Mach1 CNC Control

The care and feeding of the Mach1 CNC system

What is Mach1?

Mach1 is designed to control up to 6 axis of stepper or step/dir servos from the parallel port of a PC running Windows XP or Windows 2000 with future plans to migrate it in smaller form to the Win95/98 platform.

CNC under Windows platforms has long been thought to be difficult if not impossible due to unpredictable interrupt and timing systems. Mach1 is designed to avail itself of interrupt zero processing which allows it to bypass Microsoft's limitations in this respect. When interrupt zero takeover of the 8259 PIC is deemed impossible due to different motherboard hardware the program automatically attempts to take over interrupt 255 which is the highest priority interrupt on local PIC interrupting systems and use the Local APIC timer to instantiate a real-time subsystem under Windows.

Master5, the predecessor of Mach1 used a 8254 timer interrupt at a priority of 8 which was fairly low, but did allow Windows 98 to perform rudimentary real-time functions at 8192Hz while Mach1 running a much higher priority performs well at 25,000hz in the current version using 2 printer ports and has been tested at a top speed of 100,000Hz with one port. To maintain reliability amongst the highest number of motherboard systems the software has been locked at 25Khz but may, some day, expand beyond that speed.

Mach1 has been designed with Micro-Step controllers in mind. There are many ways to create pulse streams, each with their own advantages and disadvantages from the perspective of programmer and controller alike and a decision was made to take advantage of methods which couple themselves very well to micro-step systems. While half and full step controller do perform well, due to the major contributions to those systems of motor current and resonant frequency creation, some may be more sensitive and respond more poorly than others. The vast majority of micro-step systems will find Mach1 very responsive with no resonance or detent torque issues, allowing a much higher movement speed than previously found on a Windows platform.

How do I install it?

At the moment, in its Alpha stage, the install program is Mach1Alphaxx.exe , the xx is the current major and minor version. As of the last modification of this document the version is 7.2.

Run the program Mach1Alphaxx.exe and the installation will proceed automatically. At the end of the installation you will be asked if you wish to initialize the system, unless special circumstances are present you should always let the system initialize, this loads the OCX and driver and will remove old ones. This is important. If asked to reboot and you have never run Mach1 before you MUST do so. Failure to reboot will not only invalidate the installation when the computer crashes when you run, but may create a corrupted driver situation which can be tough to resolve without some knowledge of the control panel and driver system on your computer. If you HAVE run Mach1 before, you do not have to reboot ever again when installing a Mach1 upgrade of re-installing on the same system unless you have re-installed windows.

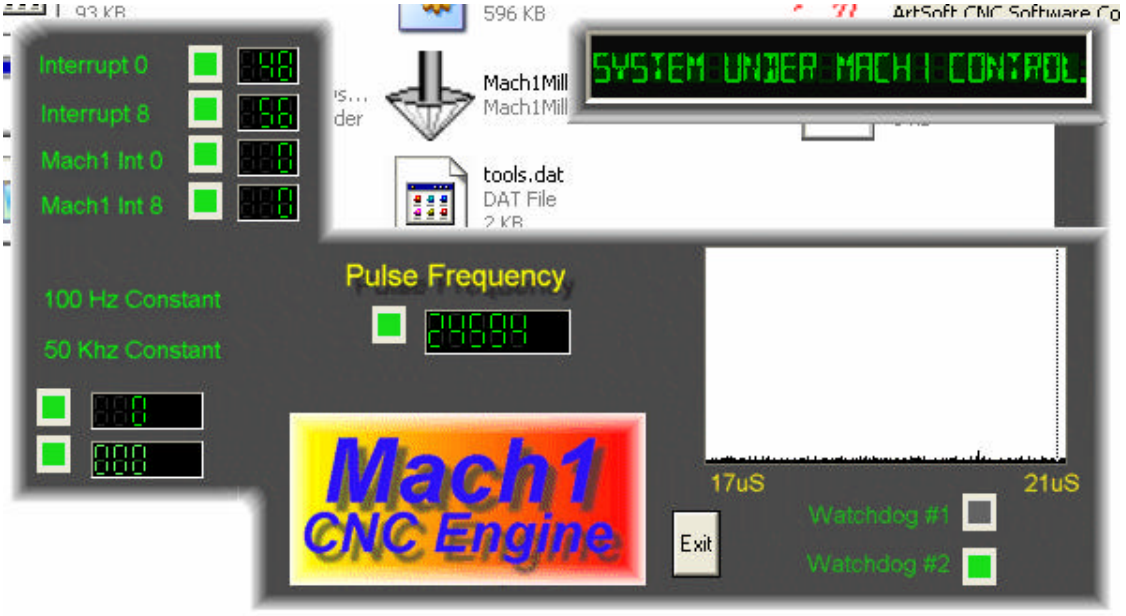
TECH TIP:

You need not read this unless you wish to understand why the reboot and why it WILL create a problem if you don't.

The Mach1 driver is written as a form of virus. It is not a destructive virus, but in order to create a real-time subsystem, it is necessary for Mach1 to create code for timing in memory and then modify that code as it checks your computer to re-write the code to match your system. Windows does not allow this to occur unless a special security key is generated to inform Windows that this type of self-modifying code is allowed in your system. A reboot is necessary to instantiate this change into your system. Without the reboot, when the code tries to self modify, you will instantly get a Blue Screen of death and the security key will be lost and the driver corrupted. Since the driver was loaded, it will reload on the next boot which will signal a corrupt file to the disk checking system and the process must be started again.

Can I test to see if the install went OK?

Glad you asked. Yes, you can and I recommend it highly. Mach1 is not a simple program. It takes great liberties with the OS in order to perform its job, this means it will not work on all systems due to many factors. QuickTime's system monitor running in the background can kill it for example and I'm sure there are MANY 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 at startup in the background, some appear as systray icons and others do not show themselves in any way. Because of this it is important, though not mandatory that you check your system when you suspect something is wrong or you just want to check that an install went well. In the installation directory (Normally C:\Mach1) there is a program called OCXDriverTest.exe. Its screen shot is below:



In the photo you can see a few interesting things. The square window on your right is a type of timing analyzer. When the timer is not running it is black. When it is running it displays a line along the bottom with small variations pushing upwards. These small variations are the changes in timing from one interrupt cycle to another. There should be no lines longer than ¼ inch or so on most systems. Even if there are variations its possible they are below the threshold necessary to create timing jitters so a movement test would be performed in that case to see if jogging is smooth.

The left side, shows Int 0 as vector 48 and Int 8 as vector 56. These are the old IDT vectors used by windows prior to Mach1 taking them over. If these boxes read zero, then you are either using a newer, more powerful motherboard or a dual Pentium system which means the driver has switched to APIC timing as an alternative method and has not redirected the old interrupts as they are lower priority then the vector Mach1 took. You can ignore all other boxes with the exception of the pulse timer. It should be fairly steady at 24,600Hz, but may vary around, even wildly, on some systems. This does not mean the pulse timer is necessarily unsteady, it may mean that the computer is fairly heavily loaded or slow to begin with, since Mach1 takes the highest priority in the system to do its job, the clock may be shunted down to a priority slow enough that 1 second is a variable length of time. Since the pulse count is based on one second of Windows time, variations in Windows time will make the pulse count look like

its swinging around a lot even when it is rock solid. Basically, if you see a similar screen to the one above, everything is working well.

You may have one of 2 or 3 things happen to you when running the test.

1) "Driver not found or installed, contact Art.", this means that the driver is not loaded for whatever reason. I have seen this on some 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 an issue loading the driver. It may need to be loaded manually. See the next section "How do I load a driver manually".
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, please notify me and We'll see if something can be done. I have had instances of people crashing due to the software incorrectly determining that the APIC mnode should be active, when it should use the original 8259 mode. I can send you a "forced 8259" mode driver to see if this cures your system. If it doesn't, theres not much can be done at this point. I try very hard to solve the problems of crashing when I can get a computer that crashes, but I have great difficulty finding them these days. Mach1 now runs even on my dual Pentium XEON machine from Dell so as each computer I find that does crash is fixed, the program gets more robust. Hopefully, most systems CAN run Mach1.

How do I load a Driver Manually?

If you are running Win2000 , or even on a very few XP systems, you may want to remove or install the driver manually. The following will show you how.

1) Open the control panel on your computer from the start button. Run the icon labeled SYSTEM.

Removing a Driver….

Select the hardware tab as shown and then hit the device manager button. A screen will appear with the following type of info. If you see a Master5 CNC Engines line, then a driver is loaded for Master5 or Mach1.

In this case the Mach1 driver is obvious. The Pulser Driver is for Master5 which can co-exist on the same machine. To remove the driver, just right click and select uninstall. It should go away immediately, if the system asks to reboot, then the driver locked up when running so you must be removing it because something went wrong. It may be corrupted so a re-install might help.

Installing a Driver:

Use the control panel as above, but select “Add Hardware” Icon. The procedure varies a bit from here depending on your system Os and what updates are done. Detailed is the Win XP proceedure.

