BASICS

The CNC Controller is a software program called Mach3 Mill and provides for controlling all the movements and associated commands. The axis directions are shown next to the controller in Figure 1. The CNC controller (Mach) does not know a practical Machine Reference Point to work from, thus, the controllers "0,0,0" is somewhat meaningless and has unlimited movement capability. When the controller is turned on it monitors axis movement starting from it's "0,0,0" value for each axis and displays the absolute movement. Machine X ,Y& Z=0 can be anywhere and can be set by the user manually, via switches, and other ways

Machine Zero is a fixed point within the machines travel limits and does not normally change. It is typically called Machine Reference point, machine zero, or simply Home Position. Machine zero is set by the user. Home location is a user choice but there are standard recommendations on where to set it. Thus a user defines a location for Home to suite his individual machine setup.

Program Zero, or a better description Part Zero, is a reference point defined during coding of a program. It is not known by the machine until defined. It is the origin for all coding in the program. For the mill, Part Zero is defined by the user as a distance from Home to the part.

So the light blue grid portrays the controller X/Y plane. The green grid shows a possible location for a controlled point and part relative to the max movement of the tables. The controlled point is the center and end of the tool in the spindle. The location of the part away from Home is called Part Zero. The above is graphically depicted in the figures below.

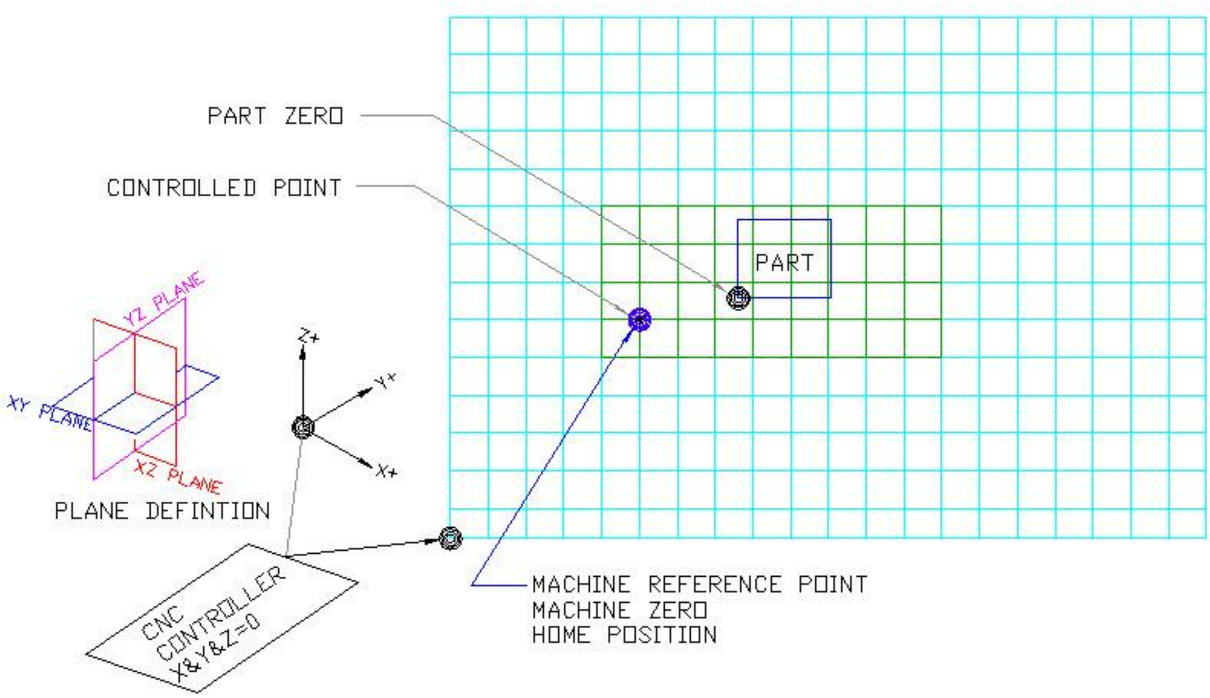


FIGURE 1

WORK OFFSET

Milling machines use Work Offsets and they are called Fixture Offsets in Mach. The fixture offset has a zero reference point (Datum) and a part within the fixture can have a temporary offset whose reference point (Datum) is relative to the fixture reference point. A Work Offset is used to define where the part is in relation to Home. The distance is always defined from Home to the Part.

