direction until the home switch operates. The axis will then move back in the other direction so as to be off the switch. During referencing the limits do not apply.

When you have referenced an axis, then zero or some other value which is set up in the Config> State dialog, can be loaded into the axis DRO as its absolute machine coordinate. If you use zero then the home switch position is also the machine zero position of the axis. If the reference goes in the negative direction of an axis (usual for X and Y) then you might get referencing to load something like -0.5” into the DRO. This means that the home is half an inch clear of the limit. This wastes a bit of the axis travel but if you overshoot, when jogging to Home, you will not accidentally trip the limits. See also Software Limits as another way of solving this problem.

If you ask Mach3 to reference before you jog off the switch then it will travel in the opposite direction (because it says that you are already on the home switch) and stop when you get off the switch. This is fine when you are on the limit at the reference end of the axis. If, however, you are on the other Limit switch (and Mach3 cannot know this as they are shared) then the axis moves forever away from the actual home point until it crashes. So the advice is always jog carefully off the limit switches, then reference. It is possible to configure mach3 so it will not automatically jog off the home switch if you are concerned about this problem

4. Configuring Mach3 for the Sieg KX Series CNC milling machine

4.1 How the Profile information is stored

When the Mach3 Loader (Mach3.exe) program is run it will prompt you for the Profile file to use (figure 4.1). This will need to be in the Mach3 folder and has the extension .XML. You can view and print the contents of Profile files with Internet Explorer (as XML is a mark-up language used on web pages). The .XML file for the Sieg KX Series CNC milling machine is on the CD supplied with the machine.

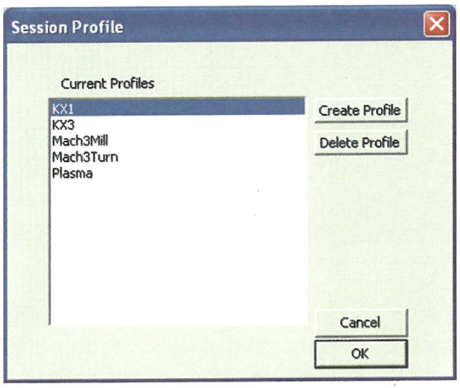


Figure 4.1 - Mach3 Profile window

Shortcuts are set up by the system installer to run Mach3.exe with default Profiles for a Mill and for Turning (i.e. Mach3Mill and Mach3Turn). You can create your own shortcuts, each with a different Profile. You can either run Mach3.exe and choose from the list of available profiles or you can set up extra shortcuts that specify the profile to use.

In a shortcut, the profile to load is given in the "/p" argument in the Target of the shortcut properties. As an example you should inspect the Properties of the Mach3Mill shortcut. This can be done, for example, by right clicking the shortcut and choosing Properties from the menu.

The .XML file for the Sieg KX Series milling machine profile **can** be edited by an external editor but you are **very strongly** advised not to do this unless you are fully conversant with the meaning of each entry in the files as some users have encountered very strange effects with miss-formatted files. Notice that some tags (e.g. the screen layout) are only created when a built-in default value is overridden using Mach3 menus. **It is much safer to use Mach3's configuration menus to update the XML profiles.**

5. Mach3 controls and running a part program

This chapter is intended for reference to explain the screen controls provided by Mach3 for setting up and running a job on the machine. It is of relevance to machine operators and for part-programmers who are going to prove their programs on Mach3.

5.1 Introduction

This chapter covers a lot of detail. You may wish to skim section 5.2 and then look at the sections for inputting and editing part programs before returning to the details of all the screen controls.

5.2 How the controls are explained in this chapter

Although at first sight you may feel daunted by the range of options and data displayed by Mach3, this is actually organised into a few logical groups. We refer to these as Families of Controls. By way of explanation of the term "control", this covers both buttons and their associated keyboard shortcuts used to operate Mach3 **and** the information displayed by DROs (digital read-outs), labels or LED's (light emitting diodes).

The elements of each control family are defined for reference in this chapter. The families are explained in order of importance for most users.

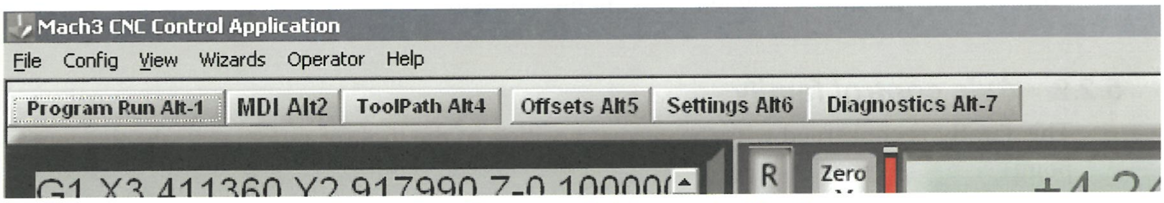


Figure 5.1 - Screen switching control family

You should, however, note that the actual screens of your **Mach3 do not include every control** of a family when the family is used. This may be to increase readability of a particular screen or to avoid accidental changes to the part being machined in a production environment

A Screen Designer is provided that allows controls to be removed or added from the screens or a set of screens. You can modify or design screens from scratch so that you can add any controls to a particular screen if your application requires this. For details see the *Mach3 Customisation* Wikipedia.

5.2.1 Screen switching controls

These controls appear on each screen. They allow switching between screens and also display information about the current state of the system.

5.2.1.1 Reset

This is a toggle. When the system is Reset the LED glows steadily, the charge pump pulse monitor (if enabled) will output pulses and the Enable outputs chosen will be active.

5.2.1.2 Labels

The "intelligent labels" display the last "error" message, the current modes, the file name of the currently loaded part program (if any) and the Profile that is in use.

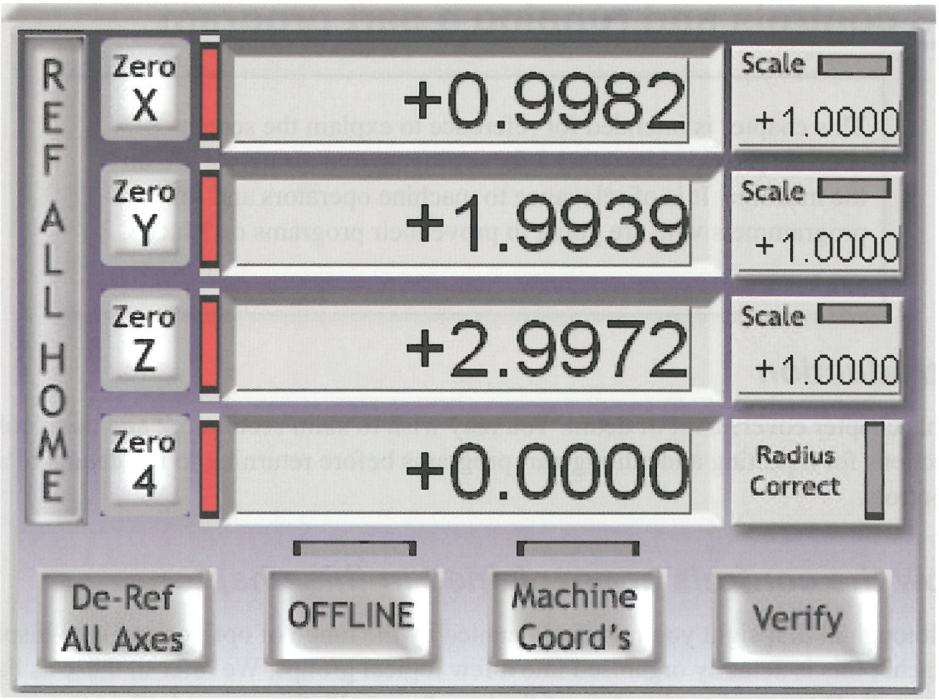


Figure 5.2 - Axis control family

5.2.1.3 Screen selection buttons

These buttons switch the display from screen to screen. The keyboard shortcuts are given after the names. For clarity in all cases when they are letters they are in upper-case. You should not, however, use the shift key when pressing the shortcut.

5.2.2 Axis control family

This family is concerned with the current position of the tool (or more precisely, the controlled point).

The axes have the following controls:

5.2.2.1 Coordinate value DRO

These are displayed in the current units (G20/G21) unless locked to the setup units on the Config>Logic dialog. The value is the coordinate of the controlled point in the displayed coordinate system. This will generally be the coordinate system of the current Work Offset (initially 1 - i.e. G54) together with any G92 offsets applied. It can however be switched to display Absolute Machine Coordinates.

You **can** type a new value into any Axis DRO. This will modify the current Work Offset to make the controlled point in the current coordinate system the value you have set. You are advised to set up Work Offsets using the Offsets screen until you are fully familiar with working with multiple coordinate systems.

5.2.2.2 Referenced

The LED is green if the axis has been referenced (i.e. is in a known actual position)

Each axis can be referenced using the *Ref All* button. Individual axes can be referenced on the *Diagnostics* screen

- If no home/reference switch is defined for the axis, then the axis will not actually be moved but, if Auto Zero DRO when homed is checked in Config>Referencing, then the absolute machine coordinate of the current position of the axis will be set to the value defined for the axis in the Home/Reference switch locations table in the Config>State dialog. This is most often zero.

- If there is a home/reference switch defined for the axis and it is not providing an active input when the Ref is requested, then the axis will be moved in the direction defined in Config>Referencing until the input does become active. It then backs off a short distance so that the input is inactive. If the input is already active then the axis just moves the same short distance into the inactive position. If Auto Zero DRO when homed is checked in Config>Referencing then the absolute machine coordinate of the current position of the axis will be set to the value defined for the axis in the Home/Reference switch locations table in the Config>State dialog.

The *De-Ref All* button does not move the axes but stops them being in the referenced state.

5.2.2.3 Machine coordinates

The MachineCoords button displays absolute machine coordinates. The LED warns that absolute coordinates are being displayed.

5.2.2.4 Scale

Scale factors for any axes can be set by G51 and can be cleared by G50. If a scale factor (other than 1.0) is set then it is applied to coordinates when they appear in G-code (e.g. as X words, Y words etc.) The Scale LED will flash as a reminder that a scale is set for an axis. The value defined by G51 will appear, and can be set, in the Scale DRO. Negative values mirror the coordinates about the relevant axis.

5.2.2.5 Soft limits

The *Softlimits* button enables the soft limits values defined in Config>Homing/Limits.

5.2.2.6 Verify

The *Verify* button will move to home switches to verify if any steps might have been lost during preceding machining operations.

5.2.2.7 Diameter/Radius correction

Rotary axes can have the approximate size of the workpiece defined using the Rotational Diameter control family. This size is used when making blended feed rate calculations for co-ordinated motion including rotational axes. The LED indicates that a non-zero value is defined.

5.2.3 "Move to" controls

There are many buttons on different screens designed to make it easy to move the tool (controlled point) to a particular location (e.g. for a tool change). These buttons include: *Goto Zs* to move all axes to zero, *Goto Tool Change*, *Goto Safe Z*, *Goto Home*.

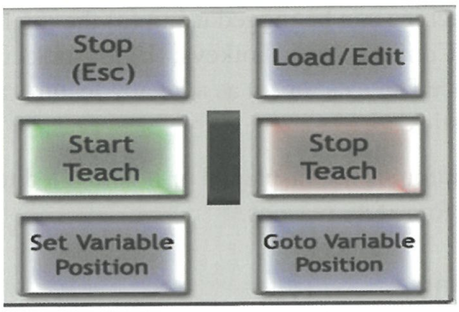


Figure 5.3 – Controlled point memories & Teach

In addition, Mach3 will remember two different sets of coordinates and go to them on demand. These are controlled by *Set Reference Point* and *Goto Ref Point*, and by *Set Variable Position* and *Goto Variable Position.*

5.2.4 MDI and Teach control family

G-code lines (blocks) can be entered, for immediate execution, into the MDI (Manual Data Input) line. This is selected by clicking in it or the MDI hotkey (Enter in the default configuration). When the MDI line is active its colour changes and a flyout box showing the recently entered commands is displayed. An example is shown in figure 5.4. The cursor up and down arrow keys can be used to select from the flyout so that you can reuse a line that you have already entered. The *Enter* key causes Mach3 to execute the current MDI line and it remains active for input of another set of commands. The *Esc* key clears the line and de-selects it. You need to remember that when it is selected, all keyboard input (and input from a keyboard emulator or custom keyboard) is written in the MDI line rather than controlling Mach3. In particular, jogging keys will not be recognised: you must *Esc* after entering MDI.

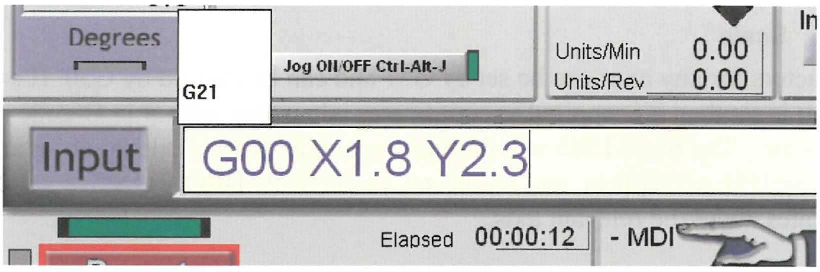


Figure 5.4 – MDI line

Mach3 can remember all the MDI lines as it executes them and store them in a file by using the Teach facility. Click *Start Teach*, enter the required commands and then click *Stop Teach*. The LED blinks to remind you that you are in Teach Mode. The commands are written in the file with the conventional name "C:/Mach3/GCode/MDITeach.tap" , clicking Load/Edit will load this file into Mach3 where it can be run or edited in the usual way – you need to go to the Program Run screen to see it. If you wish to keep a given set of taught commands then you should Edit the file and use *Save As* in the editor to give it your own name and put it in a convenient folder.

5.2.5 Jogging control family

Jogging controls are collected on a special screen which flies-out into use when the *Tab* key is pressed on the keyboard. It is hidden by a second press of *Tab*.

This is illustrated in figure 5.5.

Whenever the *Jog ON? OFF* button is displayed on the current screen then the axes of the machine can be jogged using (a) the jog hotkeys – the hotkeys are defined in Configure Axis hotkeys; (b) joysticks interfaced as USB Human Interface Devices; or (c) as a legacy feature, a Windows compatible analog joystick.

If the *Jog ON/ OFF* button is not displayed or it is toggled to OFF then jogging is not allowed for safety reasons.

5.2.5.1 Hotkey jogging

There are three modes. Continuous, Step and MPG which are selected by the *Jog Mode* button and indicated by the LEDs.

Figure 5.5 - Jogging control family

Continuous mode moves the axis or axes at the defined slow jog rate while the hotkeys are depressed

The jogging speed used with hotkeys in Continuous mode is set as a percentage of the rapid traverse rate by the *Slow Jog Percentage* DRO. This can be set (in the range 0.1% to 100%) by typing into the DRO. It can be nudged in 5% increments by the buttons or their hotkeys.

This *Slow Jog Percentage* can be overridden by depressing *Shift* with the hotkey(s). An LED beside the Cont. LED indicates this full speed jogging is selected.

Step mode moves the axis by one increment (as defined by the *Jog Increment* DRO) for each key press. The **current** feed rate (as defined by the F word) is used for these moves. The size of increment can be set by typing it into the *Step* DRO or values can be set in this DRO by cycling through a set of 10 user definable values using the *Cycle Jog Step* button.

Incremental mode is selected by the toggle button or, if in Continuous Mode, temporarily selected by holding down *Ctrl* before performing the jog.

5.2.5.2 Parallel port or Modbus MPG jogging

Up to three quadrature encoders connected to the parallel ports or ModBus can be configured as MPGs for jogging by using the *Jog Mode* button to select MPG *Jog Mode.*

The axis that the MPG will jog is indicated by the LEDs and the installed axes are cycled through by the *Alt-A* button for MPG1, *Alt-B* for MPG2 and *Alt-C* for MPG3.

Over the graphic of the MPG handle are a set of buttons for selecting the MPG mode.

In *MPG Velocity* Mode the velocity of the axis movement is related to the rotational speed of the MPG with Mach3 ensuring that the acceleration of the axis and top speed if honoured. This gives a very natural feel to axis movement. *MPG Step/Velocity* mode currently works like velocity mode.

In *Single Step* mode, each "click" from the MPG encoder requests one incremental jog step (with the distance set as for hot key Step jogging). Only one request at a time will be allowed. In other words if the axis is already moving then a "click" will be ignored. In *Multi-step* mode, clicks will be counted and queued for action. Note that this means that for large steps, rapid movement of the wheel may mean that the axis moves a considerable distance and for some time after the wheel movement has stopped. The steps are implemented with the federate given by the *MPG Feedrate* DRO

These step modes are of particular use in making very fine controlled movements when setting up work on a machine. You are advised to start using Velocity Mode.

5.2.5.3 Spindle Speed control family

The machine spindle is switched on and off by M-codes. The speed is set by Mach3.

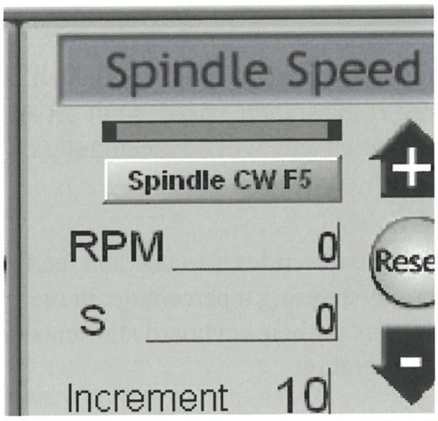


Figure 5.6 - Spindle speed control family

The *S* DRO has its value set when an S word is used in a part program. It is the desired spindle speed. It can also be set by typing into the DRO.

Mach3 will not allow you to try to set to a speed less than that set in *Min Speed* or greater than that set in *Max Speed.*

5.2.6 Feed control family

5.2.6.1 Feed Units per minute

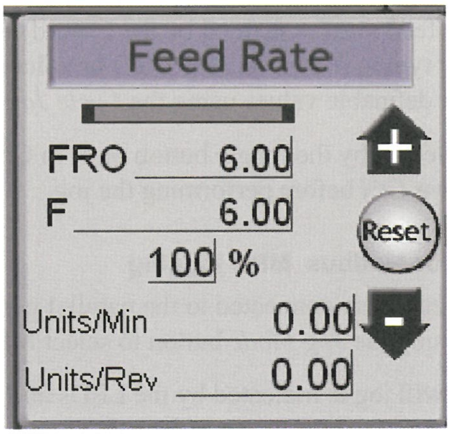


Figure 5.7 Feed control family

The *Prog Feed* DRO gives the feed rate in current units (inches/millimetres per minute). It is set by the F word in a part program or by typing into the *F* DRO. Mach3 will aim to use this speed as the actual rate of the co-ordinated movement of the tool through the material. If this rate is not possible because of the maximum permitted speed of any axis then the actual feed rate will be the highest achievable.

5.2.6.2 Feed Units per rev.

As modern cutters are often specified by the permitted cut per "tip" it may be convenient to specify the feed per revolution (i.e. feed per tip x number of tips on tool). The *Prog Feed* DRO gives the feed rate in current units (inches/millimetres) per rev of the spindle. It is set by the F word in a part program or by typing into the DRO.

To employ Feed units/rev, Mach3 must know the value of the chosen measure of the speed of the spindle (i.e. it must have been (a) defined in an S word or by data entered to *S* DRO in the Spindle speed control family.

Notice that the numeric values in the control will be very different unless spindle speed is near to 1 rpm! So using a feed per minute figure with feed per rev mode will probably produce a disastrous crash.

5.2.6.3 Feed display

The actual feed in operation allowing for the co-ordinated motion of all axes is displayed in *Units/min* and *Units/rev.* If the spindle speed is not set and the actual spindle speed is not measured then the *Feed per rev* value will be meaningless.

5.2.6.4 Feed override

Unless M49 (Disable feed rate override) is in use, the feed rate can be manually overridden, in the range 20% to 299%, by entering a percentage in the DRO. This value can be nudged (in steps of 10%) with the buttons or their keyboard shortcuts and be reset to 100%. The LED warns of an override is in operation.

The *FRO* DRO displays the calculated result of applying the percentage override to the set feed rate.

5.2.7 Program Running control family

These controls handle the execution of a loaded part program or the commands on an MDI line.

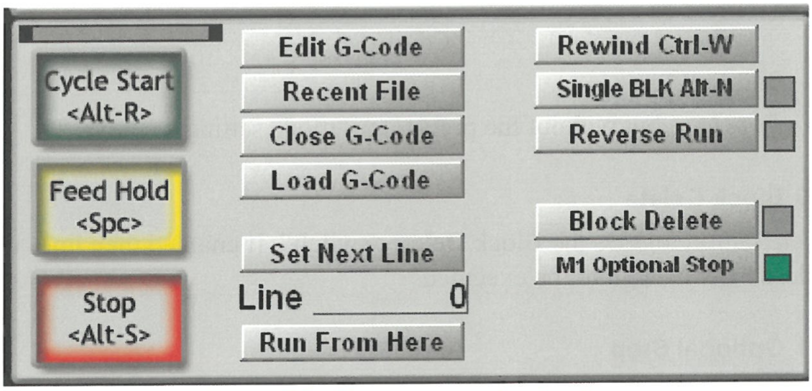


Figure 5.8 - Program running family

5.2.7.1 Cycle Start

Safety warning: Note that the *Cycle Start* button will, in general, start the spindle and axis movement. It should always be configured to require "two hand" operation and if you are assigning your own hotkeys, it should not be a single keystroke.

5.2.7.2 Feed Hold

The *Feedhold* button will stop the execution of the part program as quickly as possible but in a controlled way so it can be restarted by Cycle Start. The spindle will remain on but can be stopped manually if required.

When in Feed Hold you can jog the axes, replace a broken tool etc. If you have stopped the spindle then you will generally want to turn them on before continuing. Mach3 will however, remember the axis positions at the time of the Feed Hold and return to them before continuing the part program

5.2.7.3 Stop

Stop halts axis motion as quickly as possible. It may result in lost steps (especially on stepper motor driven axes) and restarting may not be valid.

5.2.7.4 Rewind

Rewinds the currently loaded part program.

5.2.7.5 Single BLK

SingleBLK is a toggle (with indicator LED). In Single Block mode, a Cycle Start will execute the next single line of the part program and then enter Feed Hold.

5.2.7.6 Reverse Run

Reverse Run is a toggle (with indicator LED). It should be used after a Feed Hold or Single Block and the next Cycle Start will cause the part program to run in reverse. This is particularly useful in recovering from a lost arc condition in plasma cutting or a broken tool.

5.2.7.7 Line Number

Line DRO is the ordinal number of the current line in the G-code display window (starting from 0). Note that this is not related to the "N word" line number.

You can type into this DRO to set the current line.

5.2.7.8 Run from here

Run from here performs a dummy run of the part program to establish what the modal state (G20/G21, G90/G91 etc.) should be and then prompts for a move to put the controlled point in the correct position for the start of the line in *Line Number*. You should not attempt to *Run from here* in the middle of a subroutine.

5.2.7.9 Set next line

Like *Run from here* but without the preparatory mode setting or move.

5.2.7.10 Block Delete

The *Delete* button toggles the Block Delete "switch". If enabled then lines of G-code which start with a slash - i.e. / - will not be executed.

5.2.7.11 Optional Stop

The *End* button toggles the Optional Stop "switch". If enabled then the M01 command will be treated as M00.

5.2.8 File control family

These controls, figure 5.8, are involved with the file of your part program. They should be self-evident in operation.

5.2.9 Tool details

In the Tool Details group, figure 5.9, controls display the current tool, the offsets for its length and diameter and, on systems with a Digities input, allow it to be automatically zero to the Z plane.

Figure 5.9 – Tool Details

Unless tool change requests are being ignored (Config>Logic), on encountering an M6 Mach3 will move to Safe Z and stop, flashing the *Tool Change* LED. You continue (after changing the tool) by clicking *Cycle Start*.

The elapsed time for the current job is displayed in hours, minutes and seconds.

5.2.10 G-Code and Tool path control family

The currently loaded part program is displayed in the G-code window. The current line is highlighted and can be moved using the scroll bar on the window.

The Tool path display, figure 5.10, shows the path that the controlled point will follow in the X, Y, Z planes. When a part program is executing the path is over painted in the colour selected in Config>Tool path. This over painting is dynamic and is not preserved when you change screens or indeed alter views of the tool path.

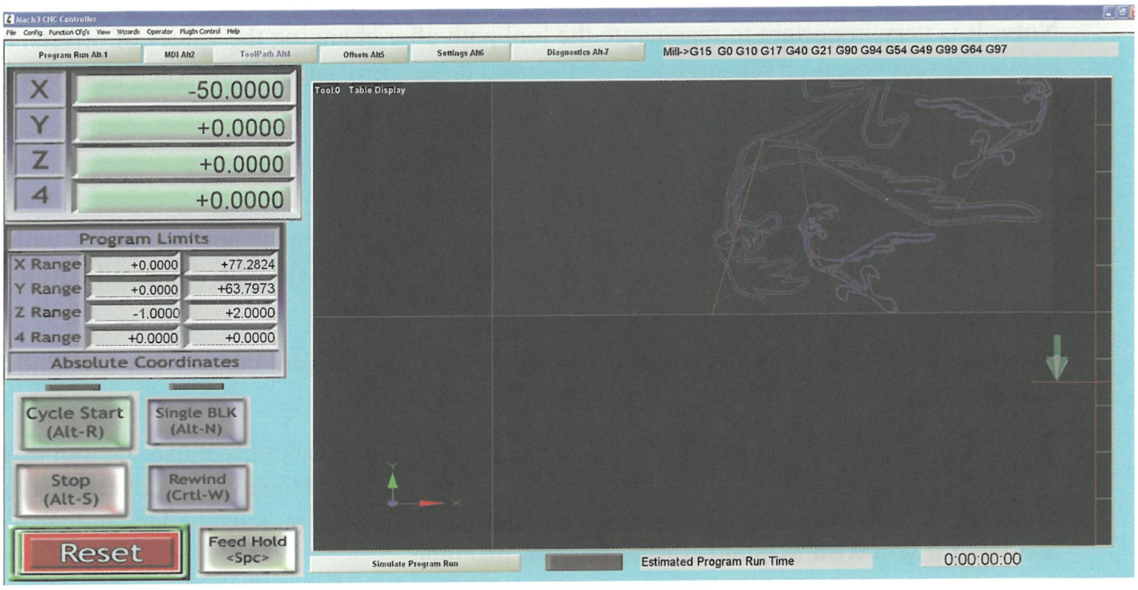


Figure 5.10 - Toolpath family

On occasions you will find that the display does not exactly follow the planned path. It occurs for the following reason. Mach3 prioritises the tasks it is doing. Sending accurate step pulses to the machine tool is the first priority. Drawing the tool path is a lower priority. Mach3 will draw points on the tool path display whenever it has spare time and it joins these points by straight lines. So, if time is short, only a few points will be drawn and circles will tend to appear as polygons where the straight sides are very noticeable. This is nothing to worry about.

The *Simulate Program Run* button will execute the G-code, but without any tool movement, and allow the time to make the part to be estimated.

The *Program Limits* data allow you to check the maximum excursion of the controlled point to be reasonable (e.g. not milling the top off the table).

The screenshot also shows axis DROs and some Program Run controls.

It is often useful to use the *Display Mode* button to toggle from Job to Table mode to show the tool path in relation to the table. See figure 5.11

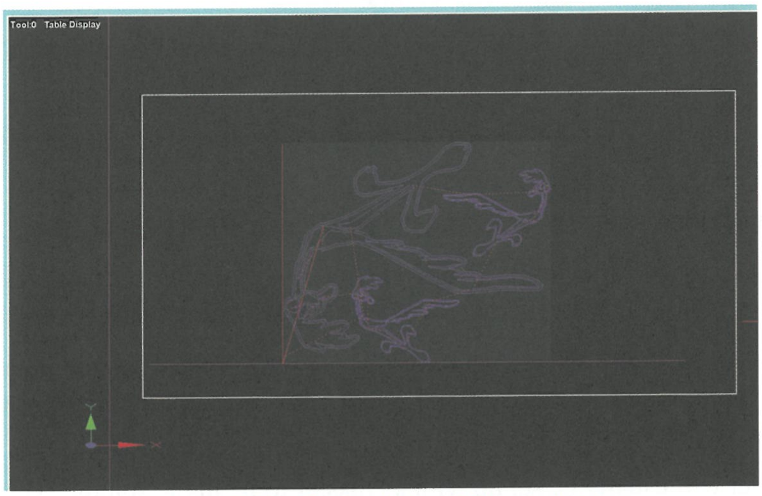


Figure 5.11 – Toolpath in relation to table

The tool path display can be rotated by left clicking and dragging the mouse in it. It can be zoomed by shift-left clicking and dragging and can be panned by dragging a right click.