When the window opens select Next and the system will search for hardware. Normally, it will find none and ask if you have already connected the hardware. Answer yes and hit the “next” button. You will be presented with a list of hardware, scroll to the bottom and you will see “Add a new hardware device” , select this option and hit next. You will be asked if you wish Windows to search for hardware or select from a list. Choose select from list. You will then be shown a hardware category list. If you have used Mach1 or Master5 before , you will see Master5 CNC engine, you may select that, if you do not have that selection, then select “show all devices” and hit next.

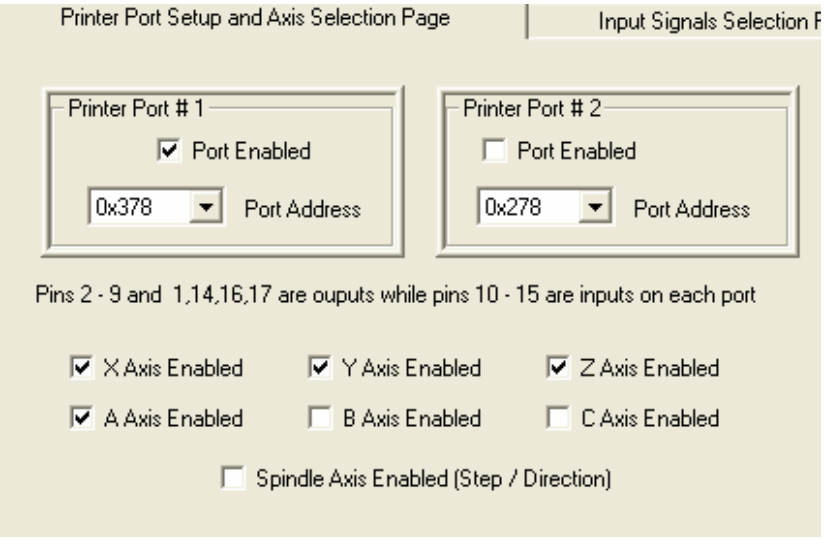
If you get a not-responding message from windows at this point, your windows installation is corrupted. Re-install Windows and then try again. Select “have disk” at the next window and point the file selector to the Mach1 directory (Normally C:\Mach1) and you will see windows finds a file called Mach1.inf. Select this file and hit select, then finish and windows will install the driver.

The Win2K procedure is a little different but close enough I think you can figure it out. (After all, you’re a mechanical kind of person right?)

OK, Now its installed, how do I run this thing?

Congratulations, if you made it here, your ready to run. Lets do this one step at a time. The only thing we'll deal with here is configuration which is performed from the menu item Config. We'll go though the steps one at a time and you should be moving in a few minutes.

First, lets get the ports configured. Select Config/Configure Ports

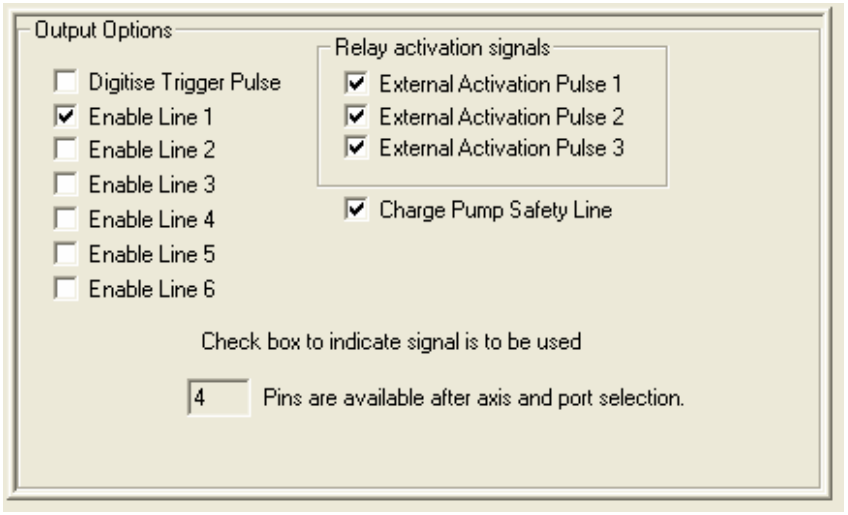


If you intend to use port #1, check its box as enabled. This simply turns on that port. It address must be correct, but the default fits 99% of systems. Other selections are available. If you NEED to know the port is correct , use the start/All Program/Accessories/System Tools/System information selection in windows to pull up system information. You printer port address is Hardware resources/IO and is shown in the dialog as printer port 0xnnn , this is the number that should already be in the Port address window above.

Now select the axis you intend to use. This is important as the system will ignore moves for unselected axis.

If you wish to use the second port as well, the same rules apply, the address can be found at the same place.

Now lets do the output pin selection. This doesn't select the pin numbers or polarity, just whether or not you intend to use a particular signal. This lets the system know how many pins are used up. By selecting the axis in the previous selection, you already told the system how many step and direction pins you will be using, so let's go see. Select "Output signals selection page" from the tabs.



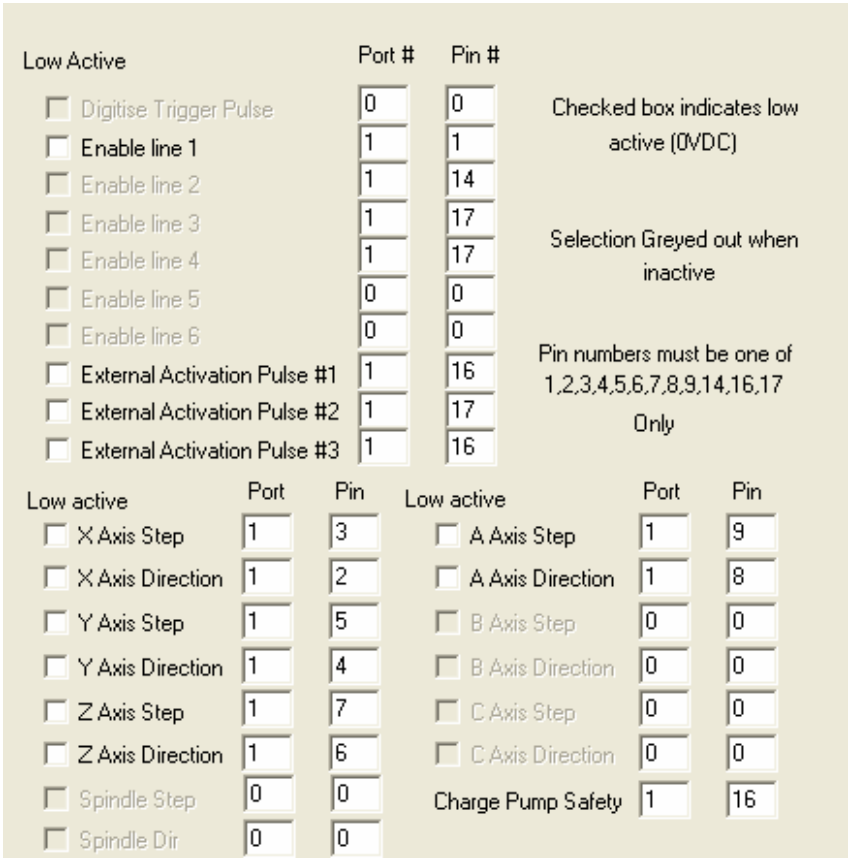
As you can see, there are not many output options. You've already indicated you need step and dir times the number of axis, here you select what additional signals you intend to use. They are as follows..

1) Digitize Trigger Pulse

You won't need this yet. It will be a trigger that can tell a laser scanner or other device to take a measurement at specific times while moving.

2) Enable line 1 – 6 . Most systems use enable lines. Some use 1 line for all controllers, some like 1 line for each controller. All lines selected as enabled by checking them will respond to an EStop by deactivating and will activate upon a Reset to turn on your controllers. Normally, only one enable is necessary.
3) Relay Activation Signals: These are for spindle relays , or mist, Flood or nay other purpose you may have. When complete, you will be able to activate these signals from the program when certain events occur.
4) Charge pump Safety line. This signal is for people who want to be safe from inadvertent movement while booting or shutting down their computer. If you build a small circuit to monitor this line, it can turn on the power to your machine only when observing a 12.5Khz signal from this line. This 12.5Khz signal will only be available when in a reset and safe condition. Rebooting or shutting down your machine will NOT create a fake 12.5Khz signal.

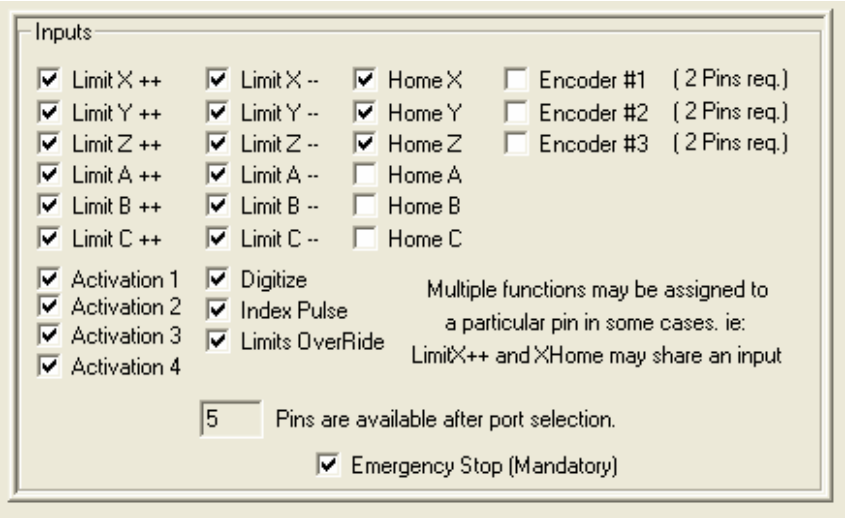
Now we need to know what these signals look like. What their parameters are. That is done on the Printer port Ouput pins tab. Select it now.