G54 is the default work offset when Mach is started. It's value can be zero depending on setup. The G54…to G5x are called Global and there are other offsets.   G52 is an offset Local to the G54. G52 is a temporary work offset. And there are G code commands to change the work offset or tool offsets.

The programmer can make use of the offsets depending on what and how something will be machined. This is graphically portrayed in FIGURE 2 where the green grid is the max movement of the tables .

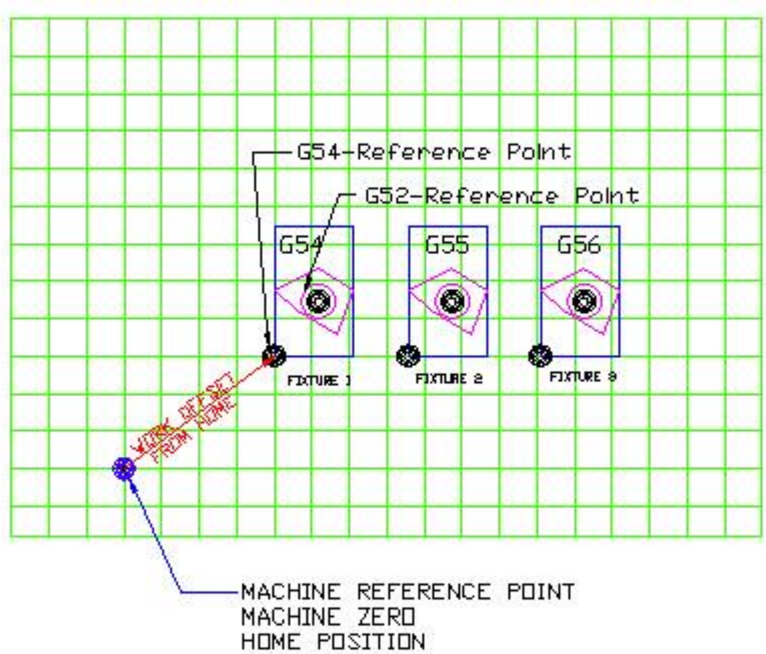
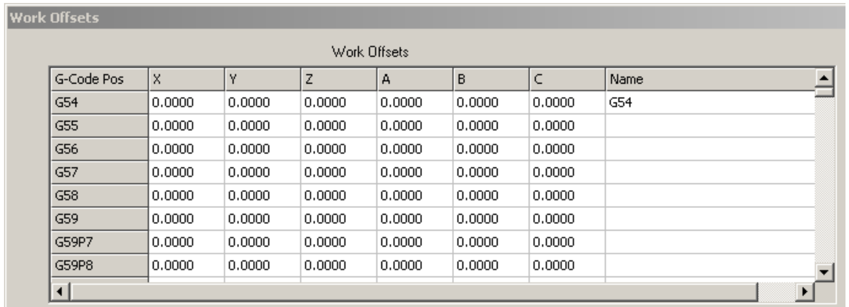


FIGURE 2

Mach 3 Mill stores the Work Offsets for each G code position in the Work Offsets Table shown in Figure 3.



Work Offsets

G-Code Pos	X	Y	Z	A	B	C	Name
G54	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	G54
G55	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
G56	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
G57	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
G58	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
G59	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
G59P7	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
G59P8	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	

FIGURE 3

NOTE:
There are different kinds of offsets used for different tasks and basically used in program coding and give the controller instructions. The word offset is used rather loosely and one needs to be distinct in the description when communicating. A good example would be saying that clicking a button on the mill screen will create an offset.

BASIC GCode COMMANDS (WORK OFFSETS)

G54-G59 and G59 P1-254

The G54 – G59 offset is a defined Datum point with their offsets from "HOME to Work / Part" and can be used to relate two coordinate systems. ie; Home position (machine coordinates) to part position shown in Figure 3. Not to confuse, but, the reader may also find work offsets called fixture or work coordinate systems.