The *Regenerate* button will regenerate the tool path display from the G-code with the currently enabled fixture and G92 offsets.

Note: It is very important to regenerate the tool path after changing the values of offsets, both to get the correct visual effect and because it is used to perform calculations when using G42 and G43 for cutter compensation.

5.2.11 Work offset and tool table control family

Work Offset and Tool tables can be accessed from the Operator menu and, of course, within a part program but it is often most convenient to manipulate them through this family. Refer to chapter 6 for details of the tables and techniques like "Touching".

Because of the underlying G-code definitions Work Offset and Tool tables work in slightly different ways.

Warning: Changing the Work and Tool offsets in use will never actually move the tool on the machine although it will of course alter the axis DRO readings. However, a move, G0, G1 etc. **after** setting new offsets will be in the new coordinate system. You must understand what you are doing if you wish to avoid crashes on your machine.

5.2.11.1 Work Offsets

Mach3 by default uses Work Offset number 1. Choosing any value from 1 to 255, and entering it in the *Current Work Offset* DRO, will make that Work Offset current. Work offsets are sometimes called Fixture Offsets.

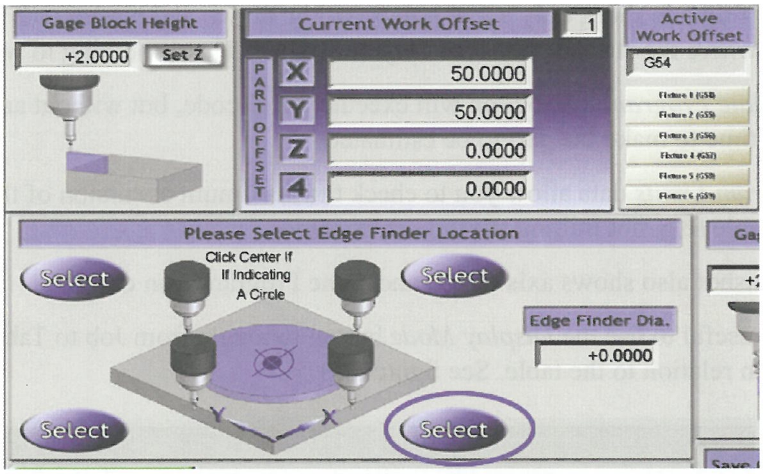


Figure 5.12 – Work offsets family

Typing into the DRO is equivalent to a part program issuing G55 to 59 or G58.1 to G59.253 (q.v.).

You can also set the current offset system using the *Fixture* buttons.

You can change the value of the offset values for the current offset system by typing into the relevant *Part Offset* DROs. (Part Offset is yet another name for Work and Fixture offsets!)

Values can also be set in these DROs by moving the axes to a desired place and clicking *as Set or Select button.* The X and Y axes and Z axis are set in slightly different ways. Z is easier to understand so we will describe it first.

The Z offset will usually be set up with a "master tool" in the spindle. The Z for other tools will then be corrected by the tool table. A gauge block or sometimes even a piece of foil or paper is slid between the tool and the top of the work (if this is to be Z = 0.0) or the table (if this is to be Z = 0.0). The Z axis is very gently jogged down until the gauge is just trapped by the tool. The thickness of the gauge is entered into the *Gauge Block Height* DRO and the *Set Z button* is clicked. This will set up the Z value of the current work offset so that the tool is at the given height.

The process for X and Y is similar except the touching might be done on any of four sides of the part and account has to be taken of the diameter of the tool (or probe) and the thickness of any gauge being used to give "feel" to the touching process.

For example to set the bottom edge of a piece of material to be Y = 0.0 with a tool of diameter 0.5" and a 0.1" gauge block, you would enter 0.7 in the *Edge Finder Dia* DRO (i.e. the diameter of the tool plus twice the gauge) and click the Select button that is ringed in figure 5.12.

Depending on your configuration of Persistent Offsets and Offsets Save in Config>State the new values will be remembered from one run of Mach3 to another.

5.2.11.2 Tools

Tools are numbered from 0 to 255. The tool number is selected by the T word in a part program or entering the number in the *T* DRO. Its offsets are only applied if they are switched on by the *Tool Offset On/Off* toggle button (or the equivalent G43 and G49 in the part program)

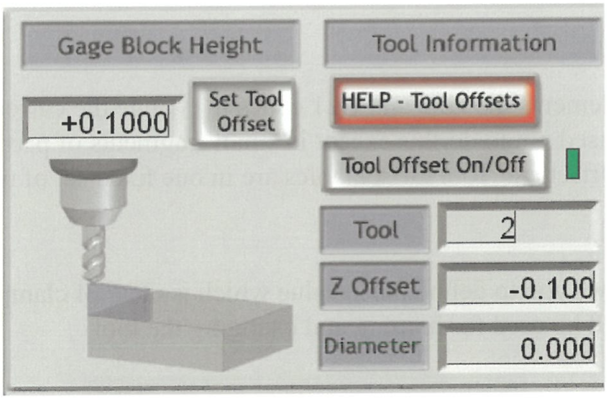


Figure 5.13 – Tool Offset

In Mach3Mill only the Z *offset* and *Diameter* are used for tools. The diameter can be entered in the DRO and the Z-offset (i.e. compensation for tool length) be entered directly or by touching. The Set Tool Offset feature works exactly as set Z with Work Offsets.

Tool Offset data is made persistent between runs in the same way as Work Offset data.

5.2.11.3 Direct access to Offset Tables

The tables can be opened and edited directly using the *Save Work Offsets* and *Save Tool Offsets* buttons or the Operator>Fixtures (i.e. Work Offsets) and Operator>Tooltable menus.

5.2.12 Rotational Diameter control family

As described in the Feed rate control family, it is possible to define the approximate size of a rotated workpiece so the rotational axis speed can be correctly included in the blended feed rate. The relevant diameters are entered in the DROs of this family.

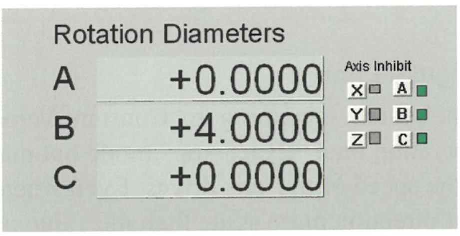


Figure 5.14 - Rotational diameters

The Axis control Family has warning LED(s) to indicate the setting of non-zero values here.

Values are not required if rotary movement is not to be coordinated with linear axes. In this case a suitable F word for degrees per minute or degrees per rev should be programmed.

5.2.13 Limits and miscellaneous control family

5.2.13.1 Override limits

Mach3 can use software to override limit switches connected to its inputs.

This can be automatic, i.e. the jogging performed immediately after a reset will not be subject to limits until the axis is jogged off the limit switches. The Toggle button and warning LED for Auto Limit Override controls this.

As an alternative, limits may be locked out using the Override Limits toggle. Its use is indicated by the LED.

5.2.14 System Settings control family

Note: The controls in this family are not in one place on the screens released with Mach3. You will need to hunt for them on Program Run, Settings and Diagnostics screens.

5.2.14.1 Units

This toggle implements the G20 and G21 codes to change the current measurement units. You are strongly advised not to do this except in small fragments of part program on account of the fact that Work Offset and Tool Offset tables are in one fixed set of units.

5.2.14.2 Safe Z

This family allows you to define the Z value which is clear of clamps and parts of the workpiece. It will be used for homing and changing the tool.

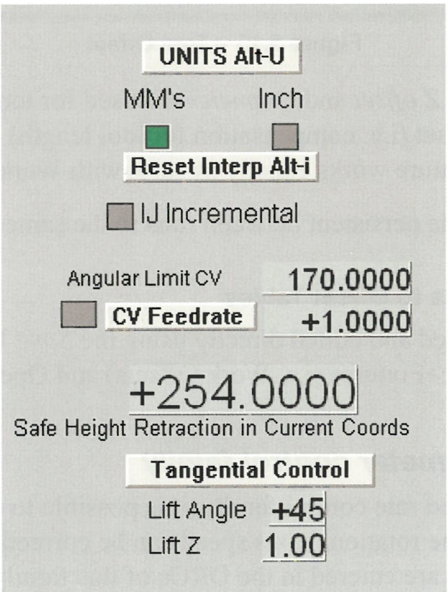


Figure 5.15 – System Settings, Safe Z controls etc.

5.2.14.3 CV Mode/Angular Limit

This LED is lit when the system is running in "Constant Velocity" mode. This will give smoother and faster operation than "Exact stop" mode but may cause some rounding at sharp corners depending on the speed of the axis drives. Even when the system is in CV mode a corner with a change of direction more acute than the value given in the *Angular Limit* DRO will be performed as if Exact Stop was selected. Full details of this are given under *Constant Velocity* in chapter 9.

5.2.14.4 Offline

This toggle and warning LED "disconnects" all the output signals of Mach3. This is intended for machine setup and testing. Its use during a part program will cause you all sorts of positioning problems.

5.2.15 Encoder control family

This family displays the values from the axis encoders and allows them to be transferred to and from the main axis DROs

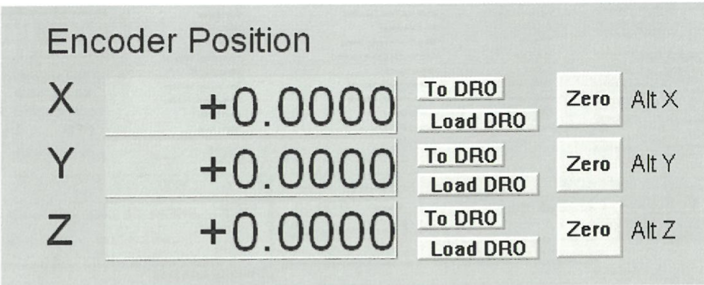


Figure 5.16 - Encoder control family

The *Zero* button will reset the corresponding encoder DRO to zero.

The *To DRO* button copies the value into the main axis DRO (i.e. applies this value as a G92 offset).

The *Load DRO* button loads the encoder DRO from the corresponding main axis DRO.

5.2.16 Automatic Z control family

Mach3 has the facility to set a lower limit for moves in the Z axis. See Config>Logic dialog for the static setting of this Inhibit-Z value.

There is also a control family which allows this *Inhibit Z* value to be set while preparing and before running a G-code program. This is shown in figure 5.17.

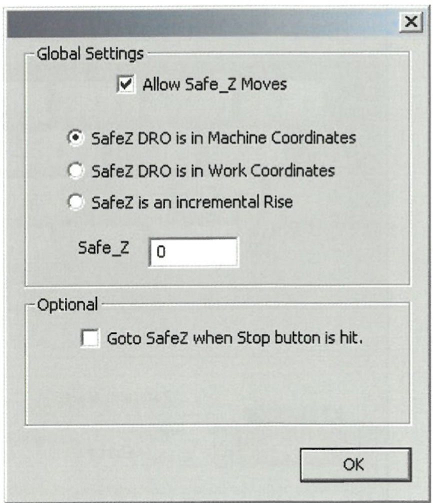


Figure 5.17 – Automatic Z control

Code the program, which might often be a DXF or HPGL import, so that it makes a single cut or set of cuts at the finally desired Z depth (perhaps Z = -0.6 inch assuming top of workpiece is Z = 0). The last command should be an M30 (Rewind)

Using the Automatic Z Control controls (a) set the *Z-inhibit* value to the Z for depth for the first roughing cut (perhaps Z= -0.05) (b) the *Lower Z-Inhibit* to the successive cut depths (we might allow 0.1 as the tool has some side support). The whole job will need seven passes to get to Z = -0.6, so (c) enter 7 in *L* (Loop). On pressing Cycle Start, the machine will automatically make the series of cuts at increasing Z depth. The DROs track the progress decrementing *L* as they are performed and updating the Z-inhibit value. If the given number of *L* does not reach the part program's requested Z depth then you can update the *L* DRO and restart the program.

5.3 Using Wizards

Mach3 Wizards are an extension to the Teach facility which allows you to define some machining operations using one or more special screens. The Wizard will then generate G-code to make the required cuts. Examples of Wizards include machining a circular pocket, drilling an array of holes and engraving text.

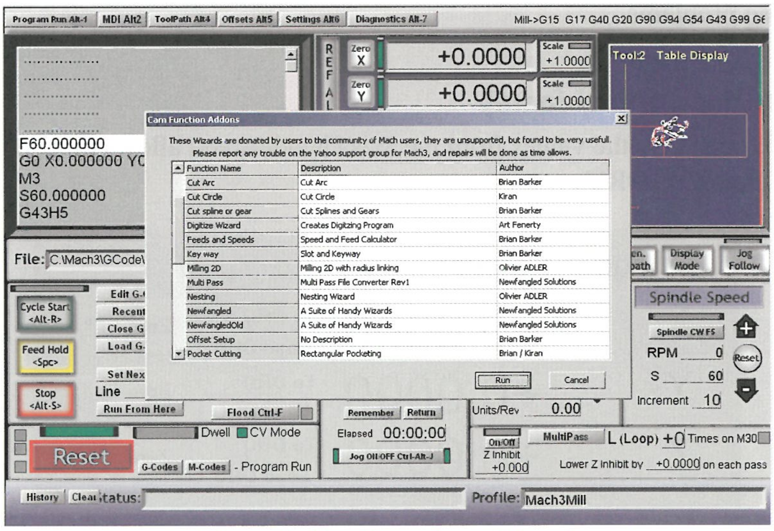


Figure 5.18 – Choosing a Wizard

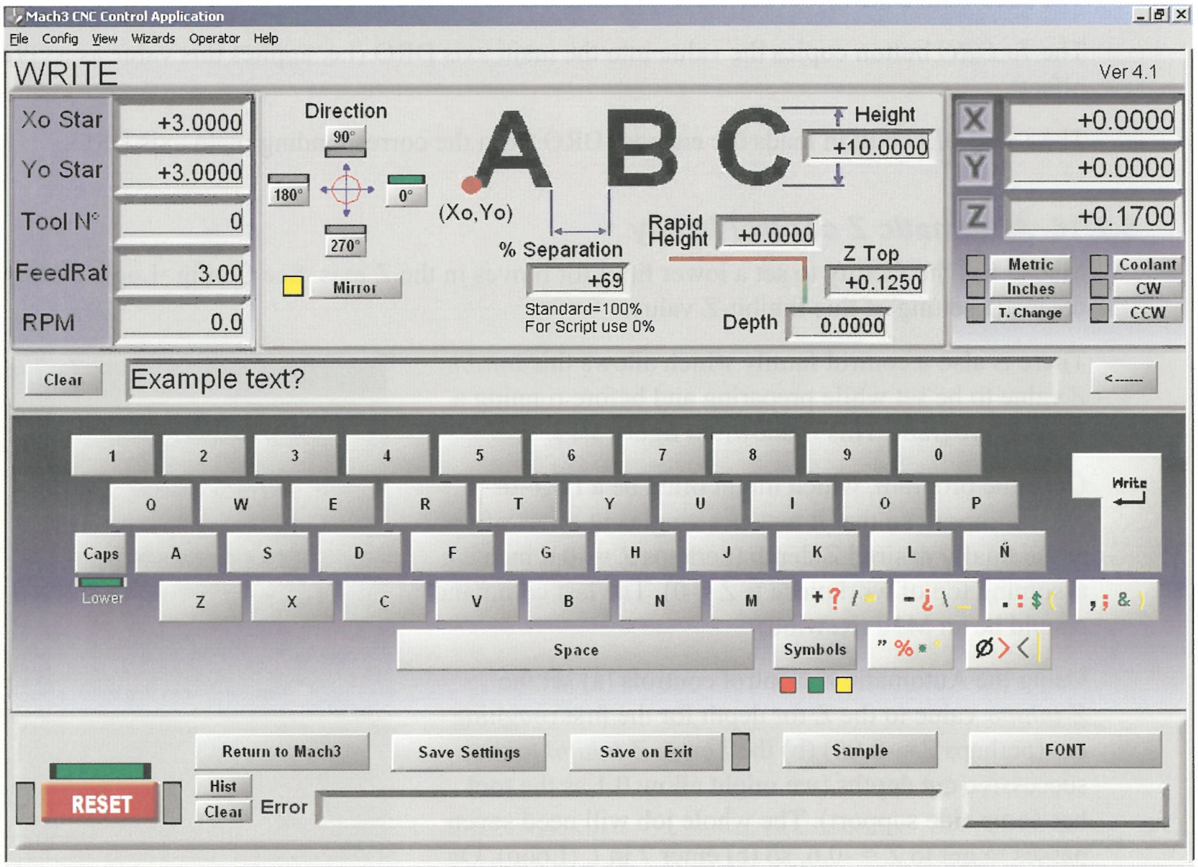


Figure 5.19 – The Write Wizard screen

The *Load Wizards* button displays a table of Wizards installed on your system. You choose the one required and click *Run*. The Wizard screen (or sometimes one of several screens) will be displayed. Chapter 3 includes an example for milling a pocket. Figure 5.19 is the Wizard for engraving text.

Wizards have been contributed by several authors and depending on their purpose, there are slight differences in the control buttons. Each Wizard will however have a means of posting the G-code to Mach3 (marked Write in figure 5.19) and a means of returning to the main Mach3 screens. Most Wizards allow you to save your settings so that running the Wizard again gives the same initial values for the DROs etc.

Figure 5.20 shows a section of the Tool path screen after the *Write* button is pressed on figure 5.19.

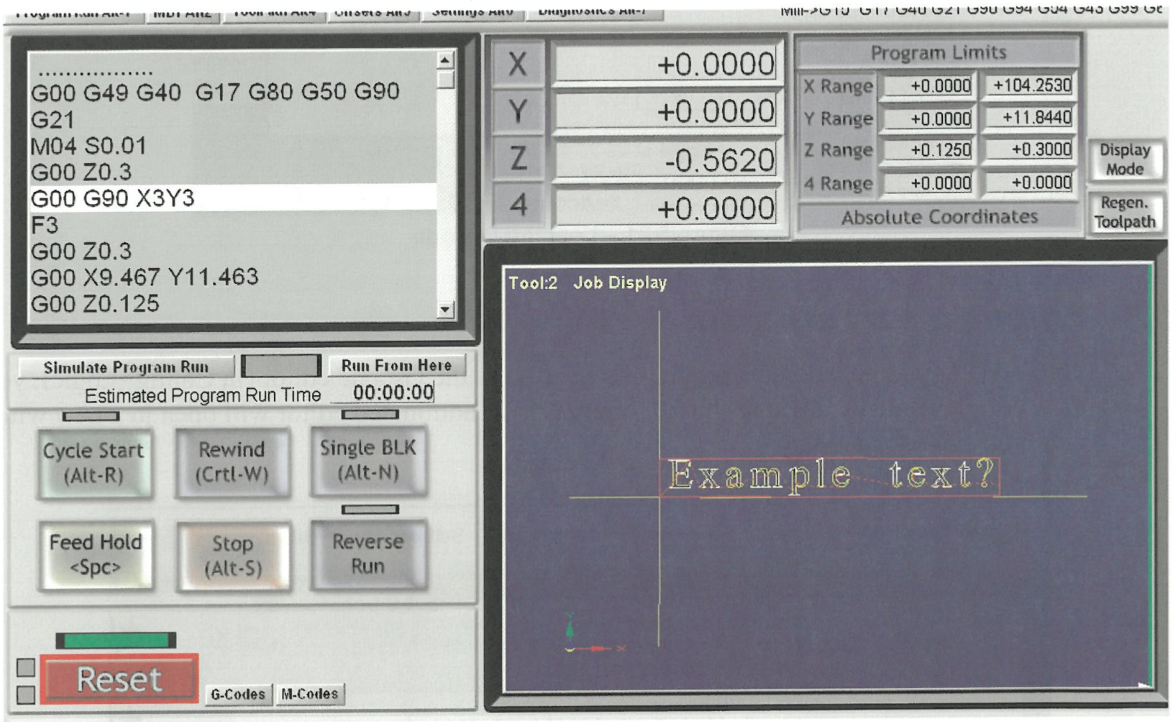


Figure 5.20 – After running the Write wizard

The *Last Wizard* buttons runs the wizard you most recently used without the trouble of selecting it from the list.

The *Conversational* button runs a set of wizards designed by Newfangled Solutions. These are supplied with Mach3 but require a separate license for them to be used to generate code.

5.4 Loading a G-code part program

If you have an existing part program which was written either by hand or a CAD/CAM package, then you load it into Mach3 using the *Load G-Code* button. You choose the file from a standard Windows file open dialog. Alternatively you can choose from a list of recently used files which is displayed by the *Recent Files* screen button.

When the file is chosen, Mach3 will load and analyse the code. This will generate a tool path for it, which will be displayed, and will establish the program extremes.

The loaded program code will be displayed in the G-code list window. You can scroll through this moving the highlighted current line using the scroll bar.

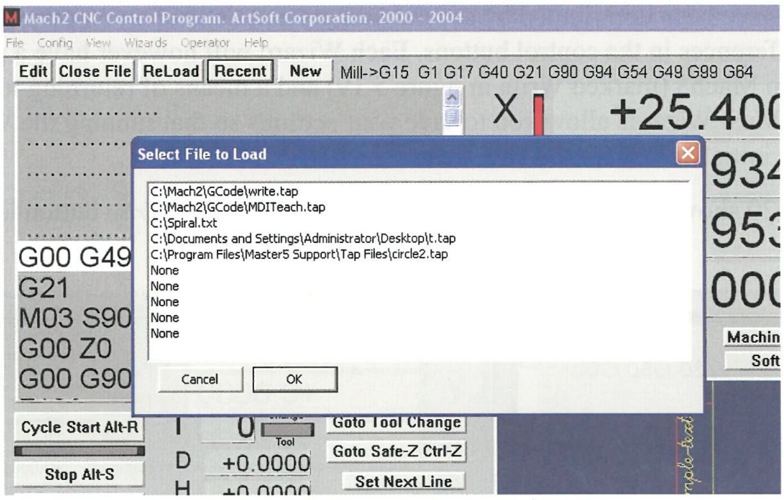


Figure 5.21 – Loading G-Code

5.5 Editing a part program

Provided you have defined a program to be used as the G-code editor (in Config>Logic), you can edit the code by clicking the *Edit* button. Your nominated editor will open in a new window with the code loaded into it.

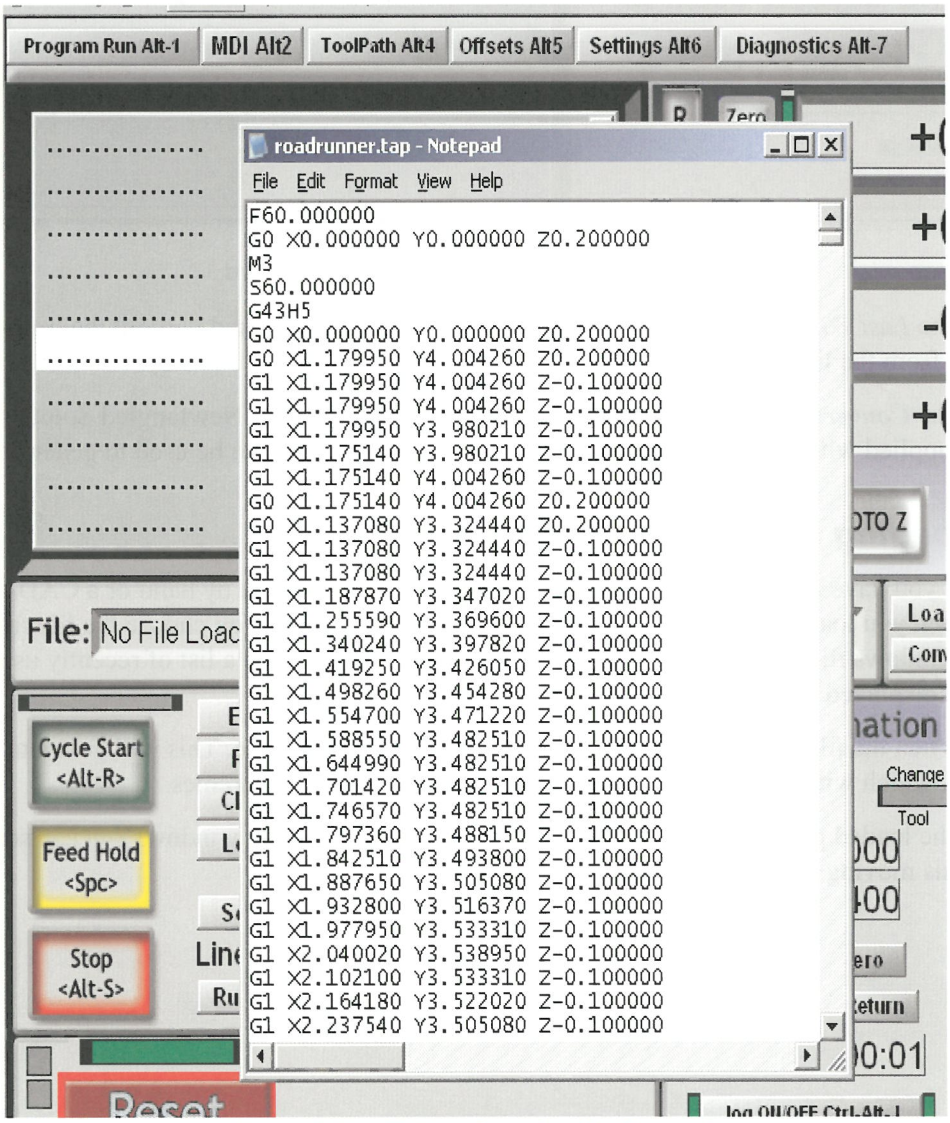


Figure 5.22 – Editing G-code with Notepad

When you have finished editing you should save the file and exit the editor. This is probably most easily done by using the close box and replying *Yes* to the "Do you want to save the changes?" dialog.

While editing, Mach3 is suspended. If you click in its window it will appear to be locked up. You can easily recover by returning to the editor and closing it.

After editing, the revised code will again be analysed and used to regenerate the tool path and extremes. You can regenerate the tool path at any time using the *Regenerate* button.

5.6 Manual preparation and running a part program

5.6.1 Inputting a hand-written program

If you want to write a program "from scratch" then you can either do so by running the editor outside Mach3 and saving the file or you can use the *Edit* button with no part program loaded. In this case you will have to *Save As* the completed file and exit the editor.

In both cases you will have to use File>Load G-code to load your new program into Mach3.

Warning: Errors in lines of code are generally ignored. You should not rely on being given a detailed syntax check.

5.6.2 Before you run a part program

It is good practice for a part program to make no assumptions about the state of the machine when it starts. It should therefore include G17/G18/G19, G20/G21, G40, G49, G61/G62, G90/G91, and G93/G94.

You should ensure that the axes are in a known reference position - probably by using the *Ref All* button.

You need to decide whether the program starts with an S word or if you need to set the spindle speed by entering a value in the *S* DRO.

You will need to ensure that a suitable feed rate is set before any G01/G02/G03 commands are executed. This may be done by an F word or entering data into the *F* DRO.

Next you may need to select a Tool and/or Work Offset.

Finally, unless the program has been proved to be valid, you should attempt a dry run, cutting "air" to see that nothing terrible happens.

5.6.3 Running your program

You should monitor the first run of any program with great care. You may find that you need to override the feed rate or, perhaps, spindle speed to minimise chattering or to optimise production. When you want to make changes you should do this on the "fly" or use the *Pause* button, make your changes and then click *Cycle Start*.

5.7 Building G-code by importing other files

Mach3 will convert files in DXF, HPGL or JPEG format into G-code which will cut a representation of them.

This is done using the File>Import HPGL/BMP/JPG or the File>Import>DXF menu. Having chosen a file type, you have to load the original file. You are prompted for parameters to define the conversion and feed commands to be included in the part program. You import the data. Mach3 has to create a .TAP working file which contains the generated G-code, so you will be prompted by a file save dialog for a name and folder for this.