You'll notice that some pins are greyed out. That's because you didn't select them to be used. As a result, the system has turned them off. By the way, any changes MUST be saved by hitting "apply" after changing them. Changing tabs will throw away any changes to be safe unless the APPLY button is pressed.

Beside each signal is a checkbox. This is to indicate if the signal is low active, or high active. Most signals are high-active, so if your unsure, leave it unchecked. Putting Zero in a port number will deactivate that signal. The port numbers should be port 1 or 2 only. The possible pin numbers are listed on the screen, the defaults are setup as Master5 is by default. Again, hit apply if you change anything.

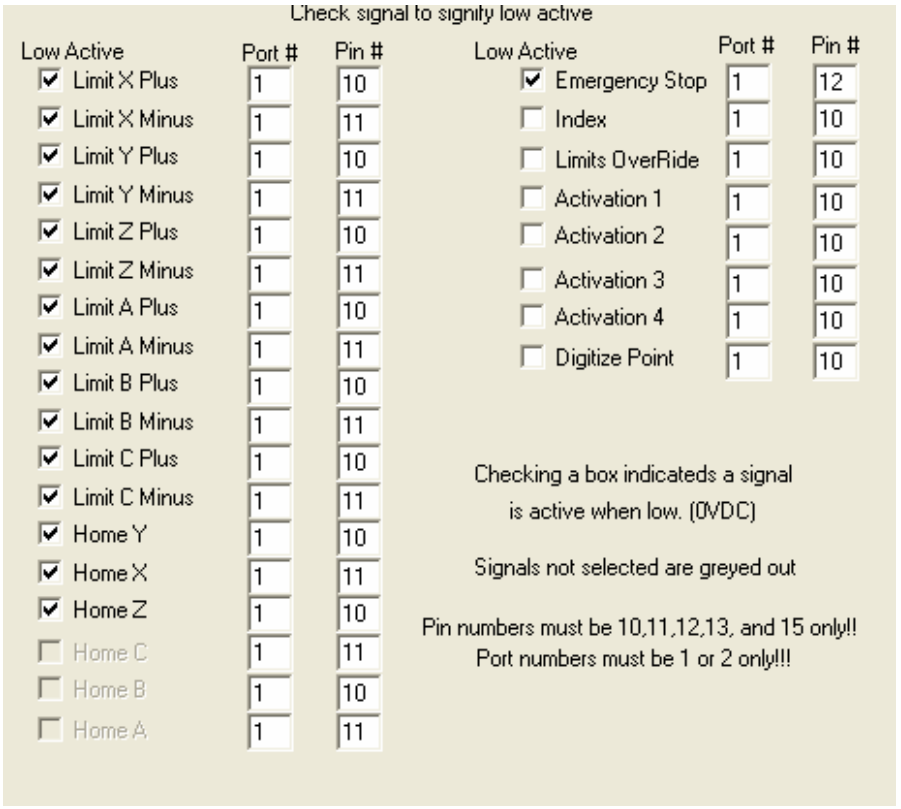
Let's look at input pins now with Input Signals Selection Page.



This is just to select the signals you will use. There are many. If you use one switch on your system in an axis, but you wish to have home, and limit functionality you can share pins, so you would select LimitX—and Home X even though you intend to use one pin for both. As you can see, I am sharing many of them. Select the ones you intend to use, then hit apply. Note: If you only have one switch on an axis, it would not make sense to select both limit++ and limit—, if you have two switches connected together, you may select both though as they can share a pin.

You'll notice that it tells you how many pins you have left after selecting the ports. Since signals can share, you may have more selections than pins.

Now , let's set the pins for these signals with the "Printer port input pins selector.



Wow , that looks complex…(Not really), the boxes indicate if a pin is considered active if you ground it or let it float high by not connecting it. You'll notice the default is to connect a pin to ground to activate it. E-Stop is always defaulted this way so that an empty port will not trigger an estop so you can play with the software without tripping off the air.

Pick the pins you'll use. The examples above are used in debugging, I can use 3 buttons to activate all the signals. I change this screen frequently, but the install defaults to Master5's setup. Obviously, it will vary widely with different people.

Many of these pins are not yet active in the software. Index is for threading on a lathe, Digitize will be used in the digitizing module, Activate 1,2,3 will be used for pallet shuttles , tool changers, …etc.. Select the ones you need. The most important one is E-Stop, if you find your tripping out and cannot reset the software into a safe condition, check the neg box or uncheck it and see if that fixes the problem, this is the most common reason a person cannot reset the main software.

You may share pins here. If you have one switch on your Z axis, and its at the top, for example, you can put Port1 Pin10 for it on LimitZ++ and set the same fro Z-Home. Both signals will simply share the pin. The software will then use the same switch for home as for limit. (Limits are not yet active, but this is the way it will work).

Well, that's it for the ports and pins. Let's see how the motors turn before we go any further.

Exit the dialog after applying any changes you've done. Press reset on the screen. You know, the flashing button.

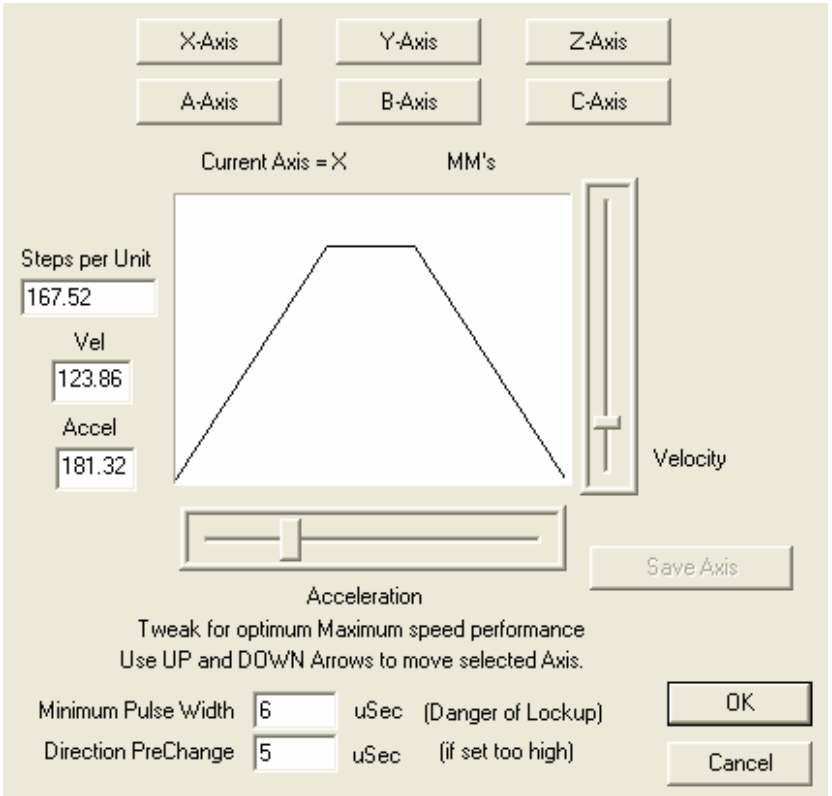
This one:

It should change from a flashing button to this button:

This means if you should press this button, you will now trigger an Emergency stop (Estop) and the button will then flash. While the button is flashing, no movement is possible and all outputs go to inactive condition (Hi or Lo) depending on your previous selections.

We need to select the setup units of the control now. The control can use either inches or mm's as its main setup unit. This is not the cutting units or the system you will necessarily use in your work, just the units that all the setups will be done in. Europeans will tend to use metric, while North Americans will tend to Metric. It doesn't matter which you choose as long as your comfortable with that system. Just select the menu item Configure/Setup Units to choose your preference.

OK, assuming you have an E-STOP green condition as above, you can now tune your motors. Bring up the Configure/Motor tuning command from the main menu. It should look like this:



Tuning in Mach1 is pretty easy. Let's go over each item.

Axis selections. Select the X Axis. All other axis are done in exactly the same way. Just press the “X” button. You can now see a visual representation of the acceleration and velocity maximum of the axis. For it too mean anything we need to set the steps per unit. If you know your steps per unit, enter it in the Box labeled (of course) Steps per unit. Now hit save axis to make sure this is stored, the velocity slider and the acceleration may change on their own as their ranges have now changed to reflect a maximum pulse rate of 25000 pulses per second divided by the number of pulses per unit. In the example above the unit is set to 167.52 steps per mm. Now select a low acceleration and a mid level velocity with the sliders labeled appropriately enough Velocity and Acceleration. Lets try moving the X motor by pressing the up and down arrow keys. You X motor should move. Be careful, limit switches are not active.