Work offsets provide for changing / adjusting an offset without affecting the other offsets. Initial offset of G54 are and should be based on Part zero and allow movement / enough offset value to a tool change location.

Here is my simplified version of Mach's 255 Work Offsets:

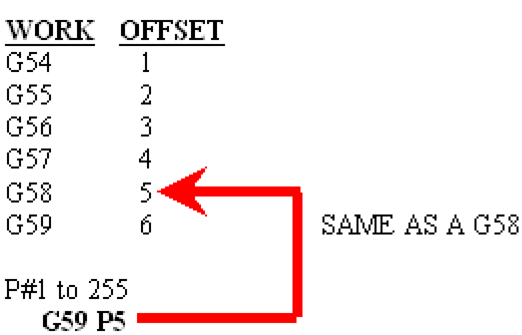


FIGURE 4

Mach Code description – G59   Command (shown in FIGURE 4)

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.

Another example of use would be making multiple parts   using the same code by providing an offset to do the other parts. ie; G54 offset used for the first part and then G55   (offset amount) such that the tool would start at a different Datum.. (See Figure2).

G10 Command – (G10 L..P.. X.. Z..)

G10 can be used to adjust the work offset and is a Permanent change to the offset from one value to another.
G10 **L2** (designates that it is work coordinate) P (is the work offset Number) X…Y…Z…. is the value
------Do not confuse above with G10 **L1** which is a tool offset change designation ------
The G10 will permanently change the work offsets (G52 will not as it can be cancelled).

G52 -Work Offset   (local offset)

Within the G54 (Global) one   can define a "local" reference location relative to G54, namely,   G52 which is some distance from the G54 X,Y,Z location. The G52 is the distance from G54 0,0,0   to the part / tool *location.*

- G52 replaced a legacy gcode of G92. It is still used, but, note that G92 is the reverse of G54 such that G92 is the distance from the part to the tool location.

- G52   is   a temporary offset - you need to have a work offset in place since it offsets the original offset by some value and can be cancelled thus the original offsets are not cancelled.

- Do Not use both G52 and G92 at the same time.

- G52 X1 Y1 Z1. is a datum shift by the amount in X,Y,Z, if the values are ZERO it is shift cancel. So G52 X0 Y0 Z0 A0 B0 C0 Cancels any G52 local offset you have set. If you do not cancel them you will get accumulations of offset and things like go to tool change go to the wrong place. You do NOT want that.

The   G52 just offsets the controlled point by some amount and remains in effect until canceled.

G92 -Work Offset

- G92 is an old / legacy command and was basically replaced by G52 along with the G54 work offset.
- G52 is local to G54 and G92 is Global, G92 actually resets internal parameters when in use.
- Do NOT to use the legacy G92 when other offsets are applied.
- Do Not use both G52 and G92 at the same time.

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.

COORDINATE SYSTEMS

There are different coordinate systems. A user will find different names from different sources and even within Mach3 software. So the following descriptions are based on the Lathe Screen for awareness. The following is a brief description of each.

<u>MACHINE COORDINATES (MC)</u>

Remember that the CNC controller (Mach) does not know a practical Machine Reference Point to work from, thus, the controllers internal reference is somewhat meaningless and has unlimited movement capability. When the controller is turned on it monitors axis movement starting from it’s internal value for each axis and displays the **absolute** movement inclusive of current setup. For CNC machining the + or – are indicative of the direction as related to the distances from the what the controller knows. Machine Coordinates are absolute distances from a fixed point (Datum). (see Figure 1 & 2)

<u>PART COORDINATES (PC)</u>

Part Coordinates are distances from a fixed point which was defined in Machine Coordinates. The controller needs to know where the part is relative to machine coordinates. **(** see Figure 2). The part zero can be located anywhere.. The part has coordinates associated with it and location is set by the user. The software can’t properly control the machining of the part if it doesn’t know where the part is. Part coordinates can be equal to or even the same as MachineCoordinates. Thus defining the part location is very dependant on how a user sets up a job including how the gcode program was created. So part coordinates are used to relate or define where the part is relative to the machine coordinates.