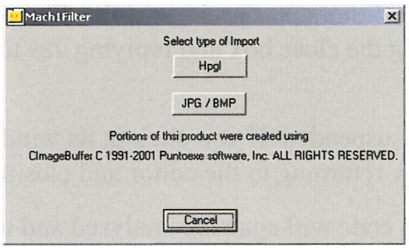


Figure 5.23 Choosing import filter

The .TAP file is then loaded into Mach3 and you can run it as with any other part program.

Full details of the conversion processes and their parameters are given in chapter 7.

6. Coordinate systems, tool table and fixtures

This chapter explains how Mach3 works out where exactly you mean when you ask the tool to move to a given position. It describes the idea of a coordinate system, defines the Machine Coordinate System and shows how you can specify the lengths of each Tool, the position of a workpiece in a Fixture and, if you need to, to add your own variable Offsets.
You may find it heavy going on the first read. We suggest that you try out the techniques using the Sieg KX Series milling machine. It is not easy to do this just "desk" running Mach3 as you need to see where an actual tool is and you will need to understand simple G-code commands like G00 and G01.
Mach3 can be used without a detailed understanding of this chapter but you will find that using its concepts makes setting up jobs on your machine is very much quicker and more reliable.

6.1 Machine coordinate system

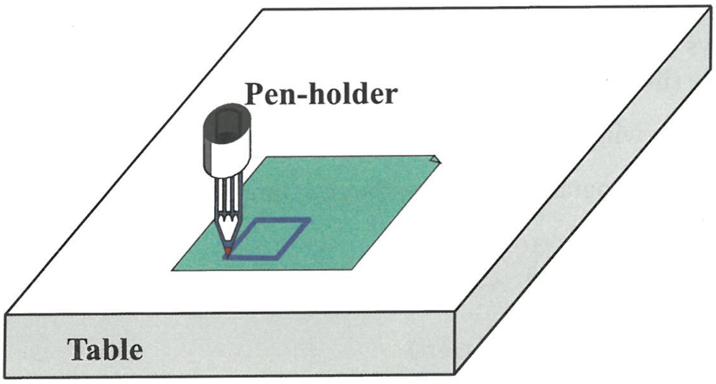


Figure 6.1 - Basic Drawing Machine

You have seen that most Mach3 screens have DROs labelled "X Axis", "Y Axis" etc. If you are going to make parts accurately and minimise the chance of your tool crashing into anything, you need to understand exactly what these values mean at all times when you are setting up a job or running a part program.

This is easiest to explain looking at a machine. We have chosen an imaginary machine that makes it easier to visualise how the coordinate system works. Figure 6.1 shows what it is like.

It is a machine for producing drawings with a ballpoint or felt tipped pen on paper or cardboard. It consists of a fixed table and a cylindrical pen-holder which can move left and right (X direction), front and back (Y direction) and up and down (Z-direction). The figure shows a square which has just been drawn on the paper.

Figure 6.2 shows the Machine Coordinate System which measures (let's say in inches) from the surface of the table at its bottom left hand corner. As you will see, the bottom left corner of the paper is at X=2, Y=1 and Z=0 (neglecting paper thickness). The point of the pen is at X=3, Y=2 and it looks as though Z=1.3.

If the point of the pen was at the corner of the table then, on this machine, it would be in its **Home** or referenced position. This position is often defined by the position of Home switches that the machine moves to when it is switched on. At any event, there will be a zero position for each axis called the **absolute machine zero**. We will come back to where Home might actually be put on a real machine.

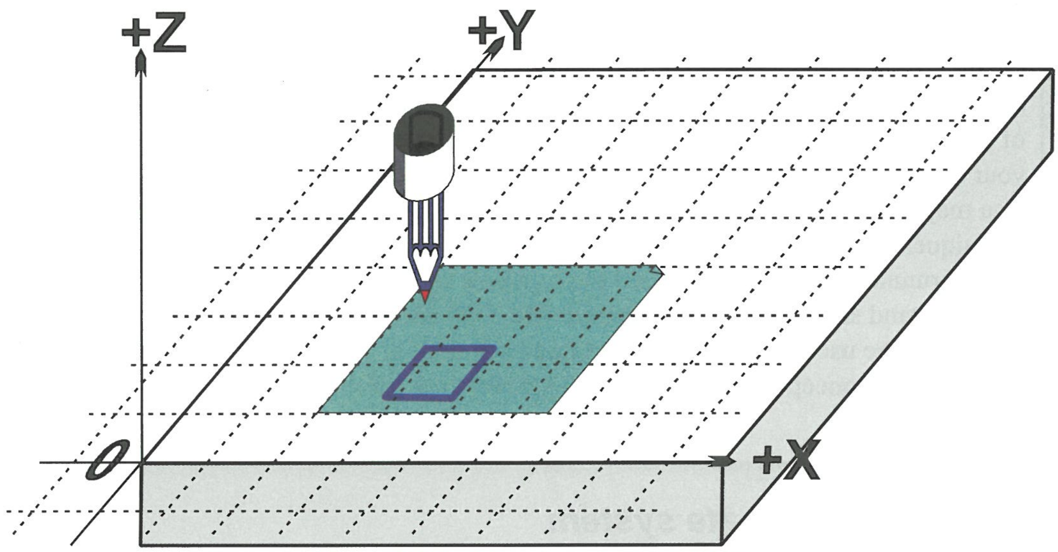


Figure 6.2 Machine coordinate system

The point of the pen, like the end of a cutting tool, is where things happen and is called the Controlled Point. The Axis DROs in Mach3 always display the coordinates of the Controlled Point relative to some coordinate system. The reason you are having to read this chapter is that it is not always convenient to have the zeros of the measuring coordinate system at a fixed place of the machine (like the corner of the table in our example).

A simple example will show why this is so.

The following part program looks, at first sight, suitable for drawing the 1in. square in Figure 6.1:

```
N10 G20 F10 G90 (set up imperial units, a slow feed rate
etc.)
N20 G0 Z2.0 (lift pen)
N30 G0 X0.8 Y0.3 (rapid to bottom left of square)
N40 G1 Z0.0 (pen down)
N50 Y1.3 (we can leave out the G1 as we have just done one)
N60 X1.8
N70 Y0.3 (going clockwise round shape)
N80 X0.8
N90 G0 X0.0 Y0.0 Z2.0 (move pen out of the way and lift it)
N100 M30 (end program)
```

Even if you cannot yet follow all the code it is easy to see what is happening. For example on line N30, the machine is told to move the Controlled Point to X=0.8, Y=0.3. By line N60 the Controlled Point will be at X=1.8, Y=1.3 and so the DROs will read:

X Axis 1.8000 Y Axis 1.3000 Z Axis 0.0000

The problem, of course, is that the square has not been drawn on the paper like in figure 6.1 but on the table near the corner. The part program writer has measured from the corner of the paper but the machine is measuring from its machine zero position.

6.2 Work offsets

Mach3, like all machine controllers, allows you to move the origin of the coordinate system or, in other words where it measures from (i.e. where on the machine is considered to be zero for moves of X, Y Z etc.)

This is called **offsetting** the coordinate system.

Figure 6.3 shows what would happen if we could offset the Current Coordinate system to the corner of the paper. **Remember** the G-code always moves the Controlled Point to the numbers given in the Current Coordinate system.

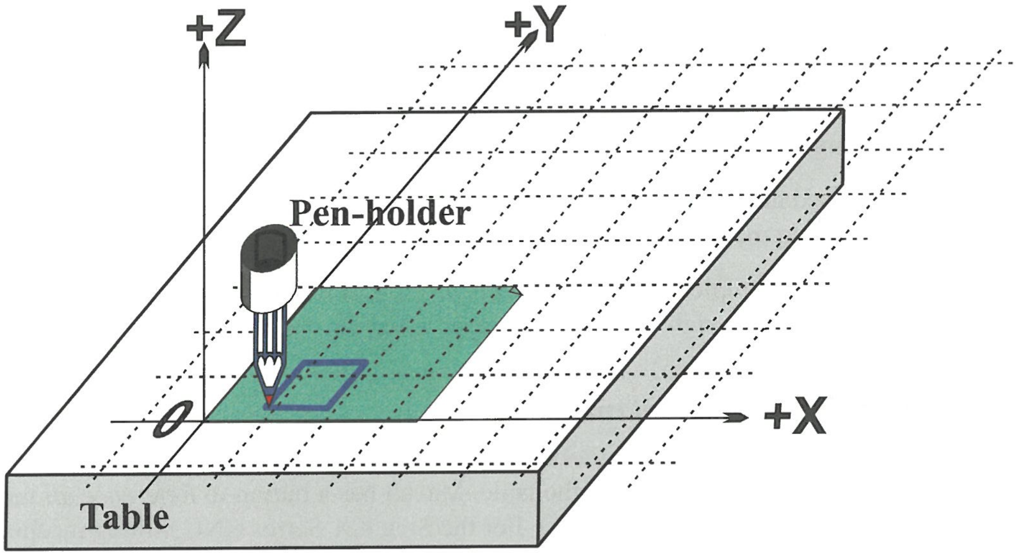


Figure 6.3 - Coordinate system origin offset to corner of paper

As there will usually be some way of fixing sheets of paper, one by one, in the position shown, this offset is called a Work **offset** and the 0, 0, 0 point is the origin of this coordinate system.

This offsetting is so useful that there are several ways of doing it using Mach3 but they are all organised using the Offsets screen (see Appendix 1 for a screenshot)

6.2.1 Setting Work origin to a given point

The most obvious way consists of two steps:

1. Display the Offsets screen. Move the Controlled Point (pen) to where you want the new origin to be. This can be done by jogging or, if you can calculate how far it is from the current position you can use G0s with manual data input
2. Click the *Touch* button next to each of the axes in the Current Work Offset part of the screen. On the first Touch you will see that the existing coordinate of the Touched axis is put into the Part Offset DRO and the axis DRO reads zero. Subsequent Touches on other axes copy the Current Coordinate to the offset and zero that axis DRO.

If you wonder what has happened, then the following may help. The work offset values are always added to the numbers in the axis DROs (i.e. the current coordinates of the controlled point) to give the absolute machine coordinates of the controlled point. Mach3 will display the absolute coordinates of the controlled point if you click the *Machine Coords* button. The LED flashes to warn you that the coordinates shown are absolute ones.

There is another way of setting the offsets which can be used if you know the position of where you want the new origin to be.

The corner of the paper is, by eye, about 2.6in. right and 1.4in. above the Home/Reference point at the corner of the table. Let's suppose that these figures are accurate enough to be used.

1. Type 2.6 and 1.4 into the X and Y Offset DROs. The Axis DROs will change (by having the offsets subtracted from them). Remember you have not moved the actual position of the Controlled point so its coordinates must change when you move the origin.

2. If you want to, you could check all is well by using the MDI line to G00 X0 Y0 Z0. The pen would be touching the table at the corner of the paper.

We have described using work offset number 1. You can use any numbers from 1 to 255. Only one is in use at any time and this can be chosen by the DRO on the Offsets screen or by using G-codes (G54 to G59 P253) in your part program.

The final way of setting a work offset is by typing a new value into an axis DRO. The current work offset will be updated so the controlled point is referred to by the value now in the axis DRO. Notice that the machine does not move. It is merely that the origin of the coordinate system has been changed. The Zero-X, Zero-Y etc. buttons are equivalent to typing 0 into the corresponding axis DRO.

You are advised not to use this final method until you are confident using work offsets that have been set up using the Offsets screen.

So, to recap the example, by offsetting the Current Coordinate system by a work offset we can draw the square at the right place on the paper wherever we have taped it down to the table.

6.2.2 Home in a practical machine

As mentioned above, although it looks tidy at first sight, it is often not a good idea to have the Home Z position at the surface of the table. Mach3 has a button to *Reference all* the axes (or you can Reference them individually). For the Sieg KX Series CNC milling machine, which has home switches installed, this will move each linear axes (or chosen axis) until its switch is operated, then move slightly off it. The absolute machine coordinate system origin (i.e. machine zero) is then set to given X, Y, Z etc. values - frequently 0.0. You can actually define a non-zero value for the home switches if you want but ignore this for now!

The Z home switch is generally set at the highest Z position above the table. Of course if the reference position is machine coordinate Z=0.0 then all the working positions are lower and will be negative Z values in machine coordinates.

Again if this is not totally clear at present do not worry. Having the Controlled Point (tool) out of the way when homed is obviously practically convenient and it is easy to use the work offset(s) to set a convenient coordinate system for the material on the table.

6.3 What about different lengths of tool?

If you are feeling confident so far then it is time to see how to solve another practical problem.

Suppose we now want to add a red rectangle to the drawing.

We jog the Z axis up and put the red pen in the holder in place of the blue one. Sadly the red pen is longer than the blue one so when we go to the Current Coordinate System origin, the tip smashes into the table. (Figure 6.5)

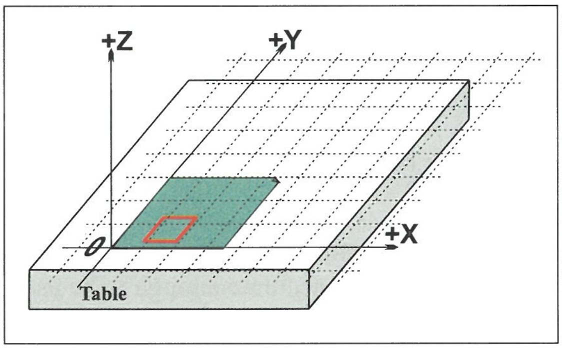


Figure 6.4 - Now we want another colour

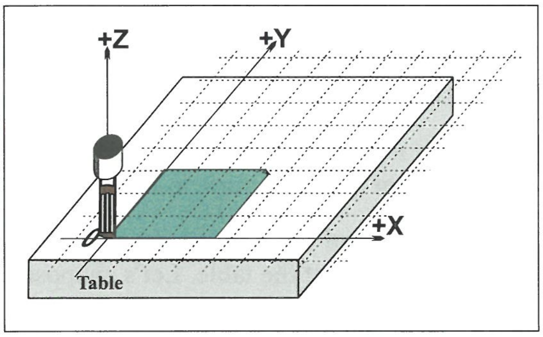


Figure 6.5 - Disaster at 0,0,0!

Mach3, like other CNC controllers, has a way for storing information about the tools (pens in our system). This **Tool Table** allows you to tell the system about up to 256 different tools.

On the Offsets screen you will see space for a Tool number and information about the tool. The DROs are labelled *Z-offset, Diameter* and *T*. Ignore the DRO Touch Correction and its associated button marked On/Off for now.

By default you will have Tool #0 selected but its offsets will be switched OFF.

Information about the tool diameter is also used for Cutter Compensation (q.v.)

6.3.1 Presettable tools

We will assume your machine has a tool-holder system, which lets you put a tool in at exactly the same position each time. This might be a mill with lots of chucks or something like an Autolock chuck (figure 6.6 - where the centre-hole of the tool is registered against a pin). If your tool position is different each time then you will have to set up the offsets each time you change it. This will be described later.

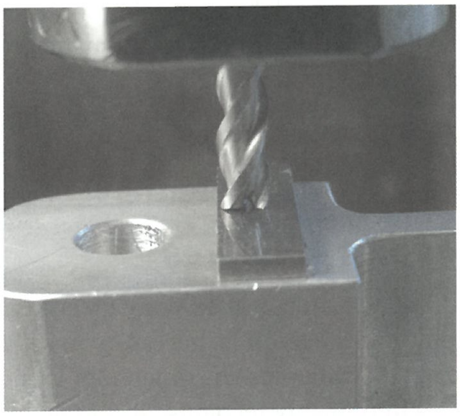


Figure 6.6 – End mill in a presettable holder

In our drawing machine, suppose the pens register in a blind hole that is 1in. deep in the pen holder. The red pen is 4.2in. long and the blue one 3.7in. long.

1. Suppose the machine has just been referenced/ homed and a work offset defined for the corner of the paper with Z = 0.0 being the table using the bottom face of the empty pen holder. You would jog the Z axis up say to 5in. and fit the blue pen. Enter "1" (which will be the blue pen) in the Tool number DRO but do not click *Offset On/Off* to ON yet. Jog the Z down to touch the paper. The Z axis DRO would read 2.7 as the pen sticks 2.7in. out of the holder. Then you click the Touch button by the Z offset. This would load the (2.7in.) into the Z offset of Tool #1. Clicking the *Offset On/Off* toggle would light the LED and apply the tool offset and so the Z axis DRO will read 0.0 You could draw the square by running the example part program as before.
2. Next, to use the red pen you would jog the Z axis up (say to Z = 5.0 again), take out the blue pen and put in the red. Physically swapping the pens obviously does not alter the axis DROs. Now you would switch Off the tool offset LED, select Tool #2 , jog and *Touch* at the corner of the paper. This would set up tool 2's Z offset to 3.2in.. Switching On the offset for Tool #2 again will display Z = 0.0 on the axis DRO so the part program would draw the red square (over the blue one).
3. Now that tools 1 and 2 are set up you can change them as often as you wish and get the correct Current Coordinate system by selecting the appropriate tool number and switching its offsets on. This tool selection and switching on and off of the offsets can be done in the part program (T word, M6, G43 and G49) and there are DROs on the standard Program Run screen.

6.3.2 Non-presettable tools

Some tool holders do not have a way of refitting a given tool in exactly the same place each time. For example the collet of a router is usually bored too deep to bottom the tool. In this case it may still be worth setting up the tool offset (say with tool #1) each time it is changed. If you do it this way you can still make use of more than one work offset (see 2 and 3 pin fixtures illustrated below). If you do not have a physical fixture it may be just as easy to redefine the Z of the work offsets each time you change the tool.

6.4 How the offset values are stored

The 254 work offsets are stored in one table in Mach3. The 255 tool offsets and diameters are stored in another table. You can view these tables using the *Work Offsets Table* and *Tool Offsets Table* buttons on the offsets screen. These tables have space for additional information which is not at present used by Mach3

Mach3 will generally try to remember the values for all work and tool offsets from one run of the program to another but will prompt you on closing down the program to check that you **do** want to save any altered values. Check boxes on the Config>State dialog (q.v.) allow you to change this behaviour so that Mach3 will either automatically save the values without bothering to ask you or will never save them automatically.

However the automatic saving options are configured, you can use the *Save* button on the dialogs which display the tables to force a save to occur.

6.5 Drawing lots of copies - Fixtures

Now imagine we want to draw on many sheets of paper. It will be difficult to tape each one in the same place on the table and so it will be necessary to set the work offsets each time. Much better would be to have a plate with pins sticking out of it and to use pre-punched paper to register on the pins. You will probably recognise this as an example of a typical fixture which has long been used in machine shops. Figure 6.7 shows the machine so equipped. It would be common for the fixture to have dowels or something similar so that it always mounts in the same place on the table.

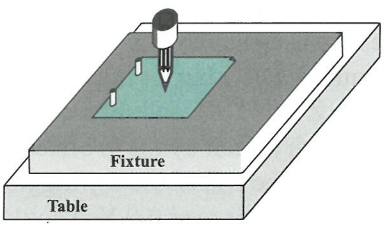


Figure 6.7 - Machine with two pin fixture

We could now move the Current Coordinate system by setting the work offsets #1 to the corner of the paper on the actual fixture. Running the example program would draw the square exactly as before. This will of course take care of the difference in Z coordinates caused by the thickness of the fixture. We can put new pieces of paper on the pins and get the square in exactly the right place on each with no further setting up.

We might also have another fixture for three-hole paper (Figure 6.8) and might want to swap between the two and three pin fixtures for different jobs so work offset #2 could be defined for the corner of the paper on the three pin fixture.

You can, of course define any point on the fixture as the origin of its offset coordinate system. For the drawing machine, we would want to make the bottom left corner of the paper be X=0 & Y=0 and the top surface of the fixture be Z=0.

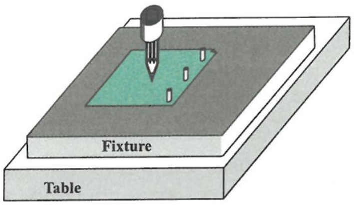


Figure 6.8 - Three pin fixture

It is common for one physical fixture to be able to be used for more than one job. Figure 6.9 shows the two and three hole fixtures combined. You would of course have two entries in the work offset corresponding to the offsets to be used for each. In figure 6.8, the Current Coordinate system is shown set for using the two-hole paper option.

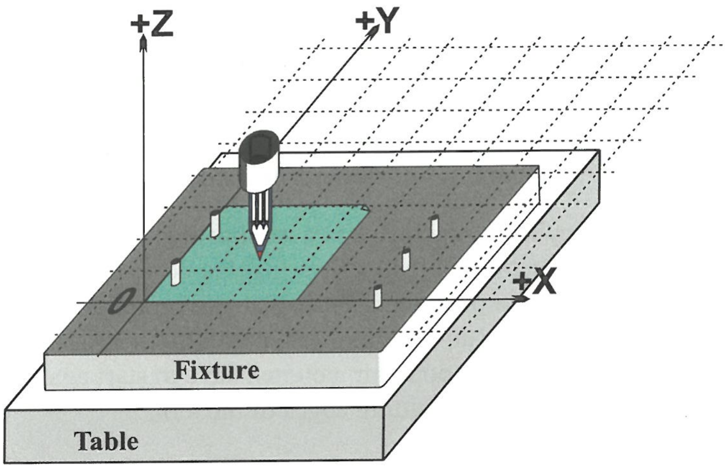


Figure 6.9 - A double fixture

6.6 Practicalities of "Touching"

6.6.1 End mills

On a manual machine tool it is quite easy to feel on the handles when a tool is touching the work but for accurate work it is better to have a feeler (perhaps a piece of paper or plastic from a candy bar) or slip gauge so you can tell when it is being pinched. This is illustrated on a mill in figure 6.10.

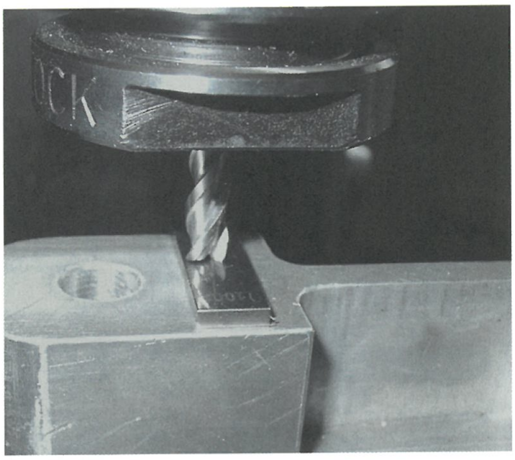


Figure 6.10 - Using a slip gauge when touching Z offset on a mill

On the Offset screen you can enter the thickness of this feeler or slip gauge into the DRO beside the *Set Tool Offset* button. When you use *Sret Tool Offset* to set an offset DRO for a tool, then the thickness of the gauge will be allowed for.

For example, suppose you had the axis DRO Z = -3.518 with the 0.1002in. slip lightly held. Choose Tool #3 by typing 3 in the Tool DRO. Enter 0.1002 in the DRO in Gauge Block Height and click Set Tool Offset. After the touch, the axis DRO reads Z = 0.1002 (i.e the Controlled Point is 0.1002) and tool 3 will have the Z offset -0.1002. Figure 6.11 shows this process just before clicking *Set Tool Offset.*

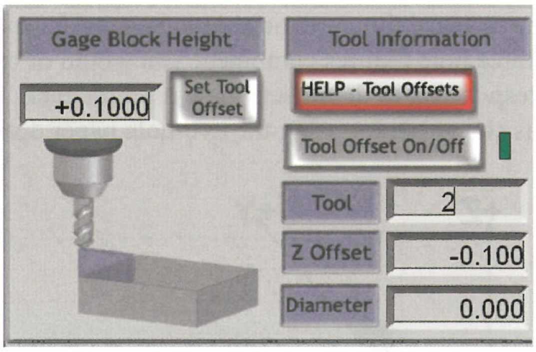


Figure 6.11 – Entering Z offset data

If you have an accurate cylindrical gauge and a reasonably sized flat surface on the top of the workpiece, then using it can be even better than jogging down to a feeler or slip gauge. Jog down so that the roller will not pass under the tool. Now very slowly jog up until you can just roll it under the tool. Then you can click the *Touch* button. There is an obvious safety advantage in that jogging a bit too high does no harm; you just have to start again. Jogging **down** to a feeler or gauge risks damage to the cutting edges of the tool.

6.6.2 Edge finding

It is very difficult to accurately set a mill to an edge in X or Y due to the flutes of the tool. A special edge-finder tool helps here. Figure 6.12 shows the minus X edge of a part being found.

The Touch Correction can be used here as well. You will need the radius of the probe tip and the thickness of any feeler or slip gauge.

6.7 G52 & G92 offsets

There are two further ways of offsetting the Controlled Point using G-codes G52 and G92.

When you issue a G52 you tell Mach3 that for any value of the controlled point (e.g. X=0, Y= 0) you want the actual machine position offset by adding the given values of X, Y and/or Z.

When you use G92 you tell Mach3 what you want the coordinates of the current Controlled Point to be values given by X, Y and/or Z.

Neither G52 nor G92 move the tool, they just add another set of offsets to the origin of the Current Coordinate system.

6.7.1 Using G52

A simple example of using G52 is where you might wish to produce two identical shapes at different

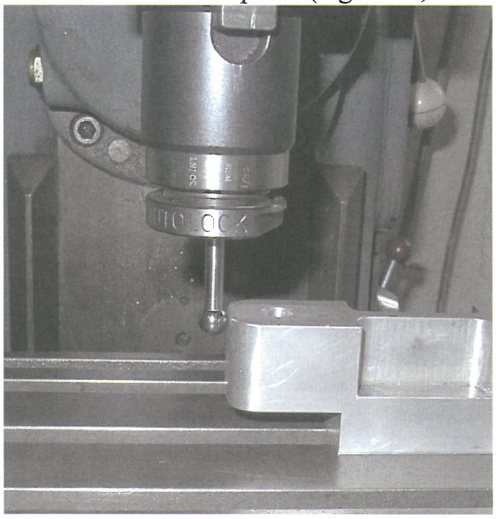


Figure 6.12 - Edge-finder in use on a mill

places on the workpiece. The code we looked at before draws a 1in. square with a corner at X = 0.8, Y = 0.3:

```
G20 F10 G90 (set up imperial units, a slow feed rate etc.)
G0 Z2.0 (lift pen)
G0 X0.8 Y0.3 (rapid to bottom left of square)
G1 Z0.0 (pen down)
Y1.3 (we can leave out the G1 as we have just done one)
X1.8
Y0.3 (going clockwise round shape)
X0.8
G0 X0.0 Y0.0 Z2.0 (move pen out of the way and lift it)
```

If we want another square but the second one with its corner at X= 3.0 and Y = 2.3 then the above code can be used twice but using G52 to apply an offset before the second copy.

```
G20 F10 G90 (set up imperial units, a slow feed rate etc.)

G0 Z2.0 (lift pen)
G0 X0.8 Y0.3 (rapid to bottom left of square)
G1 Z0.0 (pen down)
Y1.3 (we can leave out the G1 as we have just done one)
X1.8
Y0.3 (going clockwise round shape)
X0.8
G0 Z2.0 (lift pen)

G52 X2.2 Y2 (temporary offset for second square)

G0 X0.8 Y0.3 (rapid to bottom left of square)
G1 Z0.0 (pen down)
Y1.3 (we can leave out the G1 as we have just done one)
X1.8
Y0.3 (going clockwise round shape)
X0.8

G52 X0 Y0 (Get rid of temporary offsets)

G0 X0.0 Y0.0 Z2.0 (move pen out of the way and lift it)
```

Copying the code is not very elegant but as it is possible to have a G-code subroutine (See M98 and M99) the common code can be written once and called as many times as you need – twice in this example.

The subroutine version is shown below. The pen up/down commands have been tidied up and the subroutine actually draws at 0,0 with a G52 being used for setting the corner of both squares:

```
G20 F10 G90 (set up imperial units, a slow feed rate etc.)
G52 X0.8 Y0.3 (start of first square)
M98 P1234      (call subroutine for square in first position)
G52 X3 Y2.3 (start of second square)
M98 P1234      (call subroutine for square in second
position)
G52 X0 Y0 {IMPORTANT - get rid of G52 offsets)
M30 (rewind at end of program)
```

```
O1234
(Start of subroutine 1234)
G0 X0 Y0 (rapid to bottom left of square)
G1 Z0.0 (pen down)
Y1 (we can leave out the G1 as we have just done one)
X1
Y0 (going clockwise round shape)
X0
G0 Z2.0 (lift pen)
M99 (return from subroutine)
```

Notice that each G52 applies a new set of offsets which take no account of any previously issued G52.