If the motor does not move or make a sound, check the following.

1) Enable lines are set properly in the ports config. Is the motor able to be easily turned by hand, if so, the enable is not activated. Check the polarity of the signal as well, if all else seems correct, then check the “neg” box on the enable line in the Ports config dialog under Printer Port pin-outs.
2) Is the Estop flashing Reset? If so find the problem from the previous step. You should not have a flashing reset at this point.

If the motor moved adjust it by judging if it went too fast or too slow and adjust the velocity accordingly. If it’s too quickly “jerking” away from stop lower the acceleration slider and try again.

If the motor makes a sound, but doesn’t move, try lowering the velocity until it oes move and adjust slowly up from there. You can either type in new values in the boxes or move the slider. If you have a wheel on your mouse, you may roll the wheel after selecting a slider for very fine adjustment.

If nothing seems to stop the motors from growling, try raising the pulse width. This value cannot go over 15 and the lower the better. Most controllers of quality will respond with the default 1uS pulse width. If you find it goes well in one direction, but not the other, try making the Direction pre-change longer. This is the amount of time the direction bit toggles prior to pulsing the step line. The motor should move very smoothly and with a small noticeable ramping up to speed. You can make this “small” ramp as small as you wish but a noticeable ramp is recommended.

When the motor moves smoothly back and forth with the up and down arrows, press the save axis button. This will store the speed limits and acceleration profiles for future use. Do this for all axis that you intend to use and exit the dialog.

Now for Hot keys.

Use the menu item for configure/Hot Keys.

One by one, press a button for an axis of movement, you will be asked to press a key. Most users use the arrow keys and numeric keys for the various axis, they are up to you. In future versions more commands will be available to be “Hot-Keyed”.

Now you are ready to use the control. There are a few more things you can configure, like backlash, encoders, output devices (like spindle control) and most of these are self-explanatory as you select their menus. More explanation on these items can be found in the following pages on Control Usage.

Program Control:

DRO's:

The Digital readouts are controllable. By left-clicking the colored portion at the top of the DRO a menu will open asking you to either zero that axis or set it to an arbitrary value. This allows for single axis zeroing or presetting a position reference.

Velocity Control :

This window show you in real-time the speed of your toolpath. It is a blended speed of all axis to show the speed of the tool through the material. It is in units/second so may be inches per second or mm's per second depending on the units you have selected. (see next item)

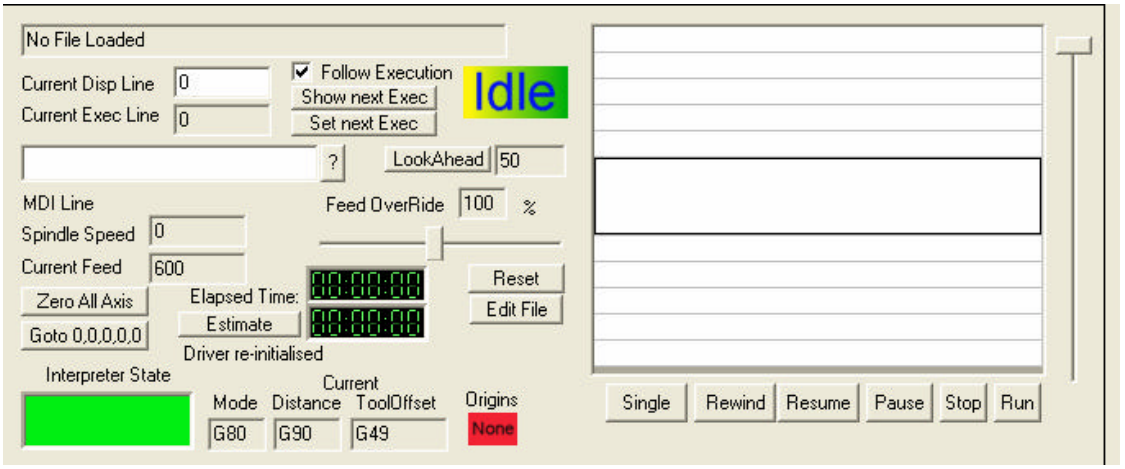
Units :

This button (display) will change the current job units from English to metric and back again. All positions are converted when switching as well as feed speeds and steps per unit. The units may also be switched with the G20 (English) or G21 (Metric) G-Code command words.

Tab Views:

Depending on your resolution, the system will either put two redundant tab views on the screen or one. The system will choose that option depending on if you use an 800x600 or a 1024x768 screen resolution. The two-tab views are used to simultaneously view two different control modules. Lets look at them one at a time.

G-Code



The top left box will display the name of any currently loaded G-Code file. Mach1 is a G-Code interpreter and all input filters, JPG, DXF, BMP, HPGL and others will create a G-Code file to be run. Input filters, when implemented will auto-load their G-Code output when creating a new file. The name of that file will be displayed here.

Current display line is the currently highlighted and enlarged line in the gcode display.

Current Exec line is the currently executing line. If you have stopped or paused a job, this is an indicator as to what line you were on. If you wish to continue a job later, this is the line number you would take note of for restarting. (not yet implemented).

LOOKAHEAD

This sets the interpreter look-ahead. It is normally set to 50 but may be decreased or increased if desired for specific reasons. Not normally required by a user.

SETNEXTEXEC

This is for continuing a job another day. Take not of the line number you hit stop on the day before and the next day you can do the following..

Load your file, reference the machine to 0,0,0. Type a number in the display line box, the file will follow you to that line. Press SetNextExec and the file will run in simulation to the required line. It will then ask if its OK to move to that position necessary to begin again. You should manually start the spindle from hot keys or by clicking the signal name for the spindle control and then press RUN. The program will continue as it left off.

ORIGINS:

This display will flash whenever a machine coordinate system is in effect which is not the true machine 0,0,0 system.

ESTIMATE:

Pressing this button will estimate your job time. It takes about 20 seconds per hour to check the actual time of job. This estimate is very close to the real cutting time. Typically it is less than 5 minutes out on a 4-5 hour job.

EDIT FILE

This will open your editor (set on the logic tab) and allow you to edit the file. Saving and exiting the editor will reload the file automatically.

MDI Line: This is the Manual Data Input line. You can type raw G-Code here. Try typing G0X10 and the motor should move 10 units of the currently selected coordinate system. If it move further or not far enough see troubleshooting.

ELAPSEDTIME: This is the time the job currently running has been running. If you charge by the hour of cut time, this is the billing time. If not, it's a neat clock thing that accentuates your screen in a nice shade of green.

FOLLOW EXECUTION: This selects if your screen will follow the running program or if you want or be free to scroll through and edit or look at your code as the programs runs.

SHOW NEXT EXEC: This will move the G-Code display to the next line to be executed just in case you have scrolled away and forgotten where you were.

FEED OVERRIDE: As it says, this will speed up your job, but be careful, in addition to a time lag of about .8 of a second, this bar will also speed up rapid moves and may exceed you motors capabilities. It will incorporate a speed limiter some day soon, but as yet has the capability to go too fast if you let it. In future it will be controlled by G-Code to turn it on and off as well.

GCODE DISPLAY: This control displays your Code, allows you to simple edit the code and will show you what line is executing. Simple edit means you cannot change the length of a line, but may change any numb er sin the line. For example a G0X55.6 could be changed to G0X12.2 but not G0X34.567. This is not meant to be an editor, (one of those will come) but just a quick change facility for debugging purposes.

SINGLE: Execute the current line and jump to the next one.

REWIND: Consider the G-Code file to be like a tape. You play it, and then rewind it. This key resets the job and gets it ready to run again. It will turn off any tool change in progress or any state of the engine currently waiting for processing.

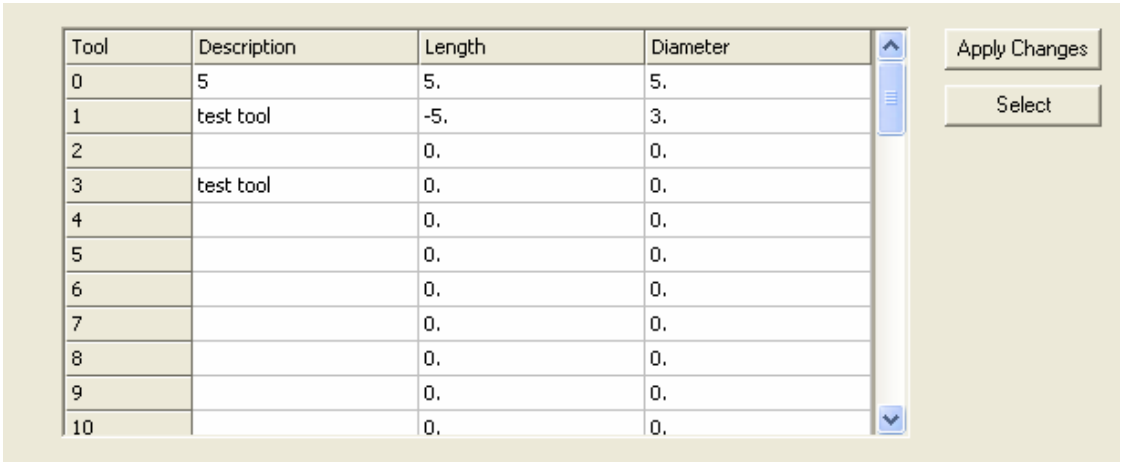
PAUSE: Ramp down to zero any cut in progress. This will stop in the middle of a line with a controlled stop that can be recovered.