In general, only the user can define where the part is located to the controller.

PROGRAM COORDINATES (PC)

The Program Coordinates are like Part Coordinates but include any G92 / G52 offsets. So if there are no G52 or G92 (one or the other / not both) offsets then Part = Program Coordinates. If there are G52 or G92 the DRO will display tool location to include them.

NOTE: The "Program" contains all the Gcode instructions for machining. The controller uses the program instructions to control tool and table movement and additionally other associated hardware.

MACH3 MILL SCREEN COORDINATES

The only coordinate system noted is Machine Coordinates and the DRO's show current position of the controlled point unless the Machine Coord's button is turned on.

FIGURE 5

The current position of the tool (controlled point) is defined by the following formula based on current values. Figure 6 is from the Diagnostics page of the Mill Screen.

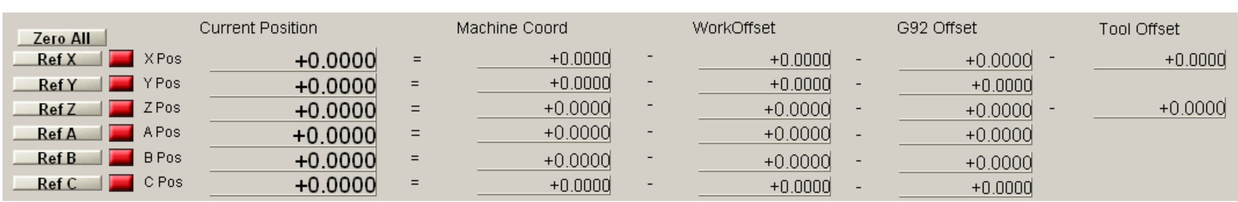


FIGURE 6

The formula can be manipulated:

Current Position = Machine Coord - Work Offset - G92 Offset - Tool Offset
Machine Coord = Current Position + Work Offset + G92 Offset + Tool Offset
Work Offset= Current Position - Machine Coord - Work Offset - G92 Offset -Tool Offset
(typical for X Y Z A axis and values can be + or -)

What is shown is the DRO's for the axes are dependent on how Mach has been set-up.
Remember the controller is dumb and it only knows what the user tells it.
It all depends on the level of automation at start up and how the user wants to work.
Consider the following progression of controller conditions at start up:

- All manual
- Software used to automate
- Software and hardware items used to automate
- Software and hardware items used to automate and other items like Macro's, etc.

COMMENTARY ON ALL OF THE ABOVE

The above is "REQUIRED" basic understanding. What's written is rather concise and how to do something WAS NOT provided. If you are having trouble at any time in comprehending what is said go back and re-read what is written as it progressively builds on each section. CNC has terminology that should be used to communicate properly but like language there can be dialects.

++

So my general comment would be to consider providing the practical "how to" do adding specifics which build / relate from your experience on the basics. There are few more basics that can be concisely defined and graphically presented for comprehension. IE; Tool table, limits and switches, set-up implications come to mind.

A way of presentation for "how to" would be starting with a manual setup, a test of understanding can be using the diagnostics screen, changing values and button clicks….

But most of all, you need to decide what, how much, level of description, etc and the logical progression of what how it will be covered. ie; Don't try to teach someone to write a macro, maybe add appendix for parameter info.

The simplest of subjects can be a book…….