6.7.2 Using G92

The simplest example with G92 is, at a given point, to set X & Y to zero but you can set any values. The easiest way to cancel G92 offsets is to enter "G92.1" on the MDI line.

6.7.3 Take care with G52 and G92

You can specify offsets on as many axes as you like by including a value for their axis letter. If an axis name is not given, then its offset remains unaltered.

Mach3 uses the same internal mechanisms for G52 and G92 offsets; it just does different calculations with your X, Y and Z words. If you use G52 and G92 together, you (and even Mach3) will become so confused that disaster will inevitably occur. If you really want to prove you have understood how they work, set up some offsets and move the controlled point to a set of coordinates, say X=2.3 and Y=4.5. Predict the absolute machine coordinates you should have and check them by making Mach3 display machine coordinates with the "Mach" button.

Do not forget to clear the offsets when you have used them.

Warning! Almost everything that can be done with G92 offsets can be done better using work offsets or perhaps G52 offsets. Because G92 relies on where the controlled point is as well as the axis words at the time G92 is issued, changes to programs can easily introduce serious bugs leading to crashes.

Many operators find it hard to keep track of three sets of offsets (Work, Tool and G52/G92) and if you get confused you will soon break either your tool or worse your machine!

6.8 Tool diameter

Suppose the blue square drawn using our machine is the outline for a hole in the lid of a child's shape-sorter box into which a blue cube will fit. Remember G-codes move the Controlled Point. The example part program drew a 1in. square. If the tool is a thick felt pen then the hole will be significantly smaller than 1in. square. See figure 6.13.

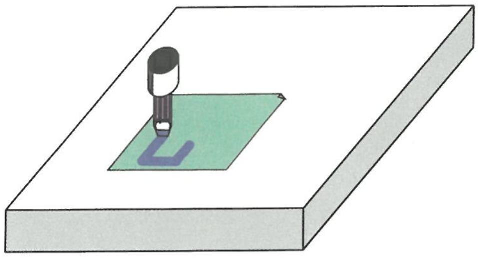


Figure 6.13 - Using a large diameter tool (felt pen)

The same problem obviously occurs with an end mill/ slot drill. You may want to cut a pocket or be leaving an island. These need different compensation.

This sounds easy to do but in practice there are many "devils in the detail" concerned with the beginning and end of the cutting. It is usual for a Wizard or your CAD/CAM software to deal with these issues. Mach3, however, allows a part program to compensate for the diameter of the chosen tool with the actual cutting moves being specified as, say, the 1in. square.

This feature is important if the author of the part program does not know the exact diameter of the cutter that will be used (e.g. it may be smaller than nominal due to repeated sharpening). The tool table lets you define the diameter of the tool or, in some applications, the difference from the nominal tool diameter of the actual tool being used – perhaps after multiple sharpening. See Cutter Compensation chapter 8 for full details.

7. DXF, HPGL and image file import

> This chapter covers importing files and their conversion to part programs by Mach3.
> It assumes a limited understanding of simple G-codes and their function.

7.1 Introduction

As you will have seen, Mach3Mill uses a part program to control the tool movement in the Sieg KX Series CNC milling machine. You may have written part programs by hand (spiral.txt is such an example) or generated them using a CAD/CAM (Computer Aided Design/Computer Aided Manufacturing) system.

Importing files which define "graphics" in DXF, HPGL, BMP or JPEG formats provides an intermediate level of programming. It is easier than coding by hand but provides much less control of the machine than a program output by a CAD/CAM package.

The Automatic Z control feature (q.v) and repetitive execution decrementing the Inhibit-Z value is a powerful tool for making a series of roughing cuts based on imported DXF and HPGL files.

7.2 DXF import

Most CAD programs will allow you to output a file in DXF format even though they do not offer any CAM features. A file will contain the description of the start and finish of lines and arcs in the drawing together with the layer that they are drawn on. Mach3 will import such a file and allow you to assign a particular tool, feed rate and "depth of cut" to each layer. The DXF file must be in **text** format, not binary, and Mach3 will only import **lines, polylines, circles and arcs (not text)**.

During import, you can (a) optimise the order of the lines to minimise non-cutting moves. (b) use the actual coordinates of the drawing or offset them so that the bottom leftmost point is 0,0.

(The DXF import is in the file menu. The dialog in figure 7.1 is displayed.

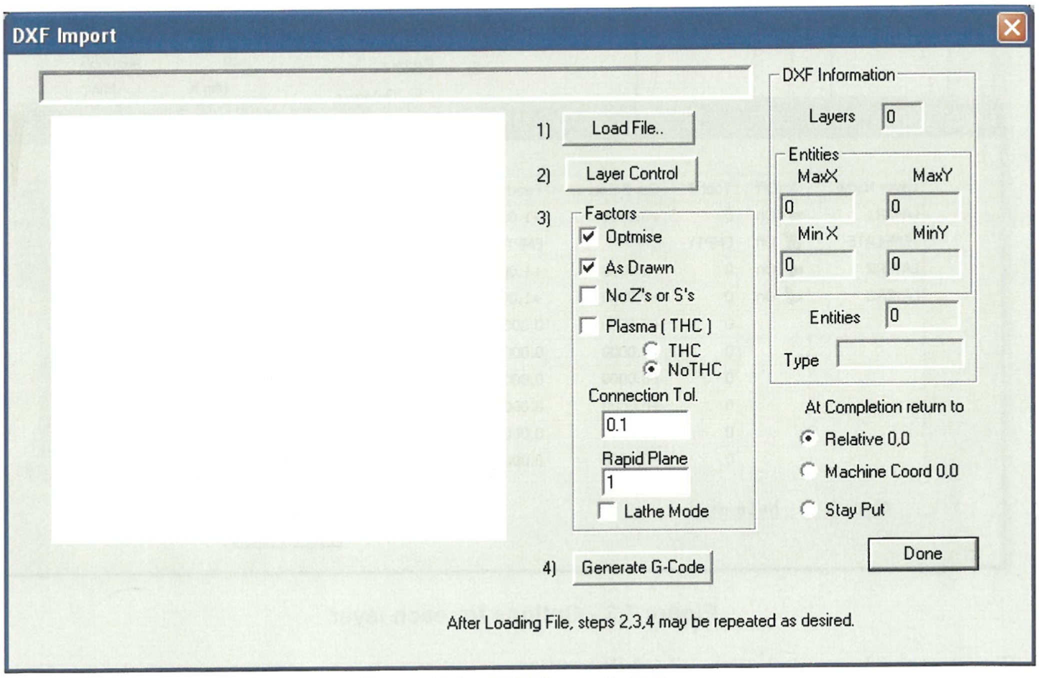


Figure 7.1 - DXF import dialog

7.2.1 File loading

This shows the four stages of importing the file. Step 1 is to load the DXF file. Clicking the *Load File* button displays an open file dialog for this. Figure 7.2 shows a file with two rectangles and a circle.

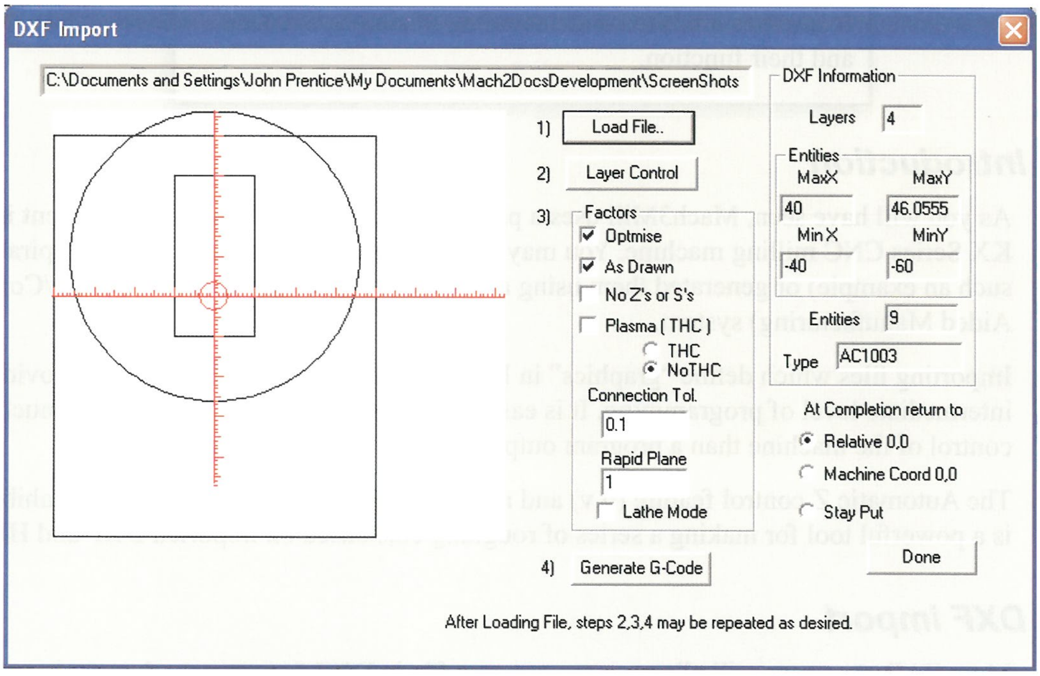


Figure 7.2 - a drawing of eight lines and one circle

7.2.2 Defining action for layers

The next stage is to define how the lines on each layer of the drawing are to be treated. Click the *Layer Control* button to display the dialog shown in figure 7.3.

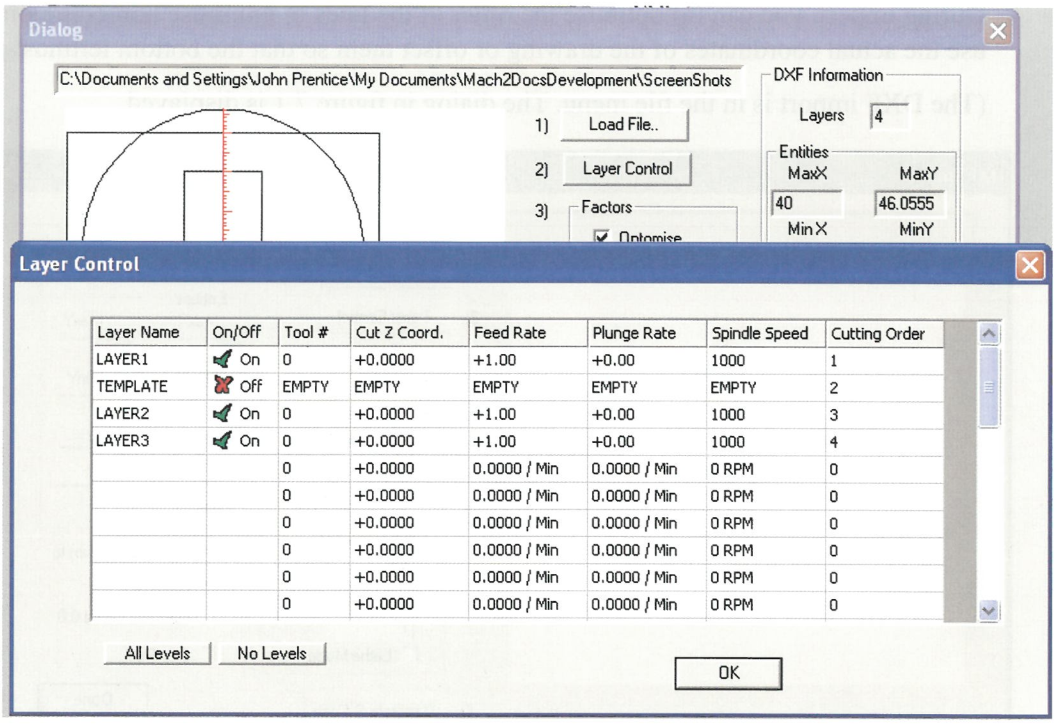


Figure 7.3 - Options for each layer

Turn on the layer or layers which have lines on them that you want to cut, choose the tool to use, the depth of cut, the feed rate to use, the plunge rate, the spindle speed and the order in which you want the layers cutting. Notice that the "Depth of cut" value is the Z value to be used in the cut so, if the surface of the work is Z = 0, it will be a negative value. The order may be important for issues like cutting holes out of a piece before it is cut from the surrounding material.

7.2.3 Conversion options

Next you choose the options for the conversion process (see step 3 on figure 7.2).

DXF Information: Gives general details of your file which are useful for diagnostic purposes.

Optimise: If *Optimise* is not checked then the entities (lines etc.) will be cut in the order in which they appear in the DXF file. If it is checked then they will be re-ordered to minimise the amount of rapid traverse movement required. Note that the cuts are always optimised to minimise the number of tool changes required.

As Drawn: If *As Drawn* is not checked then the zero coordinates of the G-code will be the "bottom left corner" of the drawing. If it is checked then the coordinates of the drawing will be the coordinates of the G-code produced.

Connection Tol. Two lines on the same layer will be considered to join if the distance between their ends is less than the value of this control. This means that they will be cut without a move to the "Rapid Plane" being inserted between them. If the original drawing was drawn with some sort of "snap" enabled then this feature is probably not required.

Rapid plane: This control defines the Z value to be adopted during rapid moves between entities in the drawing.

7.2.4 Generation of G-code

Finally click *Generate G-code* to perform step 4. It is conventional to save the generated G-code file with a .TAP extension but this is not required and Mach3 will not insert the extension automatically.

You can repeat steps 2 to 4, or indeed 1 to 4 and when you have finished these click Done.

Mach3 will load the last G-code file which you have generated. Notice the comments identifying its name and date of creation.

Notes:

- The generated G-code has feed rates depending on the layers imported.
- DXF input is good for simple shapes as it only requires a basic CAD program to generate the input file and it works to the full accuracy of your original drawing
- For milling you will have to make your own manual allowances for the diameter of the cutter. The DXF lines will be the path of the centreline of the cutter. This is not straightforward when you are cutting complex shapes.
- The program generated from a DXF file does not have multiple passes to rough out a part or clear the centre of a pocket. To achieve these automatically you will need to use a CAM program
- If your DXF file contains "text" then this can be in two forms depending on the program which generated it. The letters may be a series of lines. These will be imported into Mach3. The letters may be DXF Text objects. In this case they will be ignored. Neither of these situations will give you G-code which will engrave letters in the font used in the original drawing although the lines of an outline font may be satisfactory with a small v-point or bullnose cutter.

7.3 HPGL import

HPGL files contain lines drawn with one or more pens. Mach3Mill makes the same cuts for all pens. HPGL files can be created by most CAD software and often have the filename extension .HPL or .PLT.

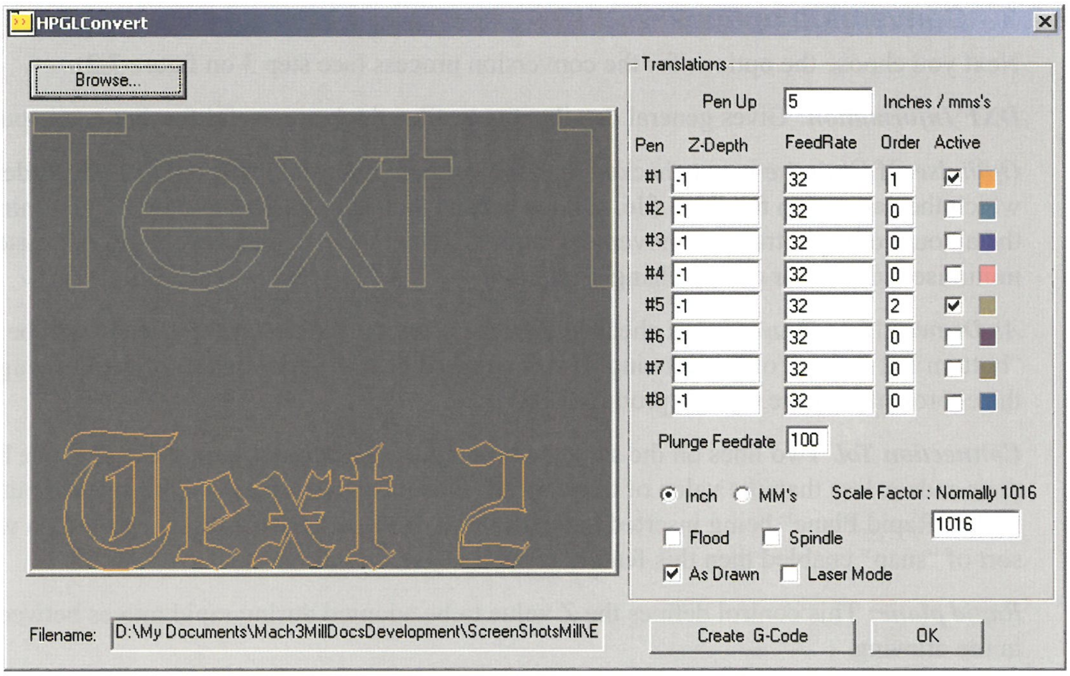


Figure 7.4 – HPGL import filter

7.3.1 About HPGL

An HPGL file represents objects to a lower precision than DXF and uses straight line segments to represent all curves even if they are circles.

The import process for HPGL is similar to DXF in that a .TAP file is produced which contains the G-code produced from the HPGL

7.3.2 Choosing the file to import

The import filter is accessed from File>Import HPGL/BMP/JPG and the HPGL button on the dialog. Figure 7.4 shows the import dialog itself.

First choose the *Scale* corresponding to that at which the HPGL file was produced. This is usually 40 HPGL units per millimetre (1016 units per inch). You can change this to suit different HPGL formats or to scale your g-code file. For example, choosing 20 (rather than 40) would double the size of the objects defined.

Now enter the name of the file containing the HPGL data or "Browse" for it. The default extension for browsing is .PLT so it is convenient to create your files named like this.

7.3.3 Import parameters

The *Pen Up* control is the Z values (in the current unit in which Mach3 is working) to be used when making moves. Pen Up will typically need to position the tool just above the work.

Different depths of cut and feed rates can be programmed for each of the "pens" used to produce the drawing. You can also define the order in which you want cuts to be made. This allows cutting the inside of an object before you cut it from the stock!

7.3.4 Writing the G-code file

Finally, having defined the import translations, click *Import File* to actually import the data to Mach3Mill. You will be prompted for the name to use for the file which will store the generated code. You should type the full name including the extension which you wish to use or select an existing file to overwrite. Conventionally this extension will be .TAP.

After writing the file click *OK* to return to Mach3. Your G-Code file will have been loaded.

Notes:

- The import filter is run by suspending Mach3 and running the filter program. If you switch to the Mach3Mill screen (for example by accidentally clicking on it) then it will appear to have locked up. You can easily continue by using the Windows task bar to return to the filter and completing the import process. This is similar to the way the Editor for part programs is run.
- If your .TAP file already exists and is open in Mach3, then the import filter will not be able to write to it. Suppose you have tested an import and want to change the translations by importing again, then you need to make sure that you close the .TAP file in Mach3Mill before repeating the import.
- It is generally easiest to work in metric units throughout when importing HPGL files.
- For milling you will have to make your own manual allowances for the diameter of the cutter. The HPGL lines will be the path of the centreline of the cutter. This allowance is not straightforward to calculate when you are cutting complex shapes.
- The program generated from a HPGL file does not have multiple passes to rough out a part or clear the centre of a pocket. To achieve these automatically you will need to use a CAM program.

7.4 Bitmap import (BMP & JPEG)

This option allows you to import a photograph and generate a G-code program which will render different shades of grey different depths of cut. The result is a photo-realistic engraving.

7.4.1 Choosing file to import

The import filter is accessed from File>Import HPGL/BMP/JPG and the JPG/BMP button on the dialog.

The first step is to define the file containing the image using the *Load Image File* button. When the file is loaded, a dialog prompts you for the area on the workpiece into which the image is to be fitted. You can use inch or metric units as you wish depending on the G20/21 mode in which you will run the generated part program. Figure 7.5 shows this dialog.

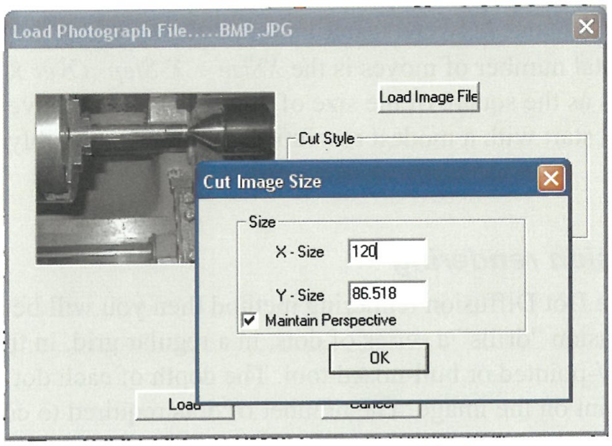


Figure 7.5 – Size of photographic import

The *Maintain Perspective* checkbox automatically computes the Y-size if a given X-size is specified and vice versa so as to preserve the aspect ratio of the original photograph. If the image is in colour it will be converted to monochrome as it is imported.

7.4.2 Choose type of rendering

Next you select the method of rendering the image. This is defining the path of the tool as it "rasterizes" the image. *Raster X/Y* cuts along the X axis moving the Y axis at the end of each X-line. *Raster Y/X* makes the raster lines go in the Y direction incrementing X for each line. *Spiral* starts at the outside of a circle bounding the image and moves into the centre. Each raster line is made up of a series of straight lines with the height of the Z coordinates of the ends depending on the shade of grey of that part of the picture.

7.4.3 Raster and spiral rendering

As you select one of these raster methods, you will be prompted by a dialog for the *step-over* values. See figure 7.6.

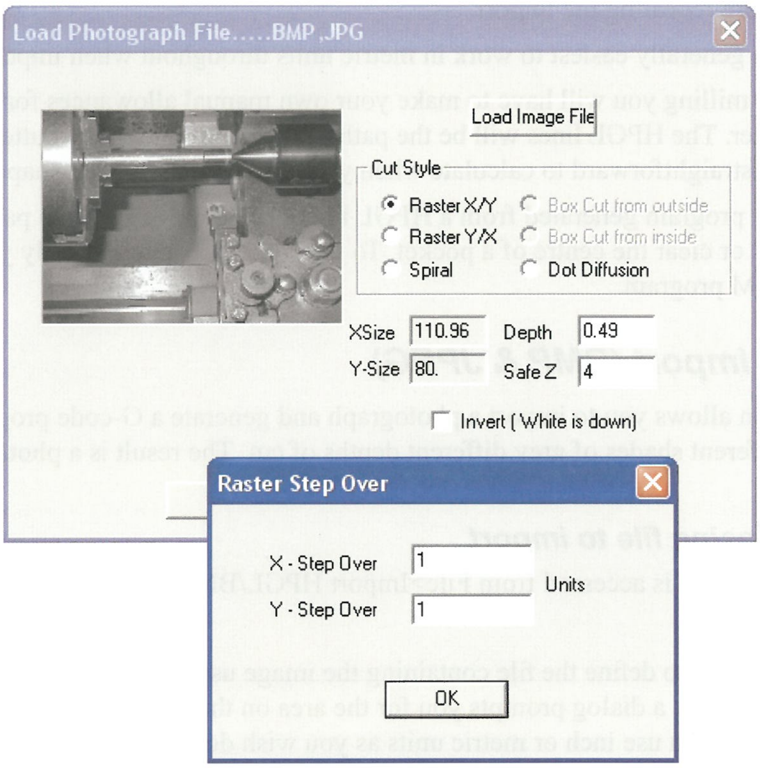


Figure 7.6 – Defining the Step-over

These define the distance between raster lines and the length of the short segments making up each line. The total number of moves is the *XSize ÷ X-Step- Over* x *YSize ÷ Y-Step-Over* and, of course, increases as the square of the size of the object and the inverse square of the size of step-over. You should start with a modest resolution to avoid impossibly big files and long cutting times.

7.4.4 Dot diffusion rendering

If you choose the Dot Diffusion rendering method then you will be asked for a different set of details. Dot diffusion "drills" a series of dots, in a regular grid, in the work. Typically these will be formed by a V-pointed or bull-nosed tool. The depth of each dot is determined by the shade of grey at the point on the image. The number of dots required to cover the area is computed by the filter on the basis of the shape of the tool and the depth (relief) of engraving you select. Figure 7.7 illustrates the data required.

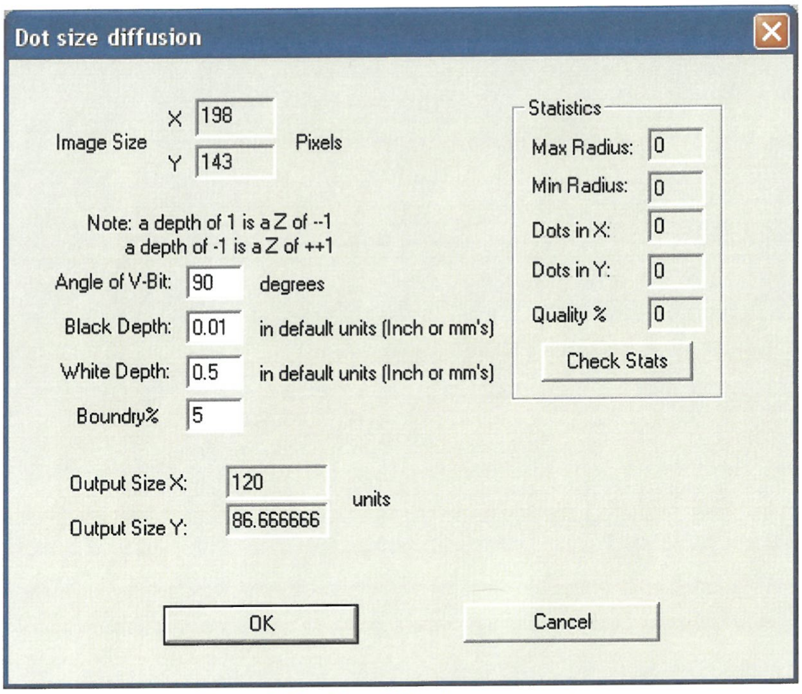


Figure 7.7 – Dot diffusion parameters

Each dot consists of a move to its location, a Z move to its depth and a Z move to above the work. You must prepare your image with a suitable photo editor to have a reasonable number of pixels to control the computation load when diffusing the dots. The statistics obtained by the *Check Stats* button will give you an idea of how sensible your choice of parameters has been.

Now having defined the rendering technique, you set the *Safe Z* at which moves over the work will be done and choose if black or white is to be the deepest cut.

7.4.5 Writing the G-code file

Finally click *Convert* to actually import the data into Mach3Mill. You will be prompted for the name to use for the file which will store the generated code. You should type the full name including the extension which you wish to use or select an existing file to overwrite. Conventionally this extension will be .TAP.

Notes:

- The import filter is run by suspending Mach3 and running the filter program. If you switch to the Mach3Mill screen (for example by accidentally clicking on it) then it will appear to have locked up. You can easily continue by using the Windows task bar to return to the filter and completing the import process. This is similar to the way the Editor for part programs is run.
- If your .TAP file already exists and is open in Mach3, then the import filter will not be able to write to it. Suppose you have tested an import and want to change the translations by importing again, then you need to make sure that you close the .TAP file in Mach3Mill before repeating the import.
- You will need to define the feed rate to be used using MDI or by editing the part program before it is run.
- Dot Diffusion places big demands on the performance of your Z axis. You must set the Safe Z as low as possible to minimise the distance travelled and have the Z axis motor tuning very carefully set. Lost steps part of the way through an engraving will ruin the job!

8. Cutter compensation

> Cutter compensation is a feature of Mach3 which you many never have to use. Most CAD/CAM programs can be told the nominal diameter of your mill and will output part programs which cut the part outline or pocket which you have drawn, by itself allowing for the tool diameter. Because the CAD/CAM software has a better overall view of the shapes being cut, it may be able to do a better job than Mach3 can when avoiding gouges at sharp internal corners.
> Having compensation in Mach3 allows you to: (a) use a tool different in diameter from that programmed (e.g. because it has been reground) or (b) to use a part program that describes the desired outline rather than the path of the centre of the tool (perhaps one written by hand).
> However, as compensation is not trivial, it is described in this chapter should you need to use it?
> This feature is under development and may change significantly in the final release of Mach3.