RESUME: continue button for the above pause. It will ramp back up to speed.

STOP: Stops a job at the end of the currently translated lines. This may be several lines ahead of where you are currently at, but allows for a stop at the end of a line. If you wish to stop and then restart a job tomorrow, this is the key to hit. Record the exec line number from above and enter it intot the restart at box to start from that line. (Not yet implemented)

Tool Tables:

:



This one is pretty obvious. It allows you to enter up to 50 tools which you can select from G-Code to use for radius compensation and length compensation. Make sure you “Apply Changes” before leaving the program or trying to use new values in your code programs.

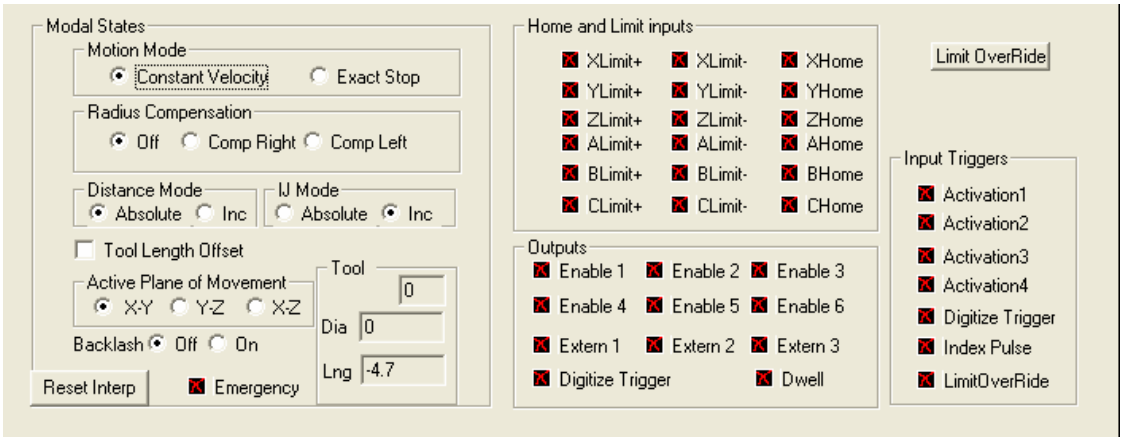
Using the select key will immediately select that tools values into the engine state.

Selecting activate while a tool is selected will turn on tool length compensation with that tool. Selecting Save Touch, will store a new length. Simply move the new tool down to the stock, and select store touch, this will store a value which would make the current Z point to be zero.

Engine State:

:

Mach1 is a state machine. That means it remembers what you have done as if you had flicked switches to turn on various modes. The values on the above screen indicate which switches are currently active. The check-lights on the signals do not represent highs on the pins, but activated signals. If you specified the signals as low active, a green check will mean a low is on the pin, either input or output. This is a handy test screen for your switches, hit a switch and the check should light up. Make sure you don't have a flashing reset button first though, All outputs are turned off in a n E-Stop condition 25000 times a second. Inputs are not monitored except for E-Stop in that condition as well.



MOTION MODE:

Mach1 has 2 motion modes. Constant Velocity and Exact Stop. In Constant velocity, in order to create a fluid path at the most constant speed possible, as one axis ramps down, the next one ramps up, this will round the corners a bit but produces a much smoother and faster job. For photo engraving or fine detailing, the constant stop mode should be used. This is slower, but all the axis will stop prior to the next one starting.

RADIUS COMPENSATION

Radius Compensation is the technique of moving a tool-path to the right or left of the actual path by the radius of the tool bit. This is commonly used in Inlay's or just if the toolpath is designed without taking into account the radius of the bit cutting it. Think of this as the way to get an actual 50mm circle out of a piece of stock, if the tool-path is 50mm's round, and you don't use compensation the circle will be too small by the diameter of the bit. These selections are on the screen for future use in the DXF module for inlay techniques. In general, in order to use offset properly, you usually use a cam program to generate the G-Code which will turn on this mode automatically. This is due to the complexities involved in figuring out the lead-in and lead-out moves. Refer to the EMC manual for proper explanation of lead-in and lead-out under radius compensation. Many people have wondered why they aren't compensating properly and 90% of the time it is because they don't understand the lead-in and lead-out parameters.

DISTANCE MODE:

Distance mode if the setting for the interpretation of distance in the G-Code. For example the statement G00 X10 can mean for the tool to move to an X Coordinate of 10 (absolute) or to move to a location in the X plane 10 forward of the current position (incremental). Usually, this parameter is left on absolute.

IJ Mode:

This is one you may not have seen before. When specifying an arc with a G2 or G3 command the I,J parameters indicate the offset from current position of the center of the desired arc. I = X offset, J = Y offset under most circumstances. These setting are sometimes in absolute and sometimes in incremental values depending on the CAM program used. While one would thing that in an absolute program the I,J's would also be absolute, this is not always the case. Normally you can leave this alone, but if you see circles being drawn very large instead of small arcs in the tool-path display, this is usually the cause, select the other mode.

TOOL LENGTH OFFSET:

Tool length offset is a method of correcting a G-Code path to correspond to a shorter or longer toolbit. It can be selected by the G43 command as in G43H1 meaning that you wish correction to be applied from tool #1. The system then takes the length of Tool #1 and adds it to the Z movements of the program. G49 will remove the correction. Refer to the EMC documentation for implementation specific details.

ACTIVE PLANE OF MOVEMENT:

This is an indication of the current engine plane selection. This is an advanced users switch and you should refer to the EMC doc's for details of the different planes and their usage.

BACKLASH:

You can turn on or off backlash here. This is remembered between program runs and there are cautions. When first selecting backlash you will notice the mode change from constant velocity (if selected) to exact stop. This is because a backlash corrected machine should not be run in constant velocity. It is necessary to stop at axis change points in order to apply the compensation. Since some machines CAN correct fast enough not to lose pulses, you can select constant velocity mode again after hitting backlash. The amount of backlash and speed of it is set in the configuration menu.

RESET INTERP:

This is used if you have stopped a complex program in the middle. Many of the state machines internal toggle switches may be in unknown states due to G code modals. This button will take you to turn on condition.

HOME ANDLIMITS:

These x's will change to checkmarks when the signal is active. This is irregardless of the polarity of the required signal and a green check means activated.

OUTPUTS:

These checkboxes indicate the state of the output signal. Clicking on the signal will force its state to change. Be careful here as Spindles and other devices may be activated just by clicking a checkbox.

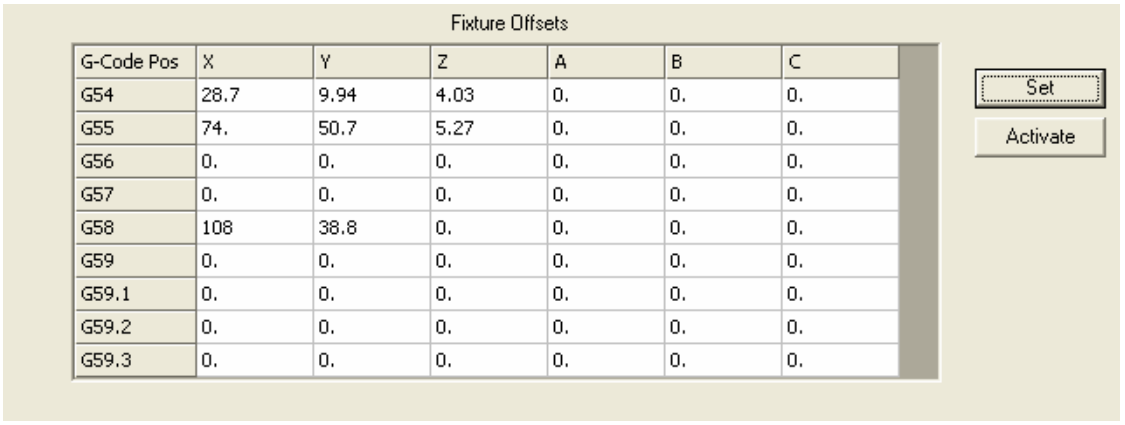
INPUT TRIGGERS:

These are the status of selected input triggers. A Checked green means it is currently being sensed as active.

DISABLE LIMITS:

This button will temporarily turn off the limit switches so you can jog off the switch. Be very careful here as the button disables ALL limit switches until you move the offending axis off the switch, at that point the limits are re-enabled.