8.1 Introduction to compensation

As we have seen, Mach3 controls the movement of the Controlled Point. In practice no tool (except perhaps a V-engraver) is a point so cuts will be made at a different place to the Controlled Point depending on the radius of the cutter.

It is generally easiest to allow your CAD/CAM software to take account of this when cutting out pockets or the outline of shapes. Mach3 does, however, support calculations to compensate for the diameter (radius) of the cutter. In industrial applications this is aimed at allowing for a cutter which, through regrinding, is not exactly the diameter of the tool assumed when the part program was written. The compensation can be enabled by the machine operator rather than requiring the production of another part program.

On the face of it, the problem should be easy to solve. All you need to do is to offset the controlled point by an appropriate X and Y to allow for the tool radius. Simple trigonometry gives the distances depending on the angle the direction of cut makes to the axes.

In practice it is not quite so easy. There are several issues but the main one is that the machine has to set a Z position before it starts cutting and at that time it does not know the direction in which the tool is going to be moving. This problem is solved by providing "pre-entry moves" which take place in waste material of the part. These ensure that the compensation calculations can be done before the actual part outline is being cut. Choice of a path which runs smoothly into the part's outline also optimises the surface finish. An exit move is sometimes used to maintain the finish at the end of a cut.

8.2 Two Kinds of Contour

Mach3 handles compensation for two types of contour:

- The contour given in the part program code is the edge of material that is not to be machined away. We will call this type a "material edge contour". This is the sort of code that might be "hand written"
- The contour given in the NC code is the tool path that would be followed by a tool of exactly the correct radius. We will call this type a "tool path contour". This is the sort of code that a CAD/CAM program might produce if it is aware of the intended cutter diameter

The interpreter does not have any setting that determines which type of contour is used, but the numerical description of the contour will, of course, differ (for the same part geometry) between the two types and the values for diameters in the tool table will be different for the two types.

8.2.1 Material Edge Contour

When the contour is the edge of the material, the outline of the edge is described in the part program. For a material edge contour, the value for the diameter in the tool table is the actual value of the diameter of the tool. The value in the table must be positive. The NC code for a material edge contour is the same regardless of the (actual or intended) diameter of the tool.

Example1:

Here is an NC program which cuts material away from the outside of the triangle in figure 8.1 below. In this example, the cutter compensation radius is the actual radius of the tool in use, which is 0.5, The value for the diameter in the tool table is twice the radius, which is 1.0.

```
N0010 G41 G1 X2 Y2 (turn compensation on and make entry move)
N0020 Y-1 (follow right side of triangle)
N0030 X-2 (follow bottom side of triangle)
N0040 X2 Y2 (follow hypotenuse of triangle)
N0050 G40 (turn compensation off)
```

This will result in the tool following a path consisting of an entry move and the path shown on the left going clockwise around the triangle. Notice that the coordinates of the triangle of material appear in the NC code. Notice also that the tool path includes three arcs which are not explicitly programmed; they are generated automatically.

8.2.2 Tool Path Contour

When the contour is a tool path contour, the path is described in the part program. It is expected that (except for during the entry moves) the path is intended to create some part geometry. The path may be generated manually or by a CAD/CAM program, considering the part geometry which is intended to be made. For Mach3 to work, the tool path must be such that the tool stays in contact with the edge of the part geometry, as shown on the left side of figure 8.1. If a path of the sort shown on the right of figure 8.1 is used, in which the tool does not stay in contact with the part geometry all the time, the interpreter will not be able to compensate properly when undersized tools are used.

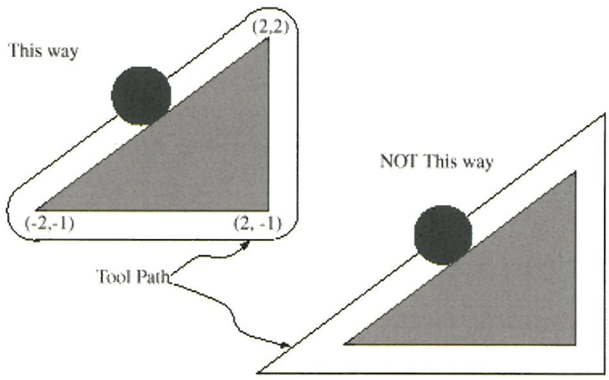


Figure 8.1 - Two possible tool paths to cut triangle

For a tool path contour, the value for the cutter diameter in the tool table will be a small positive number if the selected tool is slightly oversized and will be a small negative number if the tool is slightly undersized. As implemented, if a cutter diameter value is negative, the interpreter compensates on the other side of the contour from the one programmed and uses the absolute value of the given diameter. If the actual tool is the correct size, the value in the table should be zero.

Tool Path Contour example:

Suppose the diameter of the cutter currently in the spindle is 0.97, and the diameter assumed in generating the tool path was 1.0. Then the value in the tool table for the diameter for this tool should be -0.03. Here is an NC program which cuts material away from the outside of the triangle in the figure.

```
N0010 G1 X1 Y4.5 (make alignment move)
N0020 G41 G1 Y3.5 (turn compensation on and make first entry move)
N0030 G3 X2 Y2.5 I1 (make second entry move)
N0040 G2 X2.5 Y2 J-0.5 (cut along arc at top of tool path)
N0050 G1 Y-1 (cut along right side of tool path)
N0060 G2 X2 Y-1.5 I-0.5 (cut along arc at bottom right of tool path)
N0070 G1 X-2 (cut along bottom side of tool path)
N0080 G2 X-2.3 Y-0.6 J0.5 (cut along arc at bottom left of tool path)
N0090 G1 X1.7 Y2.4 (cut along hypotenuse of tool path)
N0100 G2 X2 Y2.5 I0.3 J-0.4 (cut along arc at top of tool path)
N0110 G40 (turn compensation off)
```

This will result in the tool making an alignment move and two entry moves, and then following a path slightly inside the path shown on the left in figure 8.1 going clockwise around the triangle. This path is to the right of the programmed path even though G41 was programmed, because the diameter value is negative.

8.2.3 Programming Entry Moves

In general, an alignment move and an entry moves are needed to begin compensation correctly. The tool should be at least a diameter away from the finished cut before the entry move is made.

9. Mach 2 G- and M-code language reference

This section defines the language (G-codes etc.) that are understood and interpreted by Mach3.
Certain functionality which was defined for machines in the NIST NMC (Next Generation Controller) architecture but is not presently implemented by Mach3 is given in grey type in this chapter.

9.1 Some definitions

9.1.1 Linear Axes

The X, Y, and Z axes form a standard right-handed coordinate system of orthogonal linear axes. Positions of the three linear motion mechanisms are expressed using coordinates on these axes.

9.1.2 Rotational Axes

The rotational axes are measured in degrees as wrapped linear axes in which the direction of positive rotation is counter clockwise when viewed from the positive end of the corresponding X, Y, or Z-axis. By "wrapped linear axis," we mean one on which the angular position increases without limit (goes towards plus infinity) as the axis turns counter clockwise and decreases without limit (goes towards minus infinity) as the axis turns clockwise. Wrapped linear axes are used regardless of whether or not there is a mechanical limit on rotation.

Clockwise or counter clockwise is from the point of view of the workpiece. If the workpiece is fastened to a turntable which turns on a rotational axis, a counter clockwise turn from the point of view of the workpiece is accomplished by turning the turntable in a direction that (for most common machine configurations) looks clockwise from the point of view of someone standing next to the machine.

9.1.3 Scaling input

It is possible to set up scaling factors for each axis. These will be applied to the values of X, Y, Z, A, B, C, I, J and R words whenever these are entered. This allows the size of features machined to be altered and mirror images to be created - by use of negative scale factors.

The scaling is the first thing done with the values and things like feed rate are always based on the scaled values.

The offsets stored in tool and fixture tables are not scaled before use. Scaling may, of course, have been applied at the time the values were entered (say using G10).

9.1.4 Controlled Point

The controlled point is the point whose position and rate of motion are controlled. When the tool length offset is zero (the default value), this is a point on the spindle axis (often called the gauge point) that is some fixed distance beyond the end of the spindle, usually near the end of a tool holder that fits into the spindle. The location of the controlled point can be moved out along the spindle axis by specifying some positive amount for the tool length offset. This amount is normally the length of the cutting tool in use, so that the controlled point is at the end of the cutting tool.

9.1.5 Co-ordinated Linear Motion

To drive a tool along a specified path, a machining system must often co-ordinate the motion of several axes. We use the term "co-ordinated linear motion" to describe the situation in which, nominally, each axis moves at constant speed and all axes move from their starting positions to their end positions at the same time. If only the X, Y, and Z axes (or any one or two of them) move, this produces motion in a straight line, hence the word "linear" in the term. In actual motions, it is often not possible to maintain constant speed because acceleration or deceleration is required at the beginning and/or end of the motion. It is feasible, however, to control the axes so that, at all times, each axis has completed the same fraction of its required motion as the other axes. This moves the tool along the same path, and we also call this kind of motion co-ordinated linear motion.

Co-ordinated linear motion can be performed either at the prevailing feed rate, or at rapid traverse rate. If physical limits on axis speed make the desired rate unobtainable, all axes are slowed to maintain the desired path.

9.1.6 Feed Rate

The rate at which the controlled point or the axes move is nominally a steady rate which may be set by the user. In the Interpreter, the interpretation of the feed rate is as follows unless inverse time feed rate (G93) mode is being used:

- For motion involving one or more of the linear axes (X, Y, Z and optionally A, B, C), without simultaneous rotational axis motion, the feed rate means length units per minute along the programmed linear XYZ(ABC) path
- For motion involving one or more of the linear axes (X, Y, Z and optionally A, B, C), with simultaneous rotational axis motion, the feed rate means length units per minute along the programmed linear XYZ(ABC) path combined with the angular velocity of the rotary axes multiplied by the appropriate axis Correction Diameter multiplied by pi ($\pi =$ 3.14152...); i.e. the declared "circumference" of the part
- For motion of one rotational axis with X, Y, and Z axes not moving, the feed rate means degrees per minute rotation of the rotational axis.
- For motion of two or three rotational axes with X, Y, and Z axes not moving, the rate is applied as follows. Let dA, dB, and dC be the angles in degrees through which the A, B, and C axes, respectively, must move. Let D = sqrt (dA2 + dB2 + dC2). Conceptually, D is a measure of total angular motion, using the usual Euclidean metric. Let T be the amount of time required to move through D degrees at the current feed rate in degrees per minute. The rotational axes should be moved in co-ordinated linear motion so that the elapsed time from the start to the end of the motion is T plus any time required for acceleration or deceleration.

9.1.7 Arc Motion

Any pair of the linear axes (XY, YZ, XZ) can be controlled to move in a circular arc in the plane of that pair of axes. While this is occurring, the third linear axis and the rotational axes can be controlled to move simultaneously at effectively a constant rate. As in co-ordinated linear motion, the motions can be co-ordinated so that acceleration and deceleration do not affect the path.

If the rotational axes do not move, but the third linear axis does move, the trajectory of the controlled point is a helix.

The feed rate during arc motion is as described in Feed Rate above. In the case of helical motion, the rate is applied along the helix. Beware as other interpretations are used on other systems.

9.1.8 Dwell

A machining system may be commanded to dwell (i.e., keep all axes unmoving) for a specific amount of time. The most common use of dwell is to break and clear chips or for a spindle to get up to speed. The units in which you specify Dwell are either seconds or Milliseconds depending on the setting on Configure>Logic

9.1.9 Units

Units used for distances along the X, Y, and Z axes may be measured in millimetres or inches. Units for all other quantities involved in machine control cannot be changed. Different quantities use different specific units. Spindle speed is measured in revolutions per minute. The positions of rotational axes are measured in degrees. Feed rates are expressed in current length units per minute or in degrees per minute, as described above.

Warning: We advise you to check very carefully the system's response to changing units while tool and fixture offsets are loaded into the tables, while these offsets are active and/or while a part program is executing.

9.1.10 Current Position

The controlled point is always at some location called the "current position" and Mach3 always knows where that is. The numbers representing the current position are adjusted in the absence of any axis motion if any of several events take place:

- Length units are changed (but see Warning above)
- Tool length offset is changed
- Coordinate system offsets are changed.

9.1.11 Selected Plane

There is always a "selected plane", which must be the XY-plane, the YZ-plane, or the XZ-plane of the machining system. The Z-axis is, of course, perpendicular to the XY-plane, the X-axis to the YZ-plane, and the Y-axis to the XZ-plane.

9.1.12 Tool Table

Zero or one tool is assigned to each slot in the tool table.

9.1.13 Tool Change

Mach3 allows you to implement a procedure to change the tools by hand when required.

9.1.14 Path Control Modes

The machining system may be put into any one of two path control modes: (1) exact stop mode, (2) constant velocity mode. In exact stop mode, the machine stops briefly at the end of each programmed move. In constant velocity mode, sharp corners of the path may be rounded slightly so that the feed rate may be kept up. These modes are to allow the user to control the compromise involved in turning corners because a real machine has a finite acceleration due to the inertia of its mechanism.

Exact stop does what it says. The machine will come to rest at each change of direction and the tool will therefore precisely follow the commanded path.

Constant velocity will overlap acceleration in the new direction with deceleration in the current one in order to keep the commanded feed rate. This implies a rounding of any corner but faster and smoother cutting. The lower the acceleration of the machine axes, the greater will be the radius of the rounded corner.

It is also possible to define a limiting angle so that changes in direction of more than this angle will always be treated as Exact Stop even though Constant Velocity is selected. This allows gentle corners to be smoother but avoids excessive rounding of sharp corners even on machines with low acceleration on one or more axes. This feature is enabled in the *Configure Logic* dialog and the limiting angle is set by a DRO. This setting will probably need to be chosen experimentally depending on the characteristics of the machine tool and, perhaps, the tool path of an individual job.

9.2 Interpreter Interaction with controls

9.2.1 Feed and Speed Override controls

Mach3 commands which enable (M48) or disable (M49) the feed and speed override switches. It is useful to be able to override these switches for some machining operations. The idea is that optimal settings have been included in the program, and the operator should not change them.

9.2.2 Block Delete control

If the block delete control is ON, lines of code which start with a slash (the block delete character) are not executed. If the switch is off, such lines are executed.

9.2.3 Optional Program Stop control

The optional program stop control (see Configure>Logic) works as follows. If this control is ON and an input line contains an M1 code, program execution is stopped at the end on the commands on that line until the *Cycle Start* button is pushed.

9.3 Tool File

Mach3 maintains a tool file for each of the 254 tools which can be used.

Each data line of the file contains the data for one tool. This allows the definition of the tool length (Z axis) and tool diameter.

9.4 The language of part programs

9.4.1 Overview

The language is based on lines of code. Each line (also called a "block") may include commands to the machining system to do several different things. Lines of code may be collected in a file to make a program.

A typical line of code consists of an optional line number at the beginning followed by one or more "words." A word consists of a letter followed by a number (or something that evaluates to a number). A word may either give a command or provide an argument to a command. For example, `G1 X3` is a valid line of code with two words. "G1" is a command meaning "move in a straight line at the programmed feed rate," and "X3" provides an argument value (the value of X should be 3 at the end of the move). Most commands start with either G or M (for General and Miscellaneous). The words for these commands are called "G codes" and "M codes."

The language has two commands (M2 or M30), either of which ends a program. A program may end before the end of a file. Lines of a file that occur after the end of a program are not to be executed in the normal flow so will generally be parts of subroutines.

9.4.2 Parameters

A Mach3 machining system maintains an array of 10,320 numerical parameters. Many of them have specific uses. The parameters which are associated with fixtures are persistent over time. Other parameters will be undefined when Mach3 is loaded. The parameters are preserved when the interpreter is reset. The parameters with meanings defined by Mach3 are given in figure 9.1

Parameter Number	Meaning	Parameter Number	Meaning
5161	G28 home X	5261	Work offset 3 X
5162	G28 home Y	5262	Work offset 3 Y
5163	G28 home Z	5263	Work offset 3 Z
5164	G28 home A	5264	Work offset 3 A
5165	G28 home B	5265	Work offset 3 B
5166	G28 home C	5266	Work offset 3 C
5181	G30 home X	5281	Work offset 4 X
5182	G30 home Y	5282	Work offset 4 Y
5183	G30 home Z	5283	Work offset 4 Z
5184	G30 home A	5284	Work offset 4 A
5185	G30 home B	5285	Work offset 4 B
5186	G30 home C	5286	Work offset 4 C
5191	Scale X	5301	Work offset 5 X
5192	Scale Y	5302	Work offset 5 Y
5193	Scale Z	5303	Work offset 5 Z
5194	Scale A	5304	Work offset 5 A
5195	Scale B	5305	Work offset 5 B
5196	Scale C	5306	Work offset 5 C
5211	G92 offset X	5321	Work offset 6 X
5212	G92 offset Y	5322	Work offset 6 Y
5213	G92 offset Z	5323	Work offset 6 Z
5214	G92 offset A	5324	Work offset 6 A
5215	G92 offset B	5325	Work offset 6 B
5216	G92 offset C	5326	Work offset 6 C
5220	Current Work offset number	...	And so on every 20 values until
5221	Work offset 1 X	...	
5222	Work offset 1 Y	...	
5223	Work offset 1 Z		
5224	Work offset 1 A	10281	Work offset 254 X
5225	Work offset 1 B	10282	Work offset 254 Y
5226	Work offset 1 C	10283	Work offset 254 Z
5241	Work offset 2 X	10284	Work offset 254 A
5242	Work offset 2 Y	10285	Work offset 254 B
5243	Work offset 2 Z	10286	Work offset 254 C
5244	Work offset 2 A	10301	Work offset 255 X
5245	Work offset 2 B	10302	Work offset 255 Y
5246	Work offset 2 C	10303	Work offset 255 Z
		10304	Work offset 255 A
		10305	Work offset 255 B
		10306	Work offset 255 C

Figure 9.1 - System defined parameters

9.4.3 Coordinate Systems

The machining system has an absolute coordinate system and 254 work offset (fixture) systems.

You can set the offsets of tools by `G10 L1 P~ X~ Z~`. The P word defines the tool offset number to be set.

You can set the offsets of the fixture systems using `G10 L2 P~ X~ Y~ Z~ A~ B~ C~` The P word defines the fixture to be set. The X, Y, Z etc words are the coordinates for the origin of the axes in terms of the absolute coordinate system.

You can select one of the first seven work offsets by using G54, G55, G56, G57, G58 and G59. Any of the 255 work offsets can be selected by `G59 P~` (e.g. `G59 P23` would select fixture 23). The absolute coordinate system can be selected by `G59 P0`.

You can offset the current coordinate system using G92 or G92.3. This offset will then be applied on top of work offset coordinate systems. This offset may be cancelled with G92.1 or G92.2.

You can make straight moves in the absolute machine coordinate system by using G53 with either G0 or G1.

9.5 Format of a Line

A permissible line of input code consists of the following, in order, with the restriction that there is a maximum (currently 256) to the number of characters allowed on a line.

- An optional block delete character, which is a slash "/".
- An optional line number.
- Any number of words, parameter settings, and comments.
- An end of line marker (carriage return or line feed or both).

Any input not explicitly allowed is illegal and will cause the Interpreter to signal an error or to ignore the line.

Spaces and tabs are allowed anywhere on a line of code and do not change the meaning of the line, except inside comments. This makes some strange-looking input legal. For example, the line `g0x +0. 12 34y 7` is equivalent to `g0 x+0.1234 y7`

Blank lines are allowed in the input. They will be ignored.

Input is case insensitive, except in comments, i.e., any letter outside a comment may be in upper or lower case without changing the meaning of a line.

9.5.1 Line Number

A line number is the letter N followed by an integer (with no sign) between 0 and 99999 written with no more than five digits (000009 is not OK, for example). Line numbers may be repeated or used out of order, although normal practice is to avoid such usage. A line number is not required to be used (and this omission is common) but it must be in the proper place if it is used.

9.5.2 Subroutine labels

A subroutine label is the letter O followed by an integer (with no sign) between 0 and 99999 written with no more than five digits (000009 is not permitted, for example). Subroutine labels may be used in any order but must be unique in a program although violation of this rule may not be flagged as an error. Nothing else except a comment should appear on the same line after a subroutine label.

9.5.3 Word

A word is a letter other than N or O followed by a real value.

Words may begin with any of the letters shown in figure 9.2. The table includes N and O for completeness, even though, as defined above, line numbers are not words. Several letters (I, J, K, L, P and R) may have different meanings in different contexts.

A real value is some collection of characters that can be processed to come up with a number. A real value may be an explicit number (such as 341 or -0.8807), a parameter value, an expression, or a unary operation value. Definitions of these follow immediately. Processing characters to come up with a number is called "evaluating". An explicit number evaluates to itself.

Letter	Meaning
A	A-axis of machine
B	B-axis of machine
C	C-axis of machine
D	Tool radius compensation number
F	Feed rate
G	General function (see Table 5)
H	Tool length offset index
I	X-axis offset for arcs X offset in G87 canned cycle
J	Y-axis offset for arcs Y offset in G87 canned cycle
K	Z-axis offset for arcs Z offset in G87 canned cycle
L	Number of repetitions in canned cycles/ subroutines Key used with G10
M	Miscellaneous function (see Table 7)
N	Line number
O	Subroutine label number
P	Dwell time in canned cycles Dwell time with G4 Key used with G10
Q	Feed increment in G83 canned cycle Repetitions of subroutine call
R	Arc radius Canned cycle retract level
S	Spindle speed
T	Tool selection
U	Synonymous with A
V	Synonymous with B
W	Synonymous with C
X	X-axis of machine
Y	Y-axis of machine
Z	Z-axis of machine

Figure 9.2 - Word initial letters

9.5.3.1 Number

The following rules are used for (explicit) numbers. In these rules a digit is a single character between 0 and 9.

- A number consists of (1) an optional plus or minus sign, followed by (2) zero to many digits, followed, possibly, by (3) one decimal point, followed by (4) zero to many digits - provided that there is at least one digit somewhere in the number.
- There are two kinds of numbers: integers and decimals. An integer does not have a decimal point in it; a decimal does.
- Numbers may have any number of digits, subject to the limitation on line length. Only about seventeen significant figures will be retained, however (enough for all known applications).
- A non-zero number with no sign as the first character is assumed to be positive.

Notice that initial (before the decimal point and the first non-zero digit) and trailing (after the decimal point and the last non-zero digit) zeros are allowed but not required. A number written with initial or trailing zeros will have the same value when it is read as if the extra zeros were not there.

Numbers used for specific purposes by Mach3 are often restricted to some finite set of values or some to some range of values. In many uses, decimal numbers must be close to integers; this includes the values of indexes (for parameters and carousel slot numbers, for example), M codes, and G codes multiplied by ten. A decimal number which is supposed to be close to an integer is considered close enough if it is within 0.0001 of an integer.

9.5.3.2 Parameter Value

A parameter value is the hash character # followed by a real value. The real value must evaluate to an integer between 1 and 10320. The integer is a parameter number, and the value of the parameter value is whatever number is stored in the numbered parameter.

The # character takes precedence over other operations, so that, for example, `#1+2` means the number found by adding 2 to the value of parameter 1, not the value found in parameter 3. Of course, `#[1+2]` does mean the value found in parameter 3. The # character may be repeated; for example `##2` means the value of the parameter whose index is the (integer) value of parameter 2.

9.5.3.3 Expressions and Binary Operations

An expression is a set of characters starting with a left bracket [and ending with a balancing right bracket]. In between the brackets are numbers, parameter values, mathematical operations, and other expressions. An expression may be evaluated to produce a number. The expressions on a line are evaluated when the line is read, before anything on the line is executed. An example of an expression is:

```
[1+acos[0]-[#3**[4.0/2]]]
```

Binary operations appear only inside expressions. Nine binary operations are defined. There are four basic mathematical operations: addition (+), subtraction (-), multiplication (*), and division (/). There are three logical operations: non-exclusive or (OR), exclusive or (XOR), and logical and (AND). The eighth operation is the modulus operation (MOD). The ninth operation is the "power" operation (**) of raising the number on the left of the operation to the power on the right.

The binary operations are divided into three groups. The first group is: power. The second group is: multiplication, division, and modulus. The third group is: addition, subtraction, logical non-exclusive or, logical exclusive or, and logical and. If operations are strung together (for example in the expression `[2.0/3*1.5-5.5/11.0]`), operations in the first group are to be performed before operations in the second group and operations in the second group before operations in the third group. If an expression contains more than one operation from the same group (such as the first / and * in the example), the operation on the left is performed first. Thus, the example is equivalent to: `[((2.0/3)*1.5)-(5.5/11.0)]` which simplifies to `[1.0-0.5]` which is 0.5.

The logical operations and modulus are to be performed on any real numbers, not just on integers. The number zero is equivalent to logical false, and any non-zero number is equivalent to logical true.

9.5.3.4 Unary Operation Value

A unary operation value is either "ATAN" followed by one expression divided by another expression (for example `ATAN[2]/[1+3]`) or any other unary operation name followed by an expression (for example `SIN[90]`). The unary operations are: ABS (absolute value), ACOS (arc cosine), ASIN (arc sine), ATAN (arc tangent), COS (cosine), EXP (e raised to the given power), FIX (round down), FUP (round up), LN (natural logarithm), ROUND (round to the nearest whole number), SIN (sine), SQRT (square root), and TAN (tangent). Arguments to unary operations which take angle measures (COS, SIN, and TAN) are in degrees. Values returned by unary operations which return angle measures (ACOS, ASIN, and ATAN) are also in degrees.

The FIX operation rounds towards the left (less positive or more negative) on a number line, so that `FIX[2.8]=2` and `FIX[-2.8]=-3`, for example. The FUP operation rounds towards the right (more positive or less negative) on a number line; `FUP[2.8]=3` and `FUP[-2.8]=-2`, for example.

9.5.4 Parameter Setting

A parameter setting is the following four items one after the other:

- a pound character #
- a real value which evaluates to an integer between 1 and 10320,
- an equal sign = , and
- a real value in square brackets [ ]. For example "#3 = [15]" is a parameter setting meaning "set parameter 3 to 15."

A parameter setting does not take effect until after all parameter values on the same line have been found. For example, if parameter 3 has been previously set to 15 and the line `#3=6 G1 x#3` is interpreted, a straight move to a point where x equals 15 will occur and the value of parameter 3 will be 6.

9.5.5 Comments and Messages

A line that starts with the percent character, `%`, is treated as a comment and not interpreted in any way.

Printable characters and white space inside parentheses is a comment. A left parenthesis always starts a comment. The comment ends at the first right parenthesis found thereafter. Once a left parenthesis is placed on a line, a matching right parenthesis must appear before the end of the line. Comments may not be nested; it is an error if a left parenthesis is found after the start of a comment and before the end of the comment. Here is an example of a line containing a comment: `G80 M5 (stop motion)`

An alternative form of comment is to use the semicolon ; character. The remainder of the line is treated as a comment.

Comments do not cause the machining system to do anything.

A comment that is included in parentheses, contains a message if `MSG,` appears after the left parenthesis and before any other printing characters. Variants of `MSG,` which include white space and lower case characters, are allowed. Note that the comma is required. The rest of the characters before the right parenthesis are considered to be a message to the operator. Messages are displayed on screen in the "Error" intelligent label.

9.5.6 Item Repeats

A line may have any number of G words, but two G words from the same modal group may not appear on the same line.

A line may have zero to four M words. Two M words from the same modal group may not appear on the same line.

For all other legal letters, a line may have only one word beginning with that letter.

If a parameter setting of the same parameter is repeated on a line, `#3=15 #3=6`, for example, only the last setting will take effect. It is silly, but not illegal, to set the same parameter twice on the same line.

If more than one comment appears on a line, only the last one will be used; each of the other comments will be read and its format will be checked, but it will be ignored thereafter. It is expected that putting more than one comment on a line will be very rare.

9.5.7 Item order

The three types of item whose order may vary on a line (as given at the beginning of this section) are word, parameter setting, and comment. Imagine that these three types of item are divided into three groups by type.

The first group (the words) may be reordered in any way without changing the meaning of the line.

If the second group (the parameter settings) is reordered, there will be no change in the meaning of the line unless the same parameter is set more than once. In this case, only the last setting of the parameter will take effect. For example, after the line `#3=15 #3=6` has been interpreted, the value of parameter 3 will be 6. If the order is reversed to `#3=6 #3=15` and the line is interpreted, the value of parameter 3 will be 15.

If the third group (the comments) contains more than one comment and is reordered, only the last comment will be used.

If each group is kept in order or reordered without changing the meaning of the line, then the three groups may be interleaved in any way without changing the meaning of the line. For example, the line `g40 g1 #3=15 (so there!) #4=-7.0` has five items and means exactly the same thing in any of the 120 possible orders - such as `#4=-7.0 g1 #3=15 g40 (so there!)` - for the five items.

9.5.8 Commands and Machine Modes

Mach3 has many commands which cause a machining system to change from one mode to another, and the mode stays active until some other command changes it implicitly or explicitly. Such commands are called "modal". For example, if spindle is turned on, it stays on until it is explicitly turned off. The G codes for motion are also modal. If a G1 (straight move) command is given on one line, for example, it will be executed again on the next line if one or more axis words is available on the line, unless an explicit command is given on that next line using the axis words or cancelling motion.

"Non-modal" codes have effect only on the lines on which they occur. For example, G4 (dwell) is non-modal.

9.6 Modal Groups

Modal commands are arranged in sets called "modal groups", and only one member of a modal group may be in force at any given time. In general, a modal group contains commands for which it is logically impossible for two members to be in effect at the same time - like measure in inches vs. measure in millimetres. A machining system may be in many modes at the same time, with one mode from each modal group being in effect. The modal groups are shown in figure 9.3.

The modal Groups for G codes are

- group 1 = {G00, G01, G02, G03, G38.2, G80, G81, G82, G84, G85, G86, G87, G88, G89} motion
- group 2 = {G17, G18, G19} plane selection
- group 3 = {G90, G91} distance mode
- group 5 = {G93, G94} feed rate mode
- group 6 = {G20, G21} units
- group 7 = {G40, G41, G42} cutter radius compensation
- group 8 = {G43, G49} tool length offset
- group 10 = {G98, G99} return mode in canned cycles
- group 12 = {G54, G55, G56, G57, G58, G59, G59.xxx} coordinate system selection
- group 13 = {G61, G61.1, G64} path control mode

The modal groups for M codes are:

- ♦ group 4 = {M0, M1, M2, M30} stopping
- ♦ group 6 = {M6} tool change
- ♦ group 7 = {M3, M4, M5} spindle turning
- ♦ group 8 = {M7, M8, M9} coolant (special case: M7 and M8 may be active at the same time)
- ♦ group 9 = {M48, M49} enable/disable feed and speed override controls

In addition to the above modal groups, there is a group for non-modal G codes:

- ♦ group 0 = {G4, G10, G28, G30, G53, G92, G92.1, G92.2, G92.3}

Figure 9.3 - Modal groups

For several modal groups, when a machining system is ready to accept commands, one member of the group must be in effect. There are default settings for these modal groups. When the machining system is turned on or otherwise re-initialized, the default values are automatically in effect.

Group 1, the first group on the table, is a group of G codes for motion. One of these is always in effect. That one is called the current motion mode.

It is an error to put a G-code from group 1 and a G-code from group 0 on the same line if both of them use axis words. If an axis word-using G-code from group 1 is implicitly in effect on a line (by having been activated on an earlier line), and a group 0 G-code that uses axis words appears on the line, the activity of the group 1 G-code is suspended for that line. The axis word-using G-codes from group 0 are G10, G28, G30, and G92.

Mach3 displays the current mode at the top of each screen.

9.7 G Codes

G codes of the Mach3 input language are shown in figure 9.4 and are the described in detail.

The descriptions contain command prototypes, set in `courier` type.

Summary of G-codes	
G0	Rapid positioning
G1	Linear interpolation
G2	Clockwise circular/helical interpolation
G3	Counter clockwise circular/Helical interpolation
G4	Dwell
G10	Coordinate system origin setting
G12	Clockwise circular pocket
G13	Counter clockwise circular pocket
G15/G16	Polar Coordinate moves in G0 and G1
G17	XY Plane select
G18	XZ plane select
G19	YZ plane select
G20/G21	Inch/Millimetre unit
G28	Return home
G28.1	Reference axes
G30	Return home
G31	Straight probe
G40	Cancel cutter radius compensation
G41/G42	Start cutter radius compensation left/right
G43	Apply tool length offset (plus)
G49	Cancel tool length offset
G50	Reset all scale factors to 1.0
G51	Set axis data input scale factors
G52	Temporary coordinate system offsets
G53	Move in absolute machine coordinate system
G54	Use fixture offset 1
G55	Use fixture offset 2
G56	Use fixture offset 3
G57	Use fixture offset 4
G58	Use fixture offset 5
G59	Use fixture offset 6 / use general fixture number
G61/G64	Exact stop/Constant Velocity mode
G68/G69	Rotate program coordinate system
G70/G71	Inch/Millimetre unit
G73	Canned cycle - peck drilling
G80	Cancel motion mode (including canned cycles)
G81	Canned cycle - drilling
G82	Canned cycle - drilling with dwell
G83	Canned cycle - peck drilling
G84	Canned cycle - right hand rigid tapping. Do not use on KX mill
G85/G86/ G88/G89	Canned cycle – boring
G90	Absolute distance mode
G91	Incremental distance mode
G92	Offset coordinates and set parameters
G92.x	Cancel G92 etc.
G93	Inverse time feed mode
G94	Feed per minute mode
G95	Feed per rev mode
G98	Initial level return after canned cycles
G99	R-point level return after canned cycles

Figure 9.4 - Table of G codes

In the command prototypes, the tilde (~) stand for a real value. As described earlier, a real value may be (1) an explicit number, 4.4, for example, (2) an expression, [2+2.4], for example, (3) a parameter value, #88, for example, or (4) a unary function value, acos[0], for example.

In most cases, if axis words (any or all of X~, Y~, Z~, A~, B~, C~, U~, V~, W~) are given, they specify a destination point. Axis numbers relate to the currently active coordinate system, unless explicitly described as being in the absolute coordinate system. Where axis words are optional, any omitted axes will have their current value. Any items in the command prototypes not explicitly described as optional are required. It is an error if a required item is omitted.

U, V and W are synonyms for A, B and C. Use of A with U, B with V etc. is erroneous (like using A twice on a line). In the detailed descriptions of codes U, V and W are not explicitly mentioned each time but are implied by A, B or C.

In the prototypes, the values following letters are often given as explicit numbers. Unless stated otherwise, the explicit numbers can be real values. For example, `G10 L2` could equally well be written `G[2*5] L[1+1]`. If the value of parameter 100 were 2, `G10 L#100` would also mean the same. Using real values which are not explicit numbers as just shown in the examples is rarely useful.

If `L~` is written in a prototype the "~" will often be referred to as the "L number". Similarly the "~" in `H~` may be called the "H number", and so on for any other letter.

If a scale factor is applied to any axis then it will be applied to the value of the corresponding X, Y, Z, A/U, B/V, C/W word and to the relevant I, J, K or R words when they are used.

9.7.1 Rapid Linear Motion - G0

(a) For rapid linear motion, program `G0 X~ Y~ Z~ A~ B~ C~`, where all the axis words are optional, except that at least one must be used. The G0 is optional if the current motion mode is G0. This will produce co-ordinated linear motion to the destination point at the current traverse rate (or slower if the machine will not go that fast). It is expected that cutting will not take place when a G0 command is executing.

(b) If G16 has been executed to set a Polar Origin then for rapid linear motion to a point described by a radius and angle `G0 X~ Y~` can be used. `X~` is the radius of the line from the G16 polar origin and `Y~` is the angle in degrees measured with increasing values counter clockwise from the 3 o'clock direction (i.e. the conventional four quadrant conventions).

Coordinates of the current point at the time of executing the G16 are the polar origin.

It is an error if:

- all axis words are omitted.

If cutter radius compensation is active, the motion will differ from the above; see Cutter Compensation. If G53 is programmed on the same line, the motion will also differ; see Absolute Coordinates.

9.7.2 Linear Motion at Feed Rate - G1

(a) For linear motion at feed rate (for cutting or not), program `G1 X~ Y~ Z~ A~ B~ C~`, where all the axis words are optional, except that at least one must be used. The G1 is optional if the current motion mode is G1. This will produce co-ordinated linear motion to the destination point at the current feed rate (or slower if the machine will not go that fast).

(b) If G16 has been executed to set a polar origin then linear motion at feed rate to a point described by a radius and angle `G0 X~ Y~` can be used. `X~` is the radius of the line from the G16 polar origin and `Y~` is the angle in degrees measured with increasing values counter clockwise from the 3 o'clock direction (i.e. the conventional four quadrant conventions).

Coordinates of the current point at the time of executing the G16 are the polar origin.

It is an error if:

- all axis words are omitted.

If cutter radius compensation is active, the motion will differ from the above; see Cutter Compensation. If G53 is programmed on the same line, the motion will also differ; see Absolute Coordinates.

9.7.3 Arc at Feed Rate - G2 and G3

A circular or helical arc is specified using either G2 (clockwise arc) or G3 (counter clockwise arc). The axis of the circle or helix must be parallel to the X, Y, or Z-axis of the machine coordinate system. The axis (or, equivalently, the plane perpendicular to the axis) is selected with G17 (Z-axis, XY-plane), G18 (Y-axis, XZ-plane), or G19 (X-axis, YZ-plane). If the arc is circular, it lies in a plane parallel to the selected plane.

If a line of code makes an arc and includes rotational axis motion, the rotational axes turn at a constant rate so that the rotational motion starts and finishes when the XYZ motion starts and finishes. Lines of this sort are hardly ever programmed.

If cutter radius compensation is active, the motion will differ from the above; see Cutter Compensation.

Two formats are allowed for specifying an arc. We will call these the centre format and the radius format. In both formats the G2 or G3 is optional if it is the current motion mode.

9.7.3.1 Radius Format Arc

In the radius format, the coordinates of the end point of the arc in the selected plane are specified along with the radius of the arc. Program `G2 X~ Y~ Z~ A~ B~ C~ R~` (or use G3 instead of G2). R is the radius. The axis words are all optional except that at least one of the two words for the axes in the selected plane must be used. The R number is the radius. A positive radius indicates that the arc turns through 180 degrees or less, while a negative radius indicates a turn of 180 degrees to 359.999 degrees. If the arc is helical, the value of the end point of the arc on the coordinate axis parallel to the axis of the helix is also specified.

It is an error if:

- both of the axis words for the axes of the selected plane are omitted,
- the end point of the arc is the same as the current point.

It is not good practice to program radius format arcs that are nearly full circles or are semicircles (or nearly semicircles) because a small change in the location of the end point will produce a much larger change in the location of the centre of the circle (and, hence, the middle of the arc). The magnification effect is large enough that a rounding error in a number can produce out-of-tolerance cuts. Nearly full circles are outrageously bad, semicircles (and nearly so) are only very bad. Other size arcs (in the range tiny to 165 degrees or 195 to 345 degrees) are OK.

Here is an example of a radius format command to mill an arc:

```
G17 G2 x 10 y 15 r 20 z 5.
```

That means to make a clockwise (as viewed from the positive Z-axis) circular or helical arc whose axis is parallel to the Z-axis, ending where X=10, Y=15, and Z=5, with a radius of 20. If the starting value of Z is 5, this is an arc of a circle parallel to the XY-plane; otherwise it is a helical arc.

9.7.3.2 Centre Format Arc

In the centre format, the coordinates of the end point of the arc in the selected plane are specified along with the offsets of the centre of the arc from the current location. In this format, it is OK if the end point of the arc is the same as the current point. It is an error if:

- when the arc is projected on the selected plane, the distance from the current point to the centre differs from the distance from the end point to the centre by more than 0.0002 inch (if inches are being used) or 0.002 millimetre (if millimetres are being used).

The centre is specified using the I and J words. There are two ways of interpreting them. The usual way is that I and J are the centre relative to the current point at the start of the arc. This is sometimes called *Incremental IJ mode.* The second way is that I and J specify the centre as actual coordinates in the current system. This is rather misleadingly called *Absolute IJ mode*. The IJ mode is set using the Configure>State… menu when Mach3 is set up. The choices of modes are to provide compatibility with commercial controllers. You will probably find Incremental to be best. In Absolute it will, of course usually be necessary to use both I and J words unless by chance the arc's centre is at the origin.

When the XY-plane is selected, program `G2 X~ Y~ Z~ A~ B~ C~ I~ J~` (or use G3 instead of G2). The axis words are all optional except that at least one of X and Y must be used. I and J are the offsets from the current location or coordinates - depending on IJ mode (X and Y directions, respectively) of the centre of the circle. I and J are optional except that at least one of the two must be used. It is an error if:

- X and Y are both omitted,
- I and J are both omitted.

When the XZ-plane is selected, program `G2 X~ Y~ Z~ A~ B~ C~ I~ K~` (or use G3 instead of G2). The axis words are all optional except that at least one of X and Z must be used. I and K are the offsets from the current location or coordinates - depending on IJ mode (X and Z directions, respectively) of the centre of the circle. I and K are optional except that at least one of the two must be used. It is an error if:

- X and Z are both omitted,
- I and K are both omitted.

When the YZ-plane is selected, program `G2 X~ Y~ Z~ A~ B~ C~ J~ K~` (or use G3 instead of G2). The axis words are all optional except that at least one of Y and Z must be used. J and K are the offsets from the current location or coordinates - depending on IJ mode (Y and Z directions, respectively) of the centre of the circle. J and K are optional except that at least one of the two must be used. It is an error if:

- Y and Z are both omitted,
- J and K are both omitted.

Here is an example of a centre format command to mill an arc in Incremental IJ mode:

```
G17 G2 x10 y16 i3 j4 z9
```

That means to make a clockwise (as viewed from the positive z-axis) circular or helical arc whose axis is parallel to the Z-axis, ending where X=10, Y=16, and Z=9, with its centre offset in the X direction by 3 units from the current X location and offset in the Y direction by 4 units from the current Y location. If the current location has X=7, Y=7 at the outset, the centre will be at X=10, Y=11. If the starting value of Z is 9, this is a circular arc; otherwise it is a helical arc. The radius of this arc would be 5.

The above arc in Absolute IJ mode would be:

```
G17 G2 x10 y16 i10 j11 z9
```

In the centre format, the radius of the arc is not specified, but it may be found easily as the distance from the centre of the circle to either the current point or the end point of the arc.

9.7.4 Dwell - G4

For a dwell, program `G4 P~` . This will keep the axes unmoving for the period of time in seconds or milliseconds specified by the P number. The time unit to be used is set up on the Config>Logic dialog. For example, with units set to Seconds, G4 P0.5 will dwell for half a second. It is an error if:

- the P number is negative.

9.7.5 Set Coordinate System Data Tool and work offset tables - G10

See details of tool and work offsets for further information on coordinate systems

To set the offset values of a tool, program
`G10 L1 P~ X~ Z~` A~, where the P number must evaluate to an integer in the range 0 to 255 - the tool number - Offsets of the tool specified by the P number are reset to the given. The A number will reset the tool tip radius. Only those values for which an axis word is included on the line will be reset. The Tool diameter cannot be set in this way.

To set the coordinate values for the origin of a fixture coordinate system, program
`G10 L2 P~ X~ Y~ Z~ A~ B~ C~`, where the P number must evaluate to an integer in the range 1 to 255 - the fixture number - (Values 1 to 6 corresponding to G54 to G59) and all axis words are optional. The coordinates of the origin of the coordinate system specified by the P number are reset to the coordinate values given (in terms of the absolute coordinate system). Only those coordinates for which an axis word is included on the line will be reset.

It is an error if:

- the P number does not evaluate to an integer in the range 0 to 255.

If origin offsets (made by G92 or G92.3) were in effect before G10 is used, they will continue to be in effect afterwards.

The coordinate system whose origin is set by a G10 command may be active or inactive at the time the G10 is executed.

The values set will not be persistent unless the tool or fixture tables are saved using the buttons on Tables screen.

Example: `G10 L2 P1 x3.5 y17.2` sets the origin of the first coordinate system (the one selected by G54) to a point where X is 3.5 and Y is 17.2 (in absolute coordinates). The Z coordinate of the origin (and the coordinates for any rotational axes) are whatever those coordinates of the origin were before the line was executed.

9.7.6 Clockwise/counter clockwise circular pocket - G12 and G13

These circular pocket commands are a sort of canned cycle, which can be used to produce a circular hole larger than the tool in use or with a suitable tool (like a woodruff key cutter) to cut internal grooves for “O” rings etc.

Program `G12 I~` for a clockwise move and `G13 I~` for a counter clockwise move.

The tool is moved in the X direction by the value of the I word and a circle cut in the direction specified with the original X and Y coordinates as the centre. The tool is returned to the centre.

Its effect is undefined if the current plane is not XY.

9.7.7 Exit and Enter Polar mode - G15 and G16

It is possible for G0 and G1 moves in the X/Y plane only, to specify coordinates as a radius and angle relative to a temporary centre point. Program `G16` to enter this mode. The current coordinates of the controlled point are the temporary centre.

Program G15 to revert to normal Cartesian coordinates.

```
G0 X10 Y10          // normal G0 move to 10,10
G16 //start of polar mode.
G10X10Y45
(this will move to X 17.xxx, Y 17.xxx which is a spot on
a circle) (of radius 10 at 45 degrees from the initial
coordinates of 10,10.)
```

This can be very useful, for example, for drilling a circle of holes. The code below moves to a circle of holes every 10 degrees on a circle of radius 50 mm centre X = 10, Y = 5.5 and peck drills to Z = -0.6

```
G21        // metric
G0 X10Y5.5
G16
G1 X50 Y0      //polar move to a radius of 50 angle 0deg
G83 Z-0.6      // peck drill
G1 Y10         // ten degrees from original center...
G83 Z-0.6
G1 Y20         // 20 degrees....etc...

G1 Y30

G1 Y40
> ...etc....
G15            //back to normal cartesian
```

Notes:

(1) You must not make X or Y moves other than by using G0 or G1 when G16 is active

(2) This G16 is different to a Fanuc implementation in that it uses the current point as the polar centre. The Fanuc version requires a lot of origin shifting to get the desired result for any circle not centred on 0,0

9.7.8 Plane Selection - G17, G18, and G19

Program G17 to select the XY-plane, G18 to select the XZ-plane, or G19 to select the YZ-plane. The effects of having a plane selected are discussed in G2/3 and Canned cycles.

9.7.9 Length Units - G20 and G21

Program G20 to use inches for length units. Program G21 to use millimetres.

It is usually a good idea to program either G20 or G21 near the beginning of a program before any motion occurs and not to use either one anywhere else in the program. It is the responsibility of the user to be sure all numbers are appropriate for use with the current length units. See also G70/G71 which are synonymous.

9.7.10 Return to Home - G28 and G30

A home position is defined (by parameters 5161-5166). The parameter values are in terms of the absolute coordinate system, but are in unspecified length units.

To return to home position by way of the programmed position, program `G28 X~ Y~ Z~ A~ B~ C~` (or use G30). All axis words are optional. The path is made by a traverse move from the current position to the programmed position, followed by a traverse move to the home position. If no axis words are programmed, the intermediate point is the current point and only one move is made.

9.7.11 Reference axes G28.1

Program `G28.1 X~ Y~ Z~ A~ B~ C~` to reference the given axes. The axes will move at the current feed rate towards the home switch (es), as defined by the Configuration. When the absolute machine coordinate reaches the value given by an axis word then the feed rate is set to that defined by Configure>Config Referencing. Provided the current absolute position is approximately correct, then this will give a soft stop onto the reference switch (es).

9.7.12 Cutter Radius Compensation - G40, G41, and G42

To turn cutter radius compensation off, program `G40`. It is OK to turn compensation off when it is already off.

Cutter radius compensation may be performed only if the XY plane is active.

To turn cutter radius compensation on left (i.e., the cutter stays to the left of the programmed path when the tool radius is positive), program `G41 D~`. To turn cutter radius compensation on right (i.e., the cutter stays to the right of the programmed path when the tool radius is positive), program `G42 D~` The D word is optional; if there is no D word, the radius of the tool currently in the spindle will be used. If used, the D number should normally be the slot number of the tool in the spindle, although this is not required. It is OK for the D number to be zero; a radius value of zero will be used.

G41 and G42 can be qualified by a P-word. This will override the value of the diameter of the tool (if any) given in the current tool table entry.

It is an error if:

- The D number is not an integer, is negative or is larger than the number of carousel slots,
- The XY-plane is not active,
- Cutter radius compensation is commanded to turn on when it is already on.

The behaviour of the machining system when cutter radius compensation is ON is described in the chapter of Cutter Compensation. Notice the importance of programming valid entry and exit moves.

9.7.13 Tool Length Offsets - G43, G44 and G49

To use a tool length offset, program `G43 H~`, where the H number is the desired index in the tool table. It is expected that all entries in this table will be positive. The H number should be, but does not have to be, the same as the slot number of the tool currently in the spindle. It is OK for the H number to be zero; an offset value of zero will be used. Omitting H has the same effect as a zero value.

G44 is provided for compatibility and is used if entries in the table give negative offsets.

It is an error if:

- The H number is not an integer, is negative, or is larger than the number of carousel slots.

To use no tool length offset, program `G49`

It is OK to program using the same offset already in use. It is also OK to program using no tool length offset if none is currently being used.

9.7.14 Scale factors G50 and G51

To define a scale factor which will be applied to an X, Y, Z, A, B, C, I & J word before it is used program `G51 X~ Y~ Z~ A~ B~ C~` where the X, Y, Z etc. words are the scale factors for the given axes. These values are, of course, never themselves scaled.

It is not permitted to use unequal scale factors to produce elliptical arcs with G2 or G3.

To reset the scale factors of all axes to 1.0 program `G50`

9.7.15 Temporary Coordinate system offset – G52

To offset the current point by a given positive or negative distance (without motion), program `G52 X~ Y~ Z~ A~ B~ C~`, where the axis words contain the offsets you want to provide. All axis words are optional, except that at least one must be used. If an axis word is not used for a given axis, the coordinate on that axis of the current point is not changed. It is an error if:

- All axis words are omitted.

G52 and G92 use common internal mechanisms in Mach3 and may not be used together.

When G52 is executed, the origin of the currently active coordinate system moves by the values given.

The effect of G52 is cancelled by programming `G52 X0 Y0` etc.

Here is an example. Suppose the current point is at X=4 in the currently specified coordinate system, then `G52 X7` sets the X-axis offset to 7, and so causes the X-coordinate of the current point to be -3.

The axis offsets are always used when motion is specified in absolute distance mode using any of the fixture coordinate systems. Thus all fixture coordinate systems are affected by G52.

9.7.16 Move in Absolute Coordinates - G53

For linear motion to a point expressed in absolute coordinates, program `G1 G53 X~ Y~ Z~ A~ B~ C~` (or similarly with G0 instead of G1), where all the axis words are optional, except that at least one must be used. The G0 or G1 is optional if it is in the current motion mode. G53 is not modal and must be programmed on each line on which it is intended to be active. This will produce co-ordinated linear motion to the programmed point. If G1 is active, the speed of motion is the current feed rate (or slower if the machine will not go that fast). If G0 is active, the speed of motion is the current traverse rate (or slower if the machine will not go that fast).

It is an error if:

- G53 is used without G0 or G1 being active,
- G53 is used while cutter radius compensation is on.

See relevant chapter for an overview of coordinate systems.

9.7.17 Select Work Offset Coordinate System - G54 to G59 & G59 P~

To select work offset #1, program `G54`, and similarly for the first six offsets. The system-number-G-code pairs are: (1-G54), (2-G55), (3-G56), (4-G57), (5-G58), (6-G59)

To access any of the 254 work offsets (1 - 254) program `G59 P~` where the P word gives the required offset number. Thus G59 P5 is identical in effect to G58.

It is an error if:

- One of these G-codes is used while cutter radius compensation is on.

See relevant chapter for an overview of coordinate systems.

9.7.18 Set Path Control Mode - G61, and G64

Program `G61` to put the machining system into exact stop mode, or `G64` for constant velocity mode. It is OK to program for the mode that is already active. These modes are described in detail above.

9.7.19 Rotate coordinate system – G68 and G69

Program `G68 A~ B~ I~ R~` to rotate the program coordinate system.

`A~` is the X coordinate and `B~` the Y coordinate of the centre of rotation in the current coordinate system (i.e. including all work and tool offsets and G52/G92 offsets.)

`R~` is the rotation angle in degrees (positive is CCW viewed from the positive Z direction).

`I~` is optional and the value is not used. If `I~` is present it causes the given R value to be added to any existing rotation set by G68.

e.g. `G68 A12 B25 R45` causes the coordinate system to be rotated by 45 degrees about the point Z=12, Y=25

Subsequently: `G68 A12 B35 I1 R40` leaves the coordinate system rotated by 85 degrees about X = 12, Y=25

Program `G69` to cancel rotation.

Notes:

- This code only allows rotation when the current plane is X-Y
- The `I` word can be used even if the centre point is different from that used before although in this case, the results need careful planning. It could be useful when simulating engine turning.

9.7.20 Length Units – G70 and G71

Program `G70` to use inches for length units. Program `G71` to use millimetres.

It is usually a good idea to program either `G70` or `G71` near the beginning of a program before any motion occurs, and not to use either one anywhere else in the program. It is the responsibility of the user to be sure all numbers are appropriate for use with the current length units. See also G20/G21 which are synonymous and preferred.

9.7.21 Canned Cycle – High Speed Peck Drill G73

The G73 cycle is intended for deep drilling or milling with chip breaking. See also G83. The retracts in this cycle break the chip but do not totally retract the drill from the hole. It is suitable for tools with long flutes which will clear the broken chips from the hole. This cycle takes a Q number which represents a "delta" increment along the Z-axis. Program

```
G73 X~ Y~ Z~ A~ B~ C~ R~ L~ Q~
```

- Preliminary motion, as described in G81 to 89 canned cycles.
- Move the Z-axis only at the current feed rate downward by delta or to the Z position, whichever is less deep.
- Rapid back out by the distance defined in the *G73 Pullback* DRO on the Settings screen.
- Rapid back down to the current hole bottom, backed off a bit.
- Repeat steps 1, 2, and 3 until the Z position is reached at step 1.
- Retract the Z-axis at traverse rate to clear Z.
- It is an error if:
- the Q number is negative or zero.

9.7.22 Cancel Modal Motion - G80

Program `G80` to ensure no axis motion will occur. It is an error if:

- Axis words are programmed when G80 is active, unless a modal group 0 G code is programmed which uses axis words.

9.7.23 Canned Cycles - G81 to G89

The canned cycles G81 through G89 have been implemented as described in this section. Two examples are given with the description of G81 below.

All canned cycles are performed with respect to the currently selected plane. Any of the three planes (XY, YZ, ZX) may be selected. Throughout this section, most of the descriptions assume the XY-plane has been selected. The behaviour is always analogous if the YZ or XZ-plane is selected.

Rotational axis words are allowed in canned cycles, but it is better to omit them. If rotational axis words are used, the numbers must be the same as the current position numbers so that the rotational axes do not move.

All canned cycles use X, Y, R, and Z numbers in the NC code. These numbers are used to determine X, Y, R, and Z positions. The R (usually meaning retract) position is along the axis perpendicular to the currently selected plane (Z-axis for XY-plane, X-axis for YZ-plane, Y-axis for XZ-plane). Some canned cycles use additional arguments.

For canned cycles, we will call a number "sticky" if, when the same cycle is used on several lines of code in a row, the number must be used the first time, but is optional on the rest of the lines. Sticky numbers keep their value on the rest of the lines if they are not explicitly programmed to be different. The R number is always sticky.

In incremental distance mode: when the XY-plane is selected, X, Y, and R numbers are treated as increments to the current position and Z as an increment from the Z-axis position before the move involving Z takes place; when the YZ or XZ-plane is selected, treatment of the axis words is analogous. In absolute distance mode, the X, Y, R, and Z numbers are absolute positions in the current coordinate system.

The L number is optional and represents the number of repeats. L=0 is not allowed. If the repeat feature is used, it is normally used in incremental distance mode, so that the same sequence of motions is repeated in several equally spaced places along a straight line. In absolute distance mode, L > 1 means "do the same cycle in the same place several times," Omitting the L word is equivalent to specifying L=1. The L number is not sticky.

When L>1 in incremental mode with the XY-plane selected, the X and Y positions are determined by adding the given X and Y numbers either to the current X and Y positions (on the first go-around) or to the X and Y positions at the end of the previous go-around (on the repetitions). The R and Z positions do not change during the repeats.