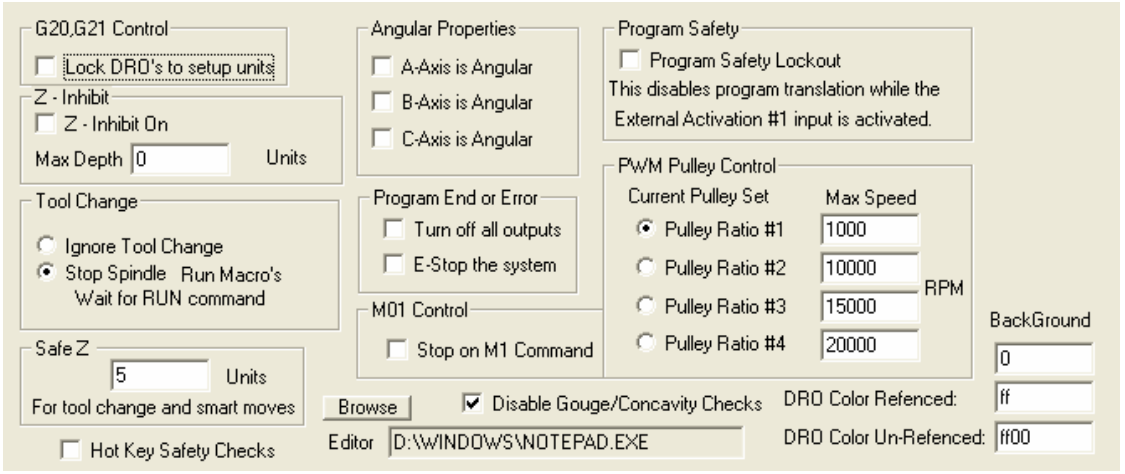
Fixtures:



Fixtures are a way of storing multiple home locations within your system. Consider a milling table with several fixtures for holding stock. You can save a home location for each of these fixtures and switch between them in G-Code or on the screen. The G-Codes to turn on a home location are on the selection buttons. Selecting G58 above for example and then pressing activate, will make a new home location of 108, 38.8. If you were at 0,0 when you pressed the button, the DRO's will now read -108,-38.8. Using an MDI command like G0X0Y0 will now take you to the new home location. There is no method of retracting a home position, though you may select another. If you wish to maintain a reference position of 0,0,0 in your machine , I recommend you keep 0,0,0,0,0,0 stored in at least one of the selections.

Pressing Set while a line is selected will store the current location as a new home in that line.

Ancillary Logic Functions:



This screen is used to set some of the variable options of the system.

G21,G20 control is used to switch between inch or metric. If you select Lock DRO's, the system will keep the DRO locked on your setup unit selection while the program executes properly in its proper unit selection. In other words, if your program is setup to program in mm's and you run a job which uses G20 to select Inches, then the DRO's will indicate 25.4 after a G0X1 move while in INCH mode.

TOOL CHANGE:

Ignore tool change means the program will register the change of the current bit internally, but will not stop for a bit to be changed.

Stop Spindle and run macro, will stop the spindle if running and run the macro "M6Start.m1s". By default this macro will raise the axis to SafeZ setting and wait for the run button to be pressed, at that time the macro M6End.m1s will be run which by default will return the toolbit to the position it was in when M6 was called. These macro's may be edited to accomplish more complex tool changes.

ANGULAR PROPERTIES:

If checked, the applicable axis will no longer respect English/metric conversions but will consider all angular moves to be in degree's.

PROGRAM END OR ERROR:

This will come into effect as a program executes the last move in a program, it can either E-Stop or clear all outputs. By using this mode , the system will turn off the spindle and any other outputs if, while running, it hits an error or stops due to end of program. (This includes M0, or M1's)

M01 Control:

This affects the G-Code behavior when M01 is encountered. You may use this at the end of a program run to automatically turn off your spindle and EStop which will shuut down the motor drivers.

Program Safety

Selecting this option will cause the translator to stop running G-Code unless the ExtAct1 signal is sensed to be active. This is used for things such as Spindle Cover safety. If the cover is not closed, you may jog around and reference and such, but a program will not run until the cover is closed.

PWM Pulley Control:

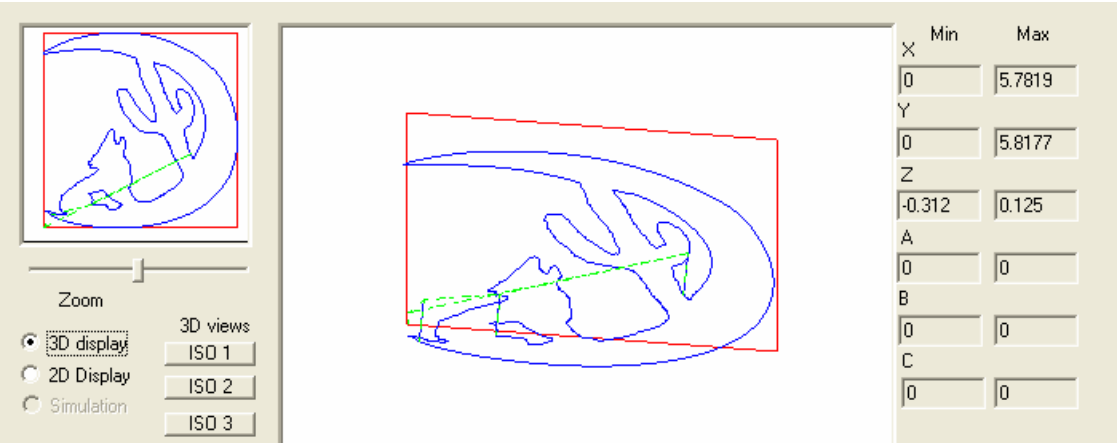
This is for machines using PWM (Pulse Width Modulated) motor speed control. Since the speed of a spindle changes in reference to the 0-100% PWM signal, this will set 4 different speed ranges. The program must know what spindle you are using to effectively set the correct PWM rate for the desired speed.

Colors:

These are used to set the colors of the DRO's, forground, background and referenced colors are available. A dro will change to the referenced color if the axis is homed. Restart the program for these colors to take immediate effect.

ToolPath Display

:



This display shows you your tool-path. The small window on your left is an overview window, it shows the complete job, the large window on the right is for detail zooming. By sliding the Zoom slider to the left you will see a black box get smaller. This is the display window into the detail box. Clicking and moving the mouse in the overview window will move the zoomed window around the job. The Green dashed lines are rapid moves while the solid blue lines are the feed rate moves.

The wide red line is the actual movement of the tool bit in real-time. This is cleared when switching windows. It is meant to be used as a diagnostic tool to check your path. Its line width size is set to .8% of the overall size of the job. This is to make it easier to see in large or small jobs without zooming.

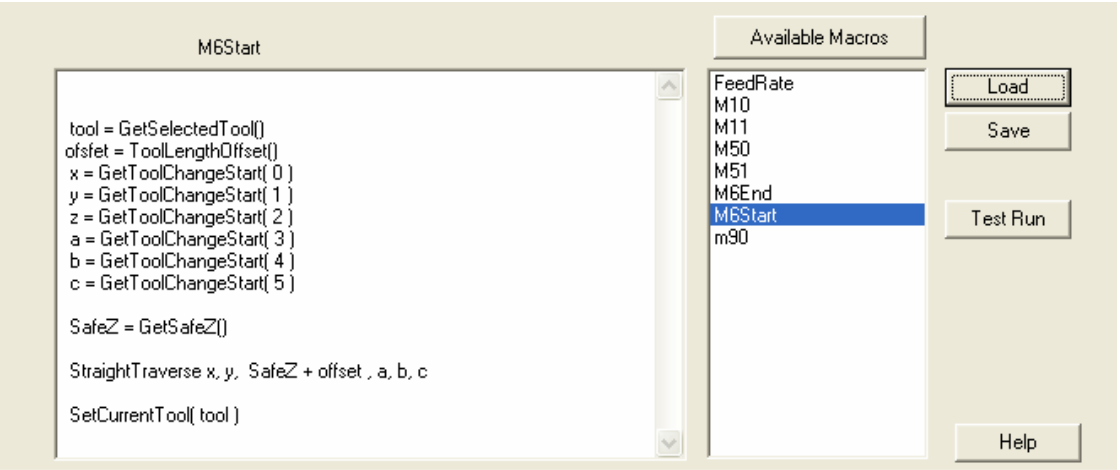
EXTREMA:

This is the maximum extent the axis will move during the current jobs run.

2D/3D

This will change the view to a rotatable 3d view from 3 different ISO perspectives.

Macro’s



The Macros screen is used for editing, creating or testing a macro. The one you see in the picture is the macro used to start the tool-change process. It is Visual Basic Scripting.

Any macro may be created and called from this screen. The macros 50 – 97 and 9 and 10 (ex M10) are called when G-Code calls a command of the same name.

Ex.

G0X0

M10 // the M10 script will be called at this point.

G0y10

The example in the window is simplt to understand when you look at it carefully.

Tool = GetSelectedTool() // this tells the program to get the last selected tool and put it into a variable named tool.

Offset = ToolLengthOfset() // same here, get the current toollength offset and put it in offset variable.