The height of the retract move at the end of each repeat (called "clear Z" in the descriptions below) is determined by the setting of the retract mode: either to the original Z position (if that is above the R position and the retract mode is G98), or otherwise to the R position.

It is an error if:

- X, Y, and Z words are all missing during a canned cycle,
- A P number is required and a negative P number is used,
- An L number is used that does not evaluate to a positive integer,
- Rotational axis motion is used during a canned cycle,
- Inverse time feed rate is active during a canned cycle,
- Cutter radius compensation is active during a canned cycle.

When the XY plane is active, the Z number is sticky, and it is an error if:

- The Z number is missing and the same canned cycle was not already active,

- The R number is less than the Z number.

When the XZ plane is active, the Y number is sticky, and it is an error if:

- The Y number is missing and the same canned cycle was not already active,
- The R number is less than the Y number.

When the YZ plane is active, the X number is sticky, and it is an error if:

- The X number is missing and the same canned cycle was not already active,
- The R number is less than the X number.

9.7.23.1 Preliminary and In-Between Motion

At the very beginning of the execution of any of the canned cycles, with the XY-plane selected, if the current Z position is below the R position, the Z-axis is traversed to the R position. This happens only once, regardless of the value of L.

In addition, at the beginning of the first cycle and each repeat, the following one or two moves are made:

- A straight traverse parallel to the XY-plane to the given XY-position,
- A straight traverse of the Z-axis only to the R position, if it is not already at the R position.

If the XZ or YZ plane is active, the preliminary and in-between motions are analogous.

9.7.23.2 G81 Cycle

The G81 cycle is intended for drilling. Program `G81 X~ Y~ Z~ A~ B~ C~ R~ L~`

- Preliminary motion, as described above.
- Move the Z-axis only at the current feed rate to the Z position.
- Retract the Z-axis at traverse rate to clear Z.

Example 1. Suppose the current position is (1, 2, 3) and the XY-plane has been selected, and the following line of NC code is interpreted.

```
G90 G81 G98 X4 Y5 Z1.5 R2.8
```

This calls for absolute distance mode (G90), old "Z" retract mode (G98) and calls for the G81 drilling cycle to be performed once. The X number and X position are 4. The Y number and Y position are 5. The Z number and Z position are 1.5. The R number and clear Z are 2.8. The following moves take place.

- A traverse parallel to the XY-plane to (4,5,3)
- A traverse parallel to the Z-axis to (4,5,2.8)
- A feed parallel to the Z-axis to (4,5,1.5)
- A traverse parallel to the Z-axis to (4,5,3)

Example 2. Suppose the current position is (1, 2, 3) and the XY-plane has been selected, and the following line of NC code is interpreted.

```
G91 G81 G98 X4 Y5 Z-0.6 R1.8 L3
```

This calls for incremental distance mode (G91), old "Z" retract mode and calls for the G81 drilling cycle to be repeated three times. The X number is 4, the Y number is 5, the Z number is -0.6 and the R number is 1.8. The initial X position is 5 (=1+4), the initial Y position is 7 (=2+5), the clear Z position is 4.8 (=1.8+3), and the Z position is 4.2 (=4.8-0.6). Old Z is 3.0

The first move is a traverse along the Z-axis to (1,2,4.8), since old Z < clear Z.

The first repeat consists of 3 moves.

- a traverse parallel to the XY-plane to (5,7,4.8)
- a feed parallel to the Z-axis to (5,7, 4.2)
- a traverse parallel to the Z-axis to (5,7,4.8)

The second repeat consists of 3 moves. The X position is reset to 9 (=5+4) and the Y position to 12 (=7+5).

- a traverse parallel to the XY-plane to (9,12,4.8)
- a feed parallel to the Z-axis to (9,12, 4.2)
- a traverse parallel to the Z-axis to (9,12,4.8)

The third repeat consists of 3 moves. The X position is reset to 13 (=9+4) and the Y position to 17 (=12+5).

- a traverse parallel to the XY-plane to (13,17,4.8)
- a feed parallel to the Z-axis to (13,17, 4.2)
- a traverse parallel to the Z-axis to (13,17,4.8)

9.7.23.3 G82 Cycle

The G82 cycle is intended for drilling. Program

```
G82 X~ Y~ Z~ A~ B~ C~ R~ L~ P~
```

- Preliminary motion, as described above.
- Move the Z-axis only at the current feed rate to the Z position.
- Dwell for the P number of seconds.
- Retract the Z-axis at traverse rate to clear Z.

9.7.23.4 G83 Cycle

The G83 cycle (often called peck drilling) is intended for deep drilling or milling with chip breaking. See also G73. The retracts in this cycle clear the hole of chips and cut off any long stringers (which are common when drilling in aluminium). This cycle takes a Q number which represents a "delta" increment along the Z-axis. Program

```
G83 X~ Y~ Z~ A~ B~ C~ R~ L~ Q~
```

- Preliminary motion, as described above.
- Move the Z-axis only at the current feed rate downward by delta or to the Z position, whichever is less deep.
- Rapid back out to the clear Z.
- Rapid back down to the current hole bottom, backed off a bit.
- Repeat steps 1, 2, and 3 until the Z position is reached at step 1.
- Retract the Z-axis at traverse rate to clear Z.

It is an error if:

- The Q number is negative or zero.

9.7.23.5 G85 Cycle

The G85 cycle is intended for boring or reaming, but could be used for drilling or milling. Program `G85 X~ Y~ Z~ A~ B~ C~ R~ L~`

- Preliminary motion, as described above.
- Move the Z-axis only at the current feed rate to the Z position.
- Retract the Z-axis at the current feed rate to clear Z.

9.7.23.6 G86 Cycle

The G86 cycle is intended for boring. This cycle uses a P number for the number of seconds to dwell. Program `G86 X~ Y~ Z~ A~ B~ C~ R~ L~ P~`

- Preliminary motion, as described above.
- Move the Z-axis only at the current feed rate to the Z position.
- Dwell for the P number of seconds.
- Stop the spindle turning.
- Retract the Z-axis at traverse rate to clear Z.
- Restart the spindle in the direction it was going.
- The spindle must be turning before this cycle is used. It is an error if:
- The spindle is not turning before this cycle is executed.

9.7.23.7 G89 Cycle

The G89 cycle is intended for boring. This cycle uses a P number, where P specifies the number of seconds to dwell. Program `G89 X~ Y~ Z~ A~ B~ C~ R~ L~ P~`

- Preliminary motion, as described above.
- Move the Z-axis only at the current feed rate to the Z position.
- Dwell for the P number of seconds.
- Retract the Z-axis at the current feed rate to clear Z.

9.7.24 Set Distance Mode - G90 and G91

Interpretation of Mach3 code can be in one of two distance modes: absolute or incremental.

To go into absolute distance mode, program `G90`. In absolute distance mode, axis numbers (X, Y, Z, A, B, C) usually represent positions in terms of the currently active coordinate system. Any exceptions to that rule are described explicitly in this section describing G-codes.

To go into incremental distance mode, program `G91`. In incremental distance mode, axis numbers (X, Y, Z, A, B, C) usually represent increments from the current values of the numbers.

I, J and K numbers always represent increments, regardless of the distance mode setting.

9.7.25 Set IJ Mode - G90.1 and G91.1

Interpretation of the IJK values in G02 and G03 codes can be in one of two distance modes: absolute or incremental.

To go into absolute IJ mode, program `G90.1`. In absolute distance mode, IJK numbers represent absolute positions in terms of the currently active coordinate system.

To go into incremental IJ mode, program `G91.1`. In incremental distance mode, IJK numbers usually represent increments from the current controlled point.

Incorrect settings of this mode will generally result in large incorrectly oriented arcs in the tool path display.

9.7.26 G92 Offsets - G92, G92.1, G92.2, G92.3

See the chapter on coordinate systems for full details. You are strongly advised not to use this legacy feature on any axis where there is another offset applied.

To make the current point have the coordinates you want (without motion), program `G92 X~ Y~ Z~ A~ B~ C~` , where the axis words contain the axis numbers you want. All axis words are optional, except that at least one must be used. If an axis word is not used for a given axis, the coordinate on that axis of the current point is not changed. It is an error if:

- All axis words are omitted.

G52 and G92 use common internal mechanisms in Mach3 and may not be used together.

When G92 is executed, the origin of the currently active coordinate system moves. To do this, origin offsets are calculated so that the coordinates of the current point with respect to the moved origin are as specified on the line containing the G92. In addition, parameters 5211 to 5216 are set to the X, Y, Z, A, B, and C-axis offsets. The offset for an axis is the amount the origin must be moved so that the coordinate of the controlled point on the axis has the specified value.

Here is an example. Suppose the current point is at X=4 in the currently specified coordinate system and the current X-axis offset is zero, then `G92 X7` sets the X-axis offset to -3, sets parameter 5211 to -3, and causes the X-coordinate of the current point to be 7.

The axis offsets are always used when motion is specified in absolute distance mode using any of the fixture coordinate systems thus all fixture coordinate systems are affected by G92.

Being in incremental distance mode has no effect on the action of G92.

Non-zero offsets may be already be in effect when the G92 is called. They are in effect discarded before the new value is applied. Mathematically, the new value of each offset is A+B, where A is what the offset would be if the old offset were zero, and B is the old offset. For example, after the previous example, the X-value of the current point is 7. If `G92 X9` is then programmed, the new X-axis offset is -5, which is calculated by [[7-9] + -3]. Put another way the `G92 X9` produces the same offset whatever G92 offset was already in place.

To reset axis offsets to zero, program `G92.1` or `G92.2` G92.1 sets parameters 5211 to 5216 to zero, whereas G92.2 leaves their current values alone.

To set the axis offset values to the values given in parameters 5211 to 5216, program `G92.3`

You can set axis offsets in one program and use the same offsets in another program. Program G92 in the first program. This will set parameters 5211 to 5216. Do not use G92.1 in the remainder of the first program. The parameter values will be saved when the first program exits and restored when the second one starts up. Use G92.3 near the beginning of the second program. That will restore the offsets saved in the first program.

9.7.27 Set Feed Rate Mode - G93, G94

Two feed rate modes are recognized: inverse time and units per minute. Program `G93` to start the inverse time mode (this is very infrequently employed). Program `G94` to start the units per minute mode.

In inverse time feed rate mode, an F word means the move should be completed in [one divided by the F number] minutes. For example, if the F number is 2.0, the move should be completed in half a minute.

In units per minute feed rate mode, an F word on the line is interpreted to mean the controlled point should move at a certain number of inches per minute, millimetres per minute, or degrees per minute, depending upon what length units are being used and which axis or axes are moving.

When the inverse time feed rate mode is active, an F word must appear on every line which has a G1, G2, or G3 motion, and an F word on a line that does not have G1, G2, or G3 is ignored. Being in inverse time feed rate mode does not affect G0 (rapid traverse) motions. It is an error if:

- Inverse time feed rate mode is active and a line with G1, G2, or G3 (explicitly or implicitly) does not have an F word.

9.7.28 Set Canned Cycle Return Level - G98 and G99

When the spindle retracts during canned cycles, there is a choice of how far it retracts:

1. retract perpendicular to the selected plane to the position indicated by the R word, or
2. retract perpendicular to the selected plane to the position that the axis was in just before the canned cycle started (unless that position is lower than the position indicated by the R word, in which case use the R word position).

To use option (1), program `G99` To use option (2), program `G98` Remember that the R word has different meanings in absolute distance mode and incremental distance mode.

9.8 Built-in M Codes

M codes interpreted directly by Mach3 are shown in figure 9.5.

M-code	Meaning
M0	Program stop
M1	Optional program stop
M2	Program end
M3	Rotate spindle clockwise
M4	Rotate spindle counter clockwise
M5	Stop spindle rotation
M6	Tool change (by two macros)
M30	Program end and Rewind
M47	Repeat program from first line
M48	Enable speed and feed override
M49	Disable speed and feed override
M98	Call subroutine
M99	Return from subroutine/repeat

Figure 9.5 - Built in M-codes

9.8.1 Program Stopping and Ending - M0, M1, M2, M30

To stop a running program temporarily (regardless of the setting of the optional stop switch), program `M0`.

To stop a running program temporarily (but only if the optional stop switch is on), program `M1`.

It is OK to program M0 and M1 in MDI mode, but the effect will probably not be noticeable, because normal behaviour in MDI mode is to stop after each line of input, anyway.

If a program is stopped by an M0, M1, pressing the cycle start button will restart the program at the following line.

Program `M2` or `M30` to end a program. M2 leaves the next line to be executed as the M2 line. M30 “rewinds” the G-code file. These commands can have the following effects depending on the options chosen on the Configure>Logic dialog:

- Axis offsets are set to zero (like G92.2) and origin offsets are set to the default (like G54).
- Selected plane is set to XY (like G17).
- Distance mode is set to absolute (like G90).

- Feed rate mode is set to Units per minute mode (like G94).
- Feed and speed overrides are set to ON (like M48).
- Cutter compensation is turned off (like G40).
- The spindle is stopped (like M5).
- The current motion mode is set to G1 (like G1).

No more lines of code in the file will be executed after the M2 or M30 command is executed. Pressing cycle start will resume the program (M2) or start the program back at the beginning of the file (M30).

9.8.2 Spindle Control - M3, M4, M5

To start the spindle turning clockwise at the currently programmed speed, program `M3`.

To start the spindle turning counter clockwise at the currently programmed speed, program `M4`.

The spindle speed is programmed by the S word.

To stop the spindle from turning, program `M5`.

It is OK to use M3 or M4 if the spindle speed is set to zero. If this is done (or if the speed override switch is enabled and set to zero), the spindle will not start turning. If, later, the spindle speed is set above zero (or the override switch is turned up), the spindle will start turning. It is permitted to use M3 or M4 when the spindle is already turning or to use M5 when the spindle is already stopped .

9.8.3 Tool change - M6

Provided tool change requests are not to be ignored (as defined in Configure>Logic), Mach3 will call a macro (q.v) M6Start when the command is encountered. It will then wait for Cycle Start to be pressed, execute the macro M6End and continue running the part program. If tool change requests are set to be ignored (in Configure>Logic) then M6 has no effect.

9.8.4 Re-run from first line - M47

On encountering an M47 the part program will continue running from its first line. It is an error if:

- M47 is executed in a subroutine

The run can be stopped by the Pause or Stop buttons

See also the use of M99 outside a subroutine to achieve the same effect.

9.8.5 Override Control - M48 and M49

To enable the speed and feed override, program `M48`. To disable both overrides, program `M49`. It is OK to enable or disable the switches when they are already enabled or disabled.

9.8.6 Call subroutine - M98

This has two formats:

(a) To call a subroutine program within the current part program file code `M98 P~ L~` or `M98 ~P ~Q` The program must contain an O line with the number given by the P word of the Call . This O line is a sort of "label" which indicates the start of the subroutine. The O line may not have a line number (N word) on it. It, and the following code, will normally be written with other subroutines and follow either M2, M30 or M99 so it is not reached directly by the flow of the program.

(b) To call a subroutine which is in a separate file code `M98` (*filename*) `L~`

For example `M98 (test.tap)`

For both formats:

The L word (or optionally the Q word) gives the number of times that the subroutine is to be called before continuing with the line following the M98. If the L (Q) word is omitted then its value defaults to 1.

By using parameters values or incremental moves, a repeated subroutine can make several roughing cuts around a complex path or cut several identical objects from one piece of material.

Subroutine calls may be nested. That is to say, a subroutine may contain a M98 call to another subroutine. As no conditional branching is permitted it is not meaningful for subroutines to call themselves recursively.

9.8.7 Return from subroutine

To return from a subroutine, program `M99.` Execution will continue after the M98 which called the subroutine.

If M99 is written in the main program, i.e. not in a subroutine, then the program will start execution from the first line again. See also M47 to achieve the same effect.

9.9 Other Input Codes

9.9.1 Set Feed Rate - F

To set the feed rate, program `F~`

The units are those defined by the G20/G21 mode.

Depending on the setting in Configure>Logic a revolution of the spindle may be defined as a pulse appearing on the Index input or be derived from the speed requested by the S word or *Set Spindle speed* DRO. For the Sieg KX Series CNC milling machine, the setting should be set to the speed requested by the S word.

The feed rate may sometimes be overridden as described in M48 and M49 above.

9.9.2 Set Spindle Speed - S

To set the speed in revolutions per minute (rpm) of the spindle, program `S~` The spindle will turn at that speed when it has been programmed to start turning. It is OK to program an S word whether the spindle is turning or not. If the speed override switch is enabled and not set at 100%, the speed will be different from what is programmed. It is OK to program S0; the spindle will not turn if that is done. It is an error if:

- The S number is negative.

9.9.3 Select Tool – T

To select a tool, program `T~` where the T number is the slot number in the tool rack for the tool required.

M06 (depending on the settings in Config>Logic) will stop execution of the part-program so you can change the tool by hand. The detailed execution of these changes is set in the *M6Start* and *M6End* macros. If you require anything special you will have to customize these.

The T word, itself, does not actually apply any offsets. Use G43 or G44, q.v., to do this. The H word in G43/G44 specifies which tool table entry to use to get the tool offset. Notice that this is different to the action when you type a tool slot number into the T DRO. In this case an implied G43 is performed so the length offset for the tool will be applied assuming that the slot number and the tool table entry number are the same.

It is OK, but not normally useful, if T words appear on two or more lines with no tool change. It is OK to program T0; no tool will be selected. This is useful if you want the spindle to be empty after a tool change. It is an error if:

- A negative T number is used, or a T number larger than 255 is used.

9.10 Error Handling

This section describes error handling in Mach3.

If a command does not work as expected or does not do anything, check that you have typed it correctly. Common mistakes are GO, instead of G0 i.e. letter O instead of zero) and too many decimal points in numbers. Mach3 does not check for axis over travel (unless software limits are in use) or excessively high feeds or speeds. Nor does it detect situations where a legal command does something unfortunate, such as machining a fixture.

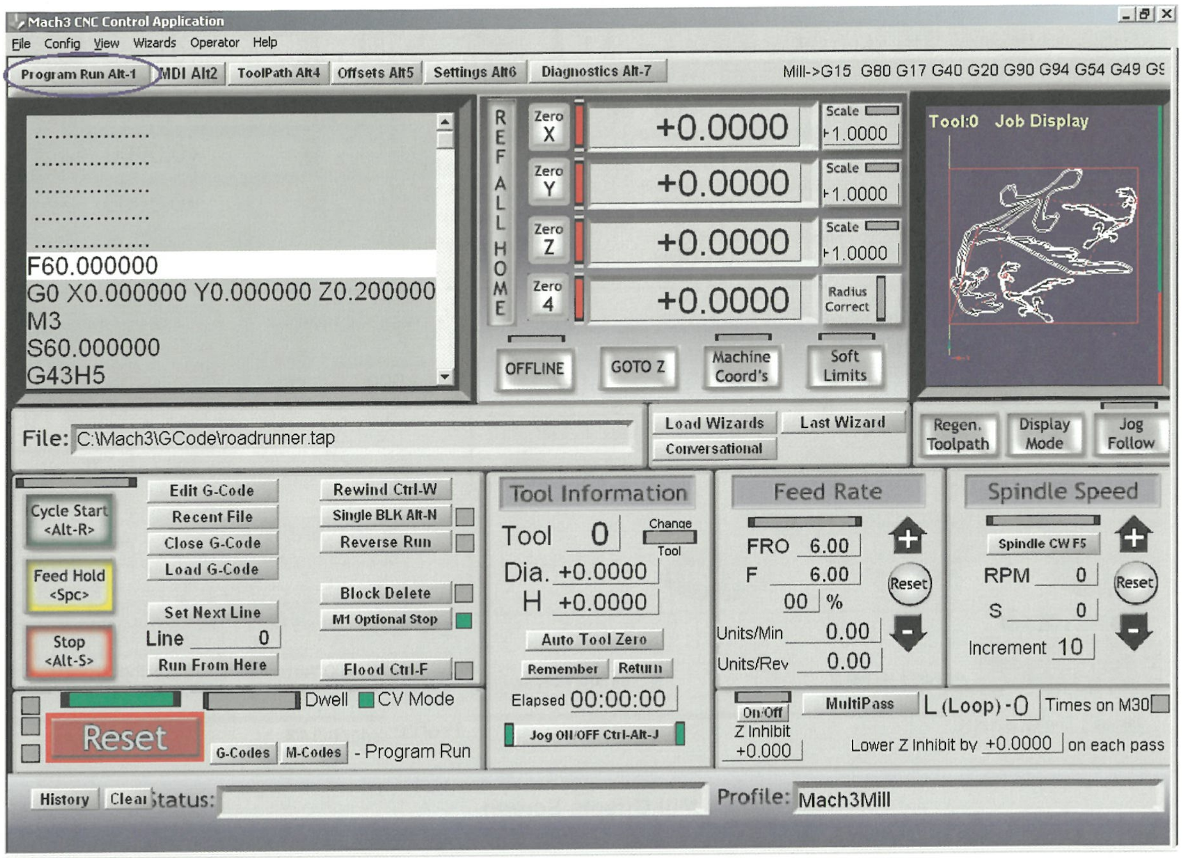
9.11 Order of Execution

The order of execution of items on a line is critical to safe and effective machine operation. Items are executed in the order shown in figure 9.6 if they occur on the same line.

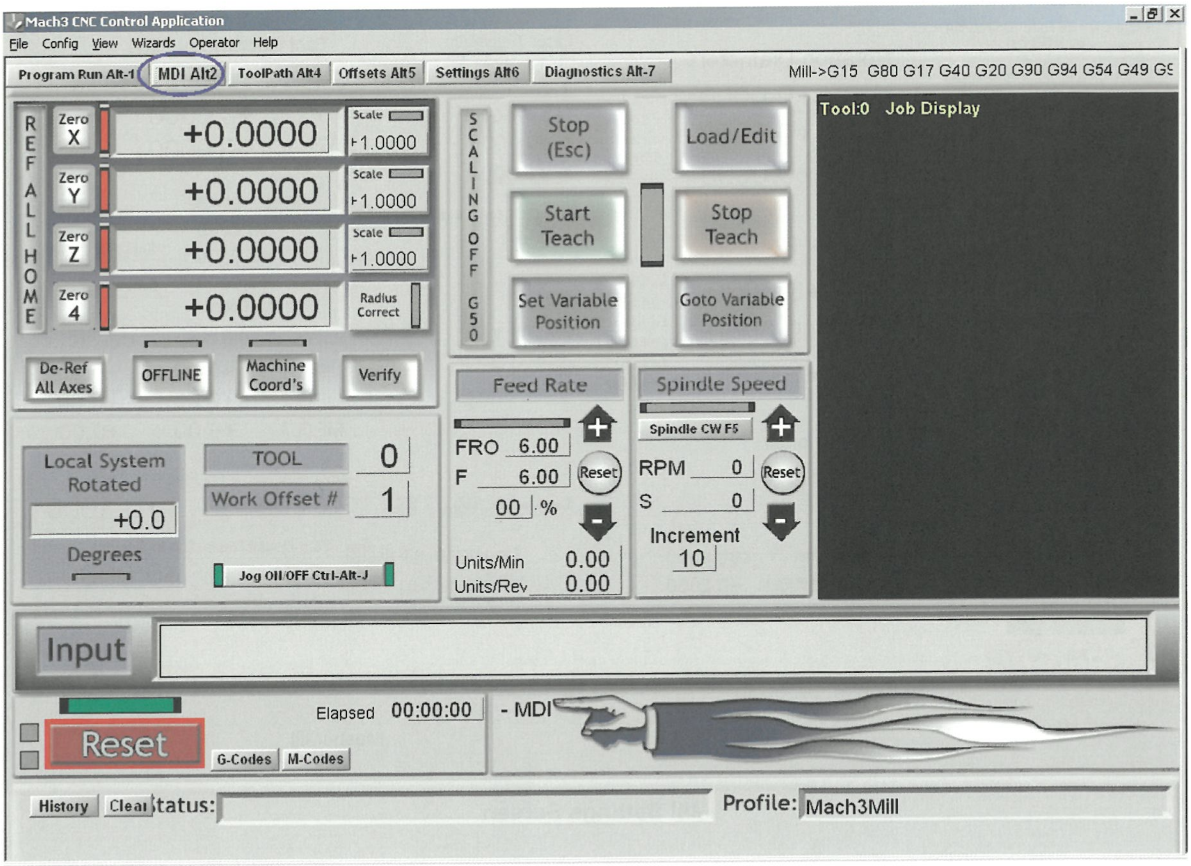
Order	Item
1	Comment
2	Set feed rate mode (G93, G94)
3	Set feed rate (F)
4	Set spindle speed (S)
5	Select tool (T)
6	Tool change (M6) and Execute M-code macros
7	Spindle On/Off (M3, M4, M5)
8	Enable/disable overrides (M48, M49)
9	Dwell (G4)
10	Set active plane (G17, G18, G18)
11	Set length units (G20, G21)
12	Cutter radius compensation On/Off (G40, G41, G42)
13	Tool table offset On/Off (G43, G49)
14	Fixture table select (G54 - G58 & G59 P~)
15	Set path control mode (G61, G61.1, G64)
16	Set distance mode (G90, G91)
17	Set canned cycle return level mode (G98, G99)
18	Home, or change coordinate system data (G10), or set offsets (G92, G94)
19	Perform motion (G0 to G3, G12, G13, G80 to G89 as modified by G53
20	Stop or repeat (M0, M1, M2, M30, M47, M99)

Table 9.6 - Order of execution on a line

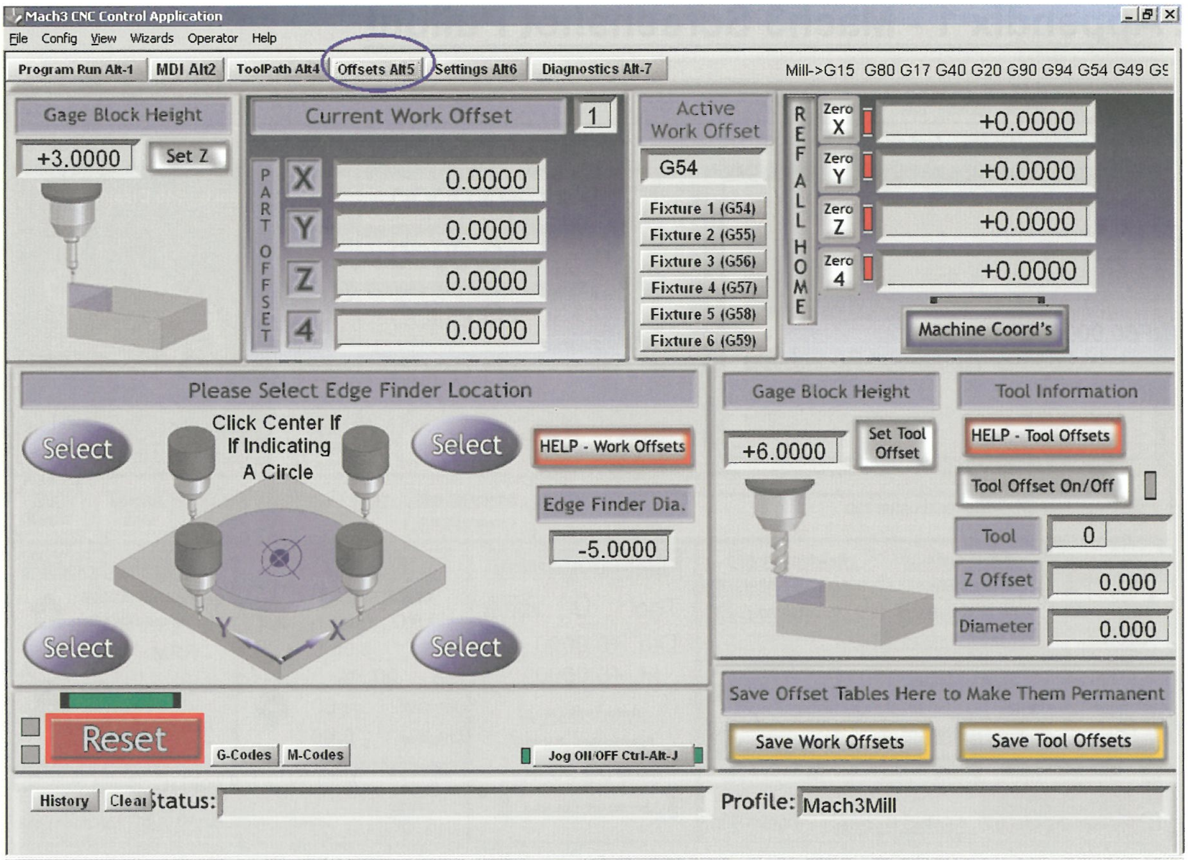
10. Appendix 1 - Mach3 Screenshot Pullout