X = GetToolStart(0) // this gets the X-Axis position when the toolchange started. Axis are numbered from 0 – 5 (x,y,z,a,b,c) in all functions that call for an axis.

The line StraightTreverse x,y,SafeZ + offset,a,b,c will make a traversemove (rapid) to the location x,y,safeZ,a,b,c

The reason the offset is added to the SafeZ is that tool length offset is not considered in the call to mnove, the call is sent absolute coordinates with no corrections internally except for origin offsets which are applied automatically.

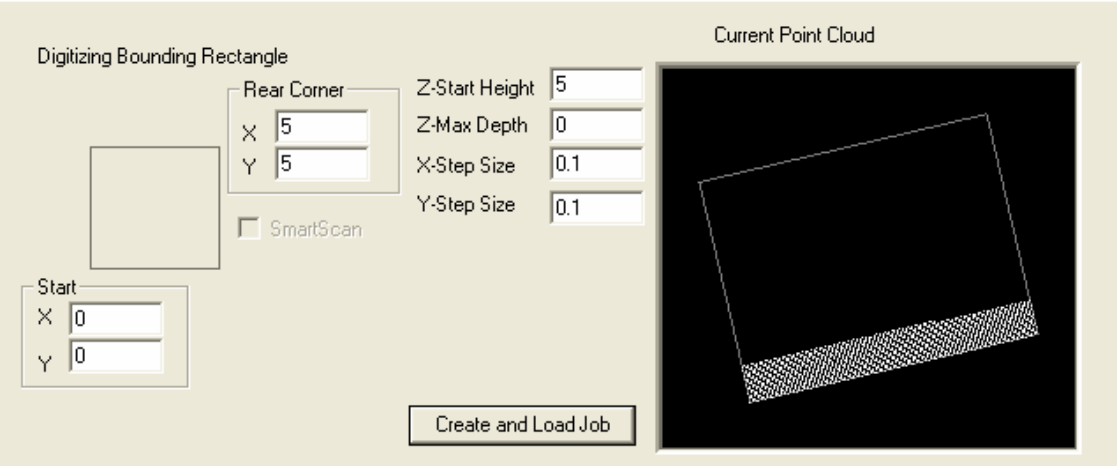
Pressing the Help button on this screen will display a list of current commands. More will be added over time as requests for functions or necessity of having a function are discovered.

Software Limits

The software limits settings are on the Jogger page. Pressing the Software Limits button will allow you set a maximum travel of any axis. This maximum only applies to an axis if it is in Reference mode. Once referenced, the axis will trip an E-Stop condition if it exceeds the maximum distance set from the home switch or tries to go below the zero position.

In the case of a Z axis, which is normally homed to the top of its travel, the E-Stop will trip if that axis tries to raise higher than home, or if it tries to exceed the max travel downward. Origin Offsets, Tool Offsets or other coordinate systems have no effect on this calculation.

Digitizing



The digitize Wizard allows you to create a grid for digitizing. The start location is the 0,0,0 point of the grid, the rear corner is literally the rear corner of the digitizing grid. The step-over variables are how often to take a sample.

Pressing create and load will create a G-Code file which can digitize a job. You will be asked for the name of the G-Code file. The job is done at rapid rate except for the probe moves G31 which are done at federate. The user should manually set the federate by typing a F command in the MDI window prior to pressing run. Ex. F600

When the run button is pressed, you will be asked for the name of the triplet file to create. This file will contain all the coordinates which occurred at the time of the probe activation. The 3d coordinate Point Cloud may be used in various CAM programs to create models from which to create a toolpath.

The M40 and M41 command is used to control the opening and closing of the triplet file. (See macro involved for usage.)

G-Code Specification for Mach1

The following G-Codes are understood by the MAch1 interpreter. That is not to say they have a function, but they are understood and those not yet functionsal will be turned on at some point.

This is a plain list of the known codes to Mach1, a full explaination of each code follows this list .

G Codes	Short translation
G0	Rapid Move
G1	FeedRate move
G2	Interpolated Arc (CW)
G3	Interpolated Arc (CCW)
G4	Dwell (Delay)
G8	Set exact Stop Mode
G9	Set Consatnt Velocity Mode
G10	Set Fixture Offsets Origins
G17	X-Y Plane Selection
G18	X-Z Plane Selection
G19	Y-Z Plane Selection
G20	INCH coordinates
G21	MM coordinates
G28	Go to machine coordinate reference point (home switches)
G30	Same as G28 at present
G31	Probe Command
G40	Cancel Radius Offset Mode
G41	Radius Offset Left Mode
G42	Radius Offset Right Mode
G43	Tool Length Offset
G49	Cancel Tool Length Offset
G53	Force Machine Coordinate move.
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

G59.1	Use Fixture Offset # 7
G59.2	Use Fixture Offset # 8
G59.3	Use Fixture Offset # 9
G61	Exact Stop Mode
G61.1	Exact Stop Mode
G64	Constant Velocity Mode
G80	Cancel Motion Mode (Cancels Canned Cycles as well)
G81	Drilling Canned Cycle
G82	Drilling with Dwell
G83	Drilling with Chip Breaking
G84	Right Hand Tapping Cycle
G85	Boring Cycle #1
G86	Boring Cycle #2
G87	Boring Cycle #3
G88	Boring Cycle #4
G89	Boring Cycle #5
G90	Absolute Distance Mode
G91	Incremental Distance Mode
G92	Offset Coordinate system
G92.1	Cancel Origin Offset
G92.2	Set Axis Offsets to zero, keep parameters.
G92.3	Set Axis Offsets to zero, zero the paramters
G93	Inverse Time Feed Mode
G94	Feed Per Minute Mode
G98	Call Subroutine
G99	Return from Subroutine

Interpreter Logic

In this section we'll deal with the various G-Codes Mach1 understands and look at examples of how to use them.

G-Code is a language that controls a state engine. A state engine is a software block which functions mush like the dashboard of a vehicle. In the case of Mach1, that vehicle is a CNC interpreter. Each G-Code command tells the interpreter to do something. It doesn't have to be motion that the code is asking the engine do. It could be setting a parameter for future use, setting a variable or setting a speed.

You must keep in mind that the state of this dashboard is cumulative, flicking a switch or setting a setting will keep that setting in effect until you switch it off. For example, executing a G20 command will set your engine to interpreting all movements as Inches, until you cancel it with a G21 to switch it back to mm's. (My apologies to all you English folk's, I'm a metric person).

In much the same vein, when issuing a movement command, the interpreter makes the assumption that the move is from the current position, to the commanded position. This makes it much easier to tell a machine to move to an X of 10 and a Y of 10 (G00 X10 Y10) because you don't have to tell it where it is to start with. I only mention this because I have seen people cut and paste G-Code without the recognition that they may be invalidating the entire section of the code, due to the fact the starting position of one line is not where they expect. It also can explain why you will sometimes not see a command on a G-Code line. Many commands that you execute act as switches so a G01 for example (federate move) , once activated will stay in effect. So a command string as follows is perfectly valid and calls for more than one move, one at a time at federate speed.

G01 X10

Y6

Z7

It should also be noted that multiple axis on a single line call for a coordinated simultaneous move all of the axis involved.

Well, with all this in mind, lets take a look at the basic commands that make up the Mach1 Engine. We'll deal with the G-Codes first and then move on to M commands (Modal Commands) and Command words

.

Example G-Code commands:

G0 Rapid Move

G00 is the command for a rapid movement. It asks the interpreter to move an axis or collection of axis's to a particular point at the highest speed possible. That highest speed may be regulated to a lower speed under certain conditions, but the command calls for the movement to be as fast as possible.

Ex: G0 X10 Y10 Z3 A4

Move quickly to X coordinate of 10, Y of 10, Z of 3, A of 4 (inches or mm's depending on your current distance mode. (See G20, G21 for distance modes)

G1 Feedrate Move

G01 is the command for a feedrate move. This asks the interpreter to move at a speed you have previously set. Other than that it is identical to the G00 move. (See F-Words for feedrate speed settings).

G02- CW arc move

G02 is the command for an clockwise interpolated Arc move. All G02 and G03 moves are done at federate speed. It makes no sense to ask for a rapid move in an arc as rapid implies you wish to move quickly and the shortest distance between two points is a straight line.

This command can be used several different ways with many different parameters and can perform two or three dimensional moves in any of three different planes of motion. It is also capable of performing helical arcs by setting parameters to tell it the ending coordinate of the Z axis (or third axis when using other planes. See G17, G18, G19 for plane settings.)

The parameters tell the interpreter the end of the arc, and the radius of the arc by either setting a radius (R word) or setting a center point (IJ words).

Ex: G02 X100 Y0 I5 0 J0 K 10

This is a helical arc. Lets assume that the current location is 0,0,0 (x,y,z) and this line is entered. The unit is to go to the end coordinates, in this case X100 Y0 Z 10 in a clockwise arc with a center located at an X of 50 and a Y of zero. The Z end coordinate is taken from the K coordinate. I have often been asked why they are labeled I,J and K and my answer is usually, “they couldn’t call it X Y and Z because you’d have more than 1 of each on the line (which is a no-no), but the real reason is that if you using different planes of motion, the parameters may be referring to a different combination of axis altogether.

We will assume an X-Y plane for the rest of this document’s examples. The I,J may be used as absolute coordinates, or as offsets from the current point depending on the I,J mode selected on the logic tab. That setting is persistent and you need only set it once if you find it does not match your CAM programs output. Some CAM programs put out incremental I,J’s, some put out Absolute I,J’s , its hard to tell what is used sometimes, so just take note that if your small arc’s look like large circles, the setting is probably wrong.

G02 X 100 Y0 K10 R50 is the same as the above line, but it uses an R word to specify a radius of 50mms’ (or inches).

G03 – CCW arc move

This is the same as G02 only with the direction reversed. It moves in a counterclockwise direction. All other aspects are the same.

G04 Dwell

Dwell is a pause. You may wish to pause your tool to cool it off, or perhaps pause for a torch to heat up. You simply supply a parameter to the Dwell command to set the time you wish to wait. This delay is set in seconds to a resolution of .0001 seconds.

EX: G04 P2.3 (this waits for 2.3 seconds before executing the next line of code.)

G08 Constant Stop (Exact Stop)

This command will set the different velocity modes of Mach1. Mach1 only has 2 modes. The Constant Velocity Mode and the Exact Stop or Exact Path mode. The two modes are used for different types of movements. Most jobs you wish to cut have two attributes. Time and Accuracy. Both are important, but both are mutually opposed. Accuracy takes time, and speed has a cost in terms of accuracy. As life is full of compromise, so too is deciding on the use of these modes. Constant Velocity begins the ramp down of one axis as the next moves axis ramps up. This lowers accuracy by creating a small rounding of sharp corners. It also improves speed by a large factor. Some jobs, like engraving , do not mesh well with this rounding, while others, like sign cutting, enjoy a great increase in speed while not noticing the rounding error.

The amount of rounding you experience in Constant Velocity mode is a factor of your acceleration and deceleration profiles. The slower you accelerate or decelerate, the more rounding you will have. You will find, through experience, that both modes are very valuable and each has its purpose. This mode can be changed by G-Code or on the Sate tab in the program. G8 sets the Exact path mode. This mode does each line, one at a time, ramping up, moving and then ramping down. It is very accurate.

G9 Constant Velocity

This mode maintains as constant a velocity as possible during the run of the program and greatly reduces the time required to complete the job. Not recommended for engraving, but great for roughing or most sign cutting.

G10 Set Fixture Offset coordinates

This is to set the Fixture Offsets coordinates. Fixture offsets are a way of re-zeroing your machine in several places. For some reason this command is forced to carry the L2 modifier. It is used to set any of the 9 fixture offsets in the following syntax.

G10 L2 P1 X110 Y45 Z3

This means (G10L2) set a fixture (P1) , make it fixture number 1, and set it to X110 Y45 Z3

This command only sets the internal parameters to those numbers. When Mach1 is restarted the fixture offsets will return to pre-specified defaults from the fixtures page. G10L2 is the command, the P word indicates the offset number (1= 54, 9 = 59.3) (see G54-G59.3).

G17 X,Y Plane selection

Selects the XY plane as the arc interpolation plane. When specifying and arc, you can tell the interpreter to draw the arc in the XY plane (normal x,y coordinates) or…..

G18 X-Z Plane Selection

You can set the plane to X-Z so the arc is drawn upwards or downwards in the X direction, or….

G19 Y-Z Plane Selection

The arc can be drawn upwards and downwards in the Y direction.

G20 Inch Coordinates

This command select inchess to be used. When activated, the current DRO readings are converted to inches if they were in mm's and all further coordinates are assumed to be in inches's.

G21 MM Coordinates

This command select mm’s to be used. When activated, the current DRO readings are converted to mm’s if they were in inchess and all further coordinates are assumed to be in mm’s.

G28 **Go to machine coordinate reference point (home switches)**

This will move the axis's which had been previously referenced to their switches to a position just in front of those switches. It is commonly used as a clear command to move the tool away to a known place for loading and unloading a table.

G30 **Same as G28 at present**

G31 **Probe Command**

The probe command begins a move to the specified location but will stop the move in a controlled ramp down when the digitize input is activated. The positional coordinates when the Digitize signal activated are stored in a triplet file, if one was previously opened, and stored in the interpreter in variable locations 2000, 2001,2002,2003,2004,2005 for axis x through C respectively.

Ex:

G31 Z10 …This will cause to begin a federate move to 10. It will ramp to stop when the digitize signal goes active and the results stored in both the triplet file (if opened) and in variable #2002

G1 Z#2002 // this would then move the Z axis to the point where the signal went active, thus the combination of

G31X10

G31X#2000 would cause X to move until a probe is hit, and then return to that position where the probe switch just gets made.

Only one axis may be specified as a G31 move at a time. You are free to do calculation based on the variable contents and moves to those variables in order to create auto centering macro's and edge finding macros.

G40 Cancel Radius Offset Mode

This cancel the Radius offset mode if it is in effect. (See G41, G42), this move should be used with a lead-out move specified.

G41 Radius Offset Left Mode

This sets the Radius Offset Left mode. This tells the toolbit to keep a distance of the current toolbit radius from the toolpath on the left hand side of the toolpath as viewed by the movement direction. This command should be combined with a lead-in move.

G42 Radius Offset Right Mode

This sets the Radius Offset Left mode. This tells the toolbit to keep a distance of the current toolbit radius from the toolpath on the left hand side of the toolpath as viewed by the movement direction. This command should be combined with a lead-in move.

G43 Tool Length Offset

This move specifies an offset based on a toollength to be added to all future moves in the Z axis.

Ex. G43 P2 tells the controller from this point forward, add the length of the toolbit entry #2 to any further Z moves. . No movement is done and the DRO corrects to current position. This move should be followed in most circumstances with a correction move to a known coordinate.

G49 **Cancel Tool Length Offset**

This cancels any offset in effect. No movement is done and the DRO corrects to current position.

G53 Force Machine Coordinate move

.

This is a forced coordinate move. Lets say you have zeroed your machine at a particular coordinate, and then later instituted a fixture or origin offset. G53 is a way of making a move in the coordinate system without the offset being in effect. G53X10 for example will not move to the fixture offset coordinate systems 10, but to the 10 in the zeroed system you started with. Take note that this is NOT the machine absolute coordinate system based on the home switches if you have zeroed elsewhere, but to the coordinate system you zeroed to prior to program run.

G54 - G59.3 **Use Fixture Offset # 1**

This command will implement a coordinate system zeroed at the location specified in the Fixture tables. These coordinates may be modified manually on the fixture page or set by G-Code by the G10 L2 command. (See G10 command). Once the fixture offsets are turned on, the machines further movements are to those systems coordinates. You may use any of the 9 coordinate systems offered.

G61 **Exact Stop Mode**

See G8

G61.1 Exact Stop Mode

See G8

G64 **Constant Velocity Mode**

See G9

G80 Cancel Motion Mode (Cancels Canned Cycles as well)

This puts the interpreter in a state where no motion is allowed. It is the default turn on state of the interpreter as well. Using a motion word like G1 will put the interpreter back in a motive state. While in the G80 state a command like “X12.9” will not move the tool. The interpreter must know it is in a motive state prior to simply entering a movement command as above.

G81	Drilling Canned Cycle
G82	Drilling with Dwell
G83	Drilling with Chip Breaking
G84	Right Hand Tapping Cycle
G85	Boring Cycle #1
G86	Boring Cycle #2
G87	Boring Cycle #3
G88	Boring Cycle #4
G89	Boring Cycle #5

G90 Absolute Distance Mode

This command selects an absolute coordinate entry state. All coordinates in movement commands are assumed to be offsets from 0,0,0,0,0…

G91 Incremental Distance Mode

This command selects an incremental state where all input coordinates are assumed to be incremental moves from the last known position.

G92 Offset Coordinate system

This command selects an offset origin base from the current zeroed position. It is used in the context of a fixture but is separate from fixtures in that it may apply to that fixture offset.

Ex. G92X10Y5 (This sets an absolute origin offset of 10,5 to be added to all subsequent moves.)

G92.1 Cancel Origin Offset

Cancels the above offset correction and resets the DRO’s to proper coordinate display.

G92.2 Set Axis Offsets to zero, keep parameters.

Set the coordinates of an origin offset to zero while keeping the offset in effect. This has the same effect as releasing the offset. The offsets previously used may be turned on again by the use of the bare G92 command.

G92.3 Set Axis Offsets to zero, zero the parameters

This command will erase the current origin offset and zero the parameters of that offset so they may not be used again.

G93 Inverse Time Feed Mode

G94 **Feed Per Minute Mode**

G98 Call Subroutine

This command will call a subroutine from within the current G-Code program. Subroutines are numbered from an O word as in the example below:

```
G0X1
G0Y6

G1X56.6
G98 P1234          // call subroutine #1234
M30
O1234              // this declares the subroutine as number 1234
G0X0
G0Y0
G99                // return from subroutine.
```

G99 Return from Subroutine

See command G98