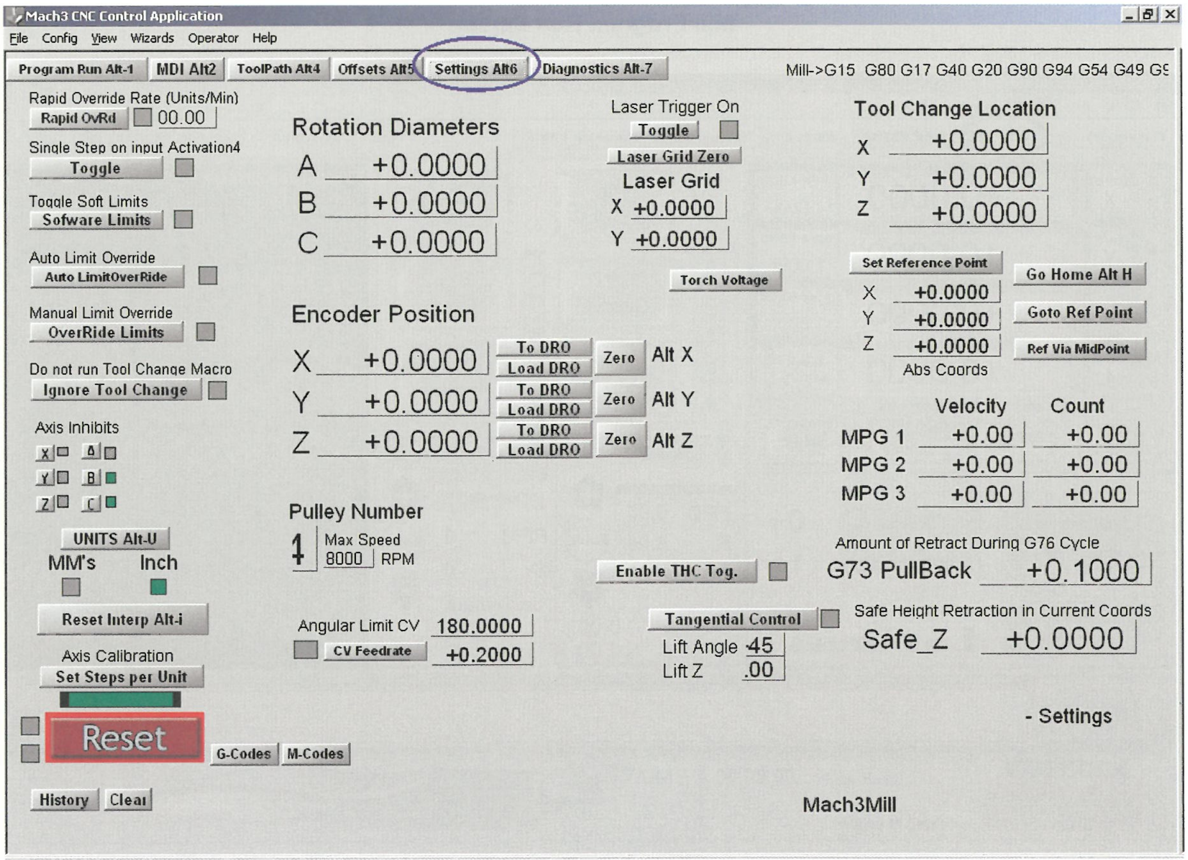
Mill Program Run Screen



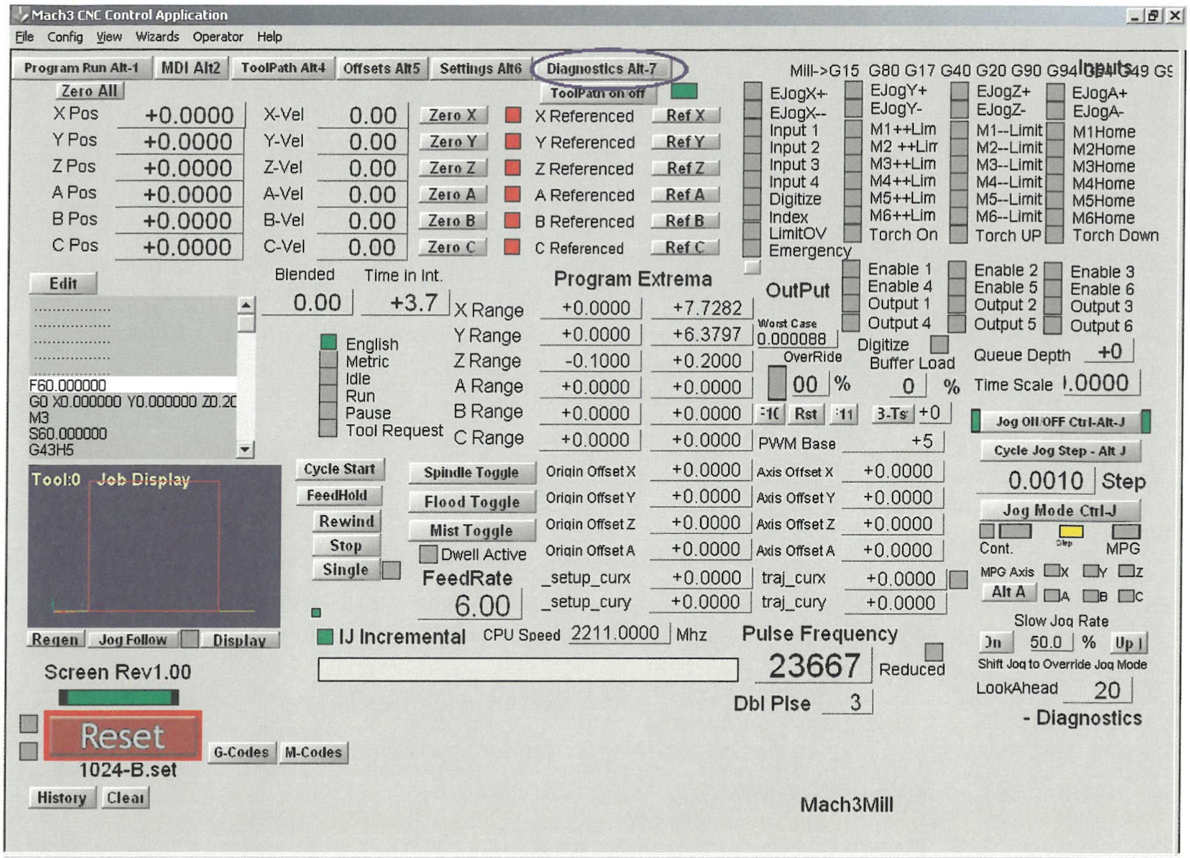
Mill MDI Screen



Mill Offsets Screen



Mill Settings Screen



Mill Diagnostics Screen

Index

Symbols

A

B

C

H

I

U

W

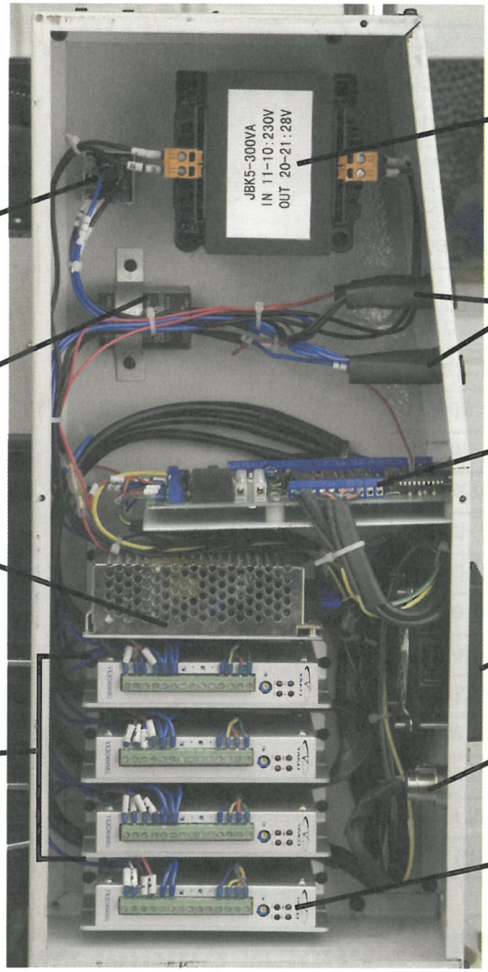
1. KX1 CNC Milling Machine Specifications

End Milling Capacity	10mm
Face Milling Capacity	20mm
Drilling Capacity	10mm
Table Size	400x145mm
Lubrication	Press Button Oilers*
Table travel - X axis	260mm
Table travel - Y axis	110mm
Head travel - Z axis	180mm
Head-Table Distance	70-250mm
Ballscrew Size (Dia./Pitch)	12mm x 4mm
X axis Motor	1.35 Nm
Y axis Motor	1.35 Nm
Z axis Motor	2.2 Nm
Throat	140mm
No. of slots on table	3 (8mm)
Positional Accuracy	0.01mm
Spindle Taper	MT2
Drawbar Thread	M10
Spindle Motor	500w Brushless DC
Spindle Speed	100-7000rpm
Max. Spindle Motor Torque	1.8Nm @ 1500RPM
Power Requirement	230v AC 50Hz or 120v AC 60Hz
Overall Dimensions (w/d/h)	630x630x630mm
Max Space Required (w/d/h)	910x 630x730mm
Shipping Dimensions	76x76x79cm
Weight (Net/Gross)	86/120kg

* Oil Daily with SAE 30 grade hydraulic oil or transmission fluid

2. Electronics Layout

Right Side View



Bridge Rectifier

Smoothing Capacitor

Logic Power Supply 12v DC

X/Y/Z Axes Drivers

Main Power Transformer

Fuse Holders

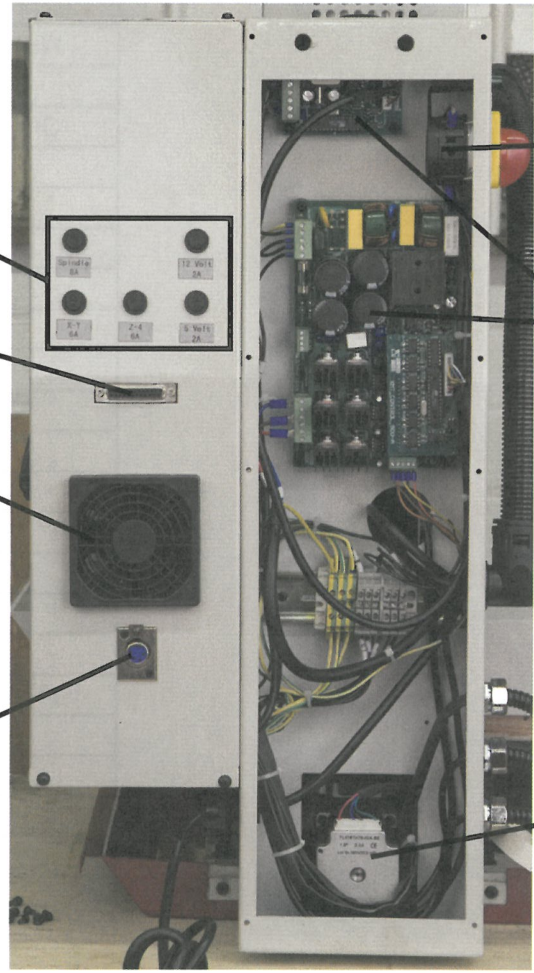
SM11 Parallel Interface Board

Cooling Fan

4th Axis Socket

4th Axis Driver

Rear View



Fuse Holders

Data Port

Cooling Fan

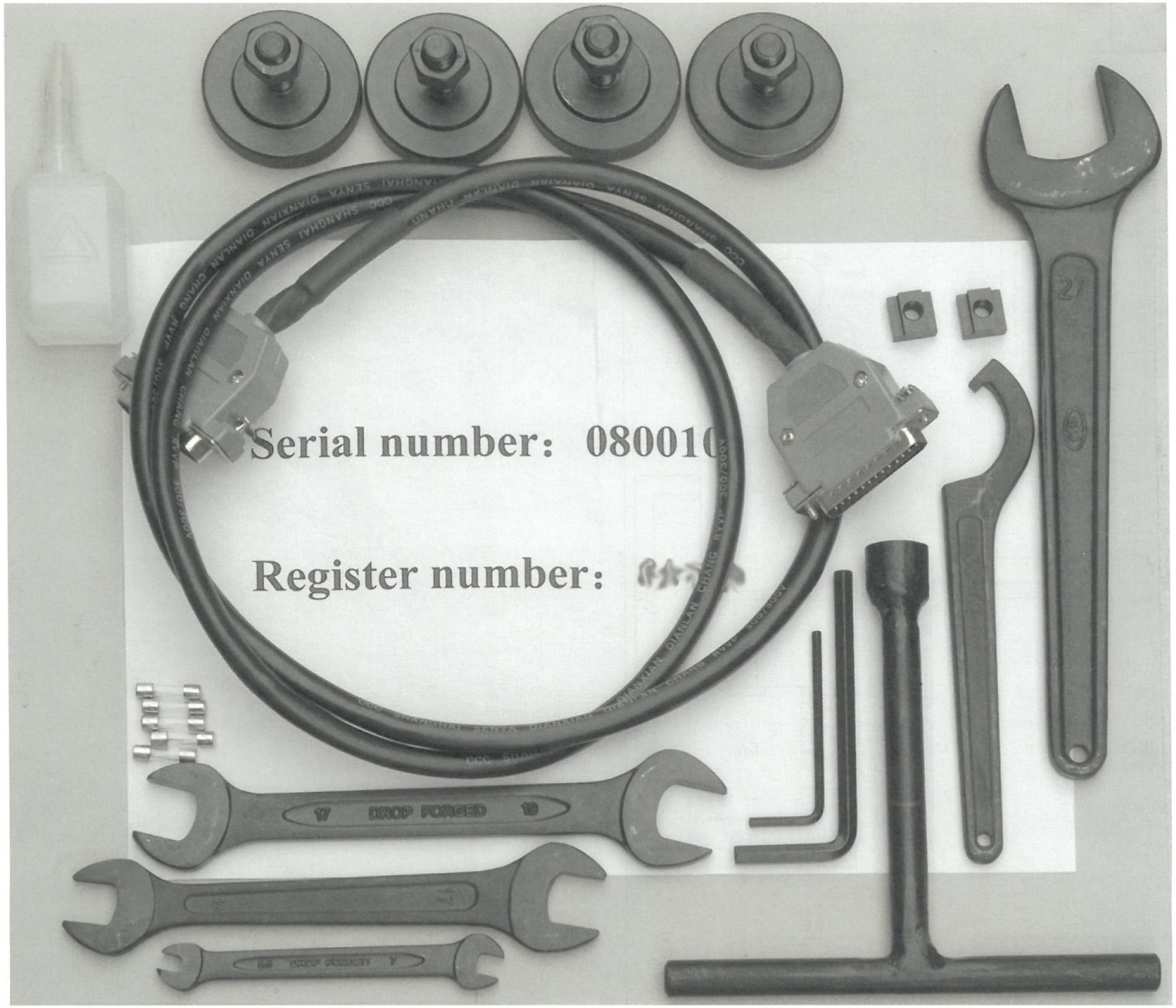
4th Axis Socket (4Way XLR)

Main Power Switch

Speed Controller

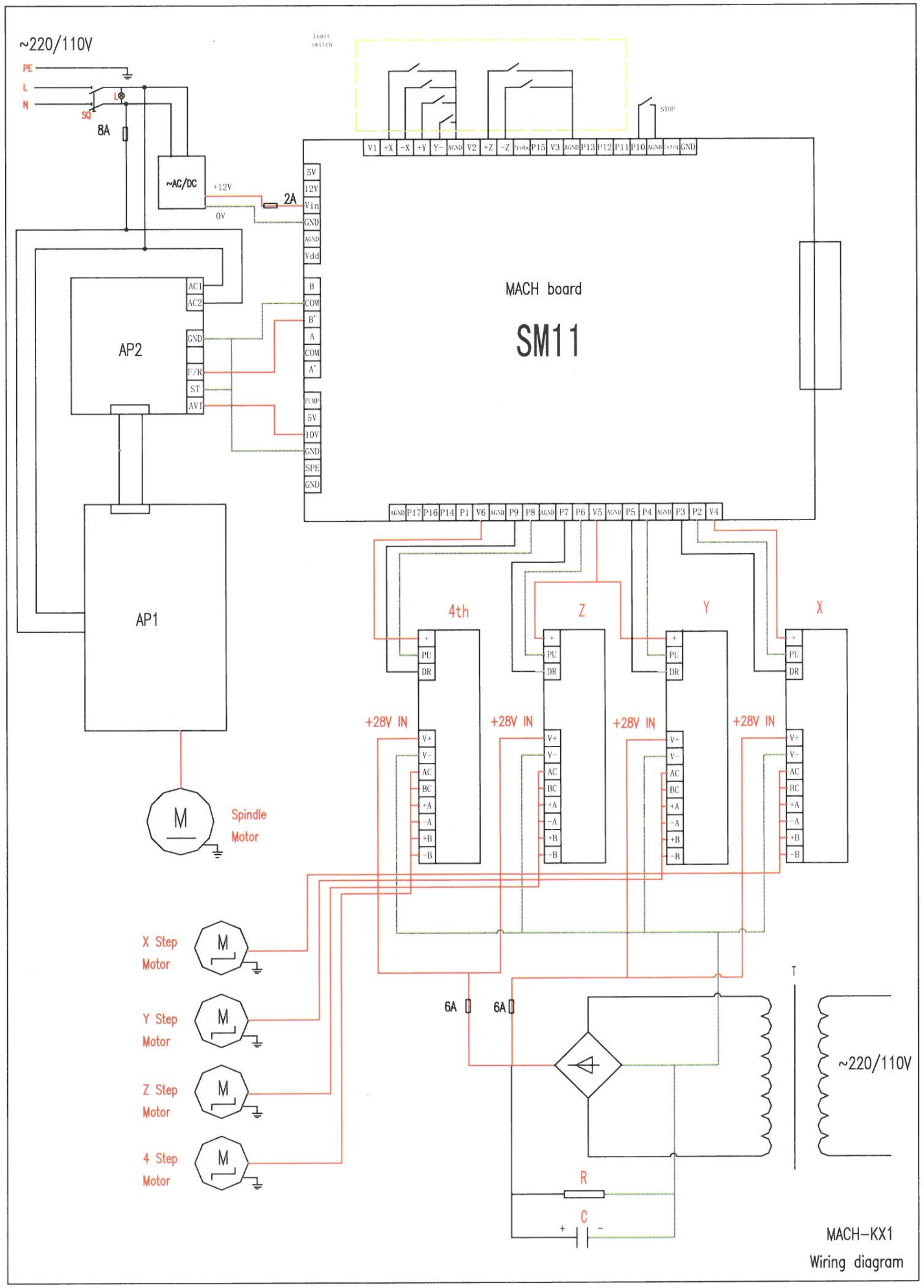
Y Axis Stepper Motor

3. Toolkit and Accessories

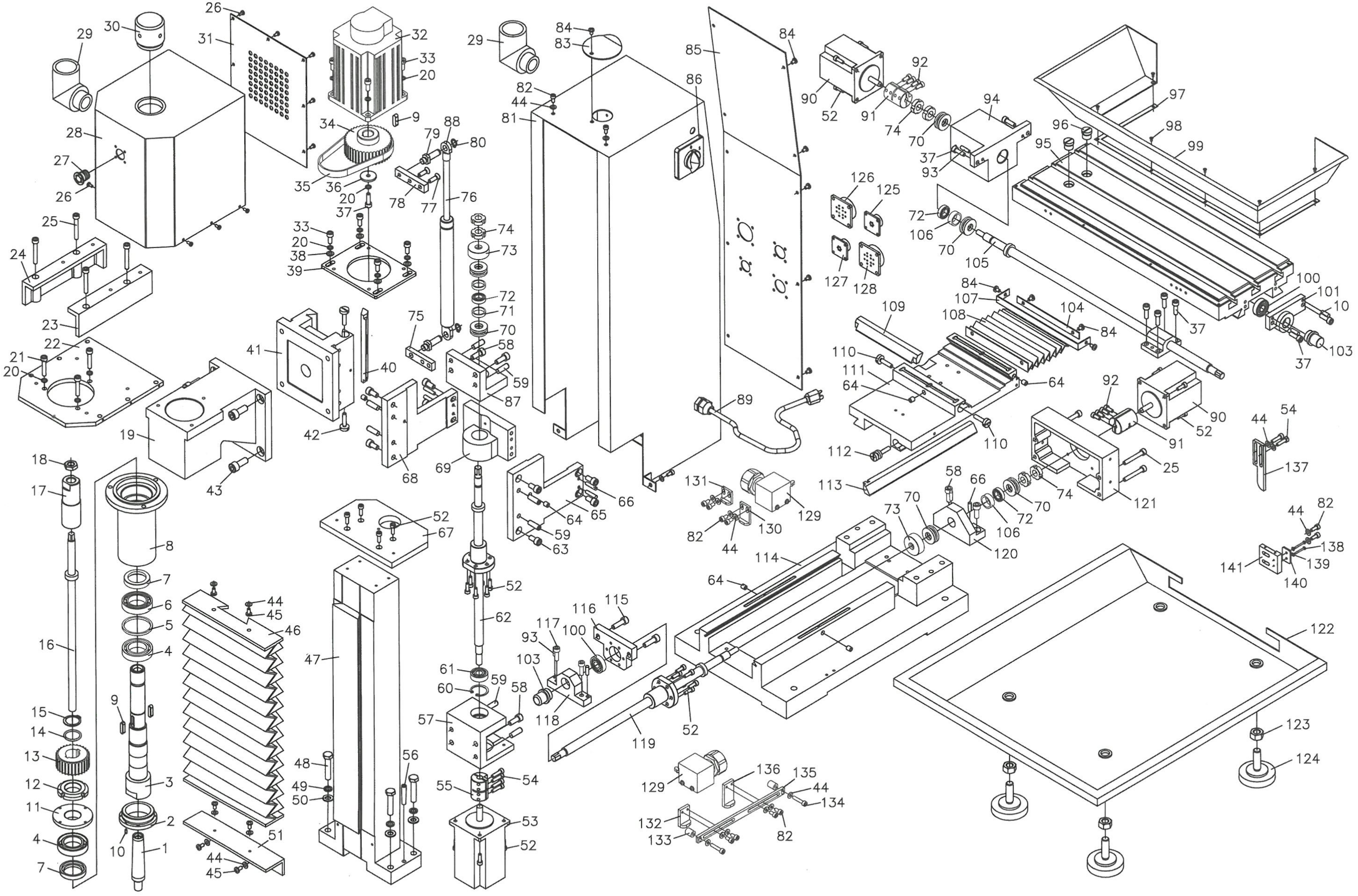


Contents:
1x 27mm AF Open Ended Spanner
1x 17/19mm AF Open Ended Spanner
1x 14/17mm AF Open Ended Spanner
1x 5.5/7mm AF Open Ended Spanner
1x 38-42mm C Spanner
1x 8mm AF Square Tee Wrench
1x 3mm AF Hex Key
1x 6mm AF Hex Key
4x Adjustable Feet
1x Data Cable (D type 25w male - 25w male)
1x Plastic Oil Bottle
2x 8mm Tee Nuts
1x Serial Number / Registration Number sheet (required for on-line support)
Spare Fuses: 2x 2A 250v; 2x 6.3A 250v; 1x 8A 250v (all 20mm quick blow glass fuses)
1x User Manual

4. KX1 circuit diagram



5. KX1 Exploded diagram



6. Spare Parts List

No.	Description	Qty
1	MT2 Drill Chuck Arbor (not supplied)	1
2	Sleeve nut	1
3	Spindle	1
4	Angular Contact Ball Bearing 71905C TA P5 (25x42x9mm)	2
5	Washer	1
6	Ball Bearing 61905-ZZ (=6905-ZZ) (25x42x9mm)	1
7	Oil seal	2
8	Spindle sleeve	1
9	Key 5*18	3
10	Screw M3*6	2
11	Nut for spindle	1
12	Nut M24*1.5	1
13	Spindle belt wheel	1
14	Rubber seal	1
15	Spring check ring	1
16	Draw bar	1
17	Taper extractor sleeve	1
18	Nut M10	1
19	Spindle sleeve bracket	1
20	Light spring washer	12

21	Screw M5*25	3
22	Base board	1
23	Belt wheel box	1
24	Belt wheel box	1
25	Screw M5*35	8
26	Screw M3*6	13
27	Emergency stop switch	1
28	Outer cover assembly	1
29	Right-angle cable tube	2
30	Dust proof cover for spindle	1
31	Back cover	1
32	Brushless motor	1
33	Screw M5*12	8
34	Motor belt wheel	1
35	Timing belt	1
36	Small washer	1
37	Screw M5*16	9
38	Washer 5	4
39	Motor mounting board	1
40	Z axis tapered gib strip	1
41	Spindle mount	1
42	Gib adjusting screw	2
43	Screw M8*25	4

44	Washer 4	24
45	Screw M4*8	6
46	Slideway cover	1
47	Column	1
48	Bolt M8*40	4
49	Spring washer 8	4
50	Washer 8	4
51	Lower cover support	1
52	Screw M4*12	28
53	Stepper motor	1
54	Screw M4*16	6
55	Z-axis coupling	1
56	Taper pin 6*35	2
57	Z-axis ball screw support	1
58	Screw M6*16	6
59	Taper pin 6*20	8
60	Circlip 24	1
61	Ball Bearing 609-ZZ (9x24x7mm)	1
62	Z-axis ball bearing screw assembly	1
63	Screw M6*12	8
64	Oil cup 6	6
65	Side support II	1
66	Taper pin A4*20	6
67	Column upper board	1

68	Side support I	1
69	Ball bearing nut seat	1
70	Thrust Ball Bearing 51100 (10x24x9mm)	6
71	Washer	2
72	Ball Bearing 61800-ZZ (=6800-ZZ) (10x19x5mm)	3
73	Bearing protect sleeve	2
74	Nut M10*1	6
75	Gas strut bottom mounting	1
76	Gas strut	1
77	Screw M5*16	2
78	Gas strut top mounting	1
79	Gas strut pivot	2
80	Circlip	2
81	Column cover	1
82	Screw M4*8	14
83	Ball screw access cover	1
84	Screw M4*6	15
85	Back cover board	1
86	Switch	1
87	Bearing seat	1
88	Gas strut end	1
89	Power line	1
90	Stepper motor	2
91	Coupling sleeve	2

92	Screw M4*10	8
93	Taper pin A4*16	4
94	X axis motor mount	1
95	Work table	1
96	Screw	2
97	Rubber strip	1
98	Screw M3*6	6
99	Guard for work table	1
100	Ball Bearing 609-ZZ (9x24x7mm)	2
101	Bearing seat	1
102	Taper pin A4*14	2
103	Small sleeve	2
104	Cover support II	1
105	X axis ball screw assembly	1
106	Washer	2
107	Cover support I	1
108	Slideway cover	1
109	X axis tapered gib strip	1
110	Screw	2
111	Saddle	1
112	Screw	1
113	Y axis tapered gib strip	1
114	Base	1
115	Screw M6*20	2
116	Y axis ball nut mount	1

117	Screw M5*20	2
118	Y axis bearing seat	1
119	Y axis ball screw	1
120	Y axis bearing mount	1
121	Y axis motor mount	1
122	Swarf tray	1
123	Nut M10	4
124	Adjustable foot	4
125	Five-core plug	1
126	Nineteen-core plug	1
127	Five-core plug	1
128	Sixteen-core plug	1
129	Travel switch	2
130	X-axis left bump block	1
131	X-axis right bump block	1
132	Y-axis right bump block	1
133	Plank mat block	2
134	Screw M4*20	2
135	Y-axis stop mount	1
136	Y-axis left bump block	1
137	Z-axis upper bump block	1
138	Screw M2*18	6
139	Washer 2	6
140	Mounting block	3
141	Travel switch seat	3