Just some thoughts / comments on it all,

RICH

++

Coordinate Spaces - a Guide
Roger Caffin

Understanding all the work spaces and offsets system in Mach3 can be bewildering. The Mach3 Mill manual tries to explain all this in Chapter 7, but sometimes it seems you need to have fully understood Ch 7 before reading it, so you can notice the bits in Ch 7 which really matter. It's a common problem, and the usual explanation is that the manual was written by the programmer rather than by a user. To make matters worse, the manual adds
'The final way of setting a work offset is by typing a new value into an 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.'
Needless to say, the simplest method of resetting the coordinates is just that: typing into the DROs on the screen. I am sure many people start this way without understanding Offsets (I did). It works just fine for any simple machining task.
Unfortunately, if you want to do any macro programming, you find you cannot (easily) enter data into the DROs directly. Actually you can, using two different methods, including an undocumented feature which we will cover later, but not understanding how Offsets work can cause massive grief when your programming gets more sophisticated. So you have to RTFM. And you have to understand the difference between Machine Coordinates, the Controlled Point, the Workplace Offsets, 'other' Offsets and the Tool Offsets. So we will try to explain how it works.
We assume you have some familiarity with Mach3 already, so a few short cuts are taken in presenting some commands. We end with some unexpected details about hidden parameters in Mach3 as they do relate. This may be the most useful part of the whole document!
Please note: this has been written for a Mill. Mach3 for a Lathe is a bit different: for a start the job spins around an axis. You can access various offsets via the Tool Table, but there is no Settings screen.

Controlled Point (Coordinate Space)
This is what any CNC program is ALL about. The Controlled Point is the middle of the tip of the cutter as it spins. You can think of it as being the tip of a tiny V-point cutter if this helps. A CNC program moves this Controlled Point around. The fact that the cutter is really a lump of very sharp carbide some 20 mm in diameter and spinning at 3,000 RPM is incidental: hopefully any material in the way will be removed without damage.
This does mean that issuing a command such as `G0 X10` from X=0 will move the Controlled Point to X=10. That is the whole point of CNC after all. Normally the DROs on the screen show this 'Controlled Point', so the command `G0 X10` will move the machine to where the X DRO shows 10 units. What really defines the value in the X DRO is more complicated. Note that while we talk about just the X axis, the comments really cover all the axes, even the rotary ones. I use 'X' as a short-hand.
Also note that you can switch the DRO display to show 'Machine Coordinates' rather than 'Controlled Point'. You will eventually need to do this. The equation across the top of the Diagnostics screen may help you understand a lot of this too.

Machine Coordinates
While the Controlled Point relates to the part you are machining, Mach3 really has no understanding of that. It only knows about the machine (barely) and how to move the axes

around. The movement of the axes takes place in a Machine Coordinate space, and this is what Mach3 uses for all its movements. The Coord Space values seen in the DROs on the screen are there mostly for user convenience. Keep this firmly in your mind.
But any coordinate system has to have an origin: a place where X=0 etc. This place is called Home. Home can be set in two main ways: by moving an axis until a Home switch is triggered, or by telling Mach3 that 'right here' is Home. If you have defined Home pins in the Ports&Pins screen, then the first method applies. If you want to load pallets onto the milling table with a robot for instance, you will have to have Home switches.
If you do not have Home switches – that is if they are not defined in Ports&Pins, the second method (an arbitrary definition that 'here' is 'Home') applies. It could be argued that not having Home switches can be more flexible, but it all depends. It can be fine for a hobbyist, but it creates problems in a 24/7 production environment. Having Home switches which are not super-reproducible can actually be a hazard as the Home point can wander around a little.
You can go to the Home position in three different ways. G30 will take you there in a straight line from where ever you are now. If there is something in the way – tough, big crash follows. Be cautious with G30.
G28 will also take you there, but via an intermediate position given in the command. This *may* let you move around an obstacle – or it may not. Read the fine manual and consider carefully. Personally, I favour very explicit move commands in such a case.
You can also go to the Home position with the command `G0 G53 X0 Y0 Z0`. Including the G53 in the command tells Mach3 to work in the Machine Coordinate space rather than the Controlled Point space - for straight line moves (G0, G1). Done blindly, this could be the same as G30 – crash. Fortunately this method is far more flexible as you can put any X value there, not just zero. In particular, the command `G0 G53 X0 Y0` (without any Z movement) can be very useful as we explain below.

Absolute vs Incremental Distance Modes

XYZ
In absolute distance mode, the axis values (X~ Y~ Z~ A~ B~ C~) in a command represent positions in terms of the currently active Coordinate Space. (The ~ sign means 'some value'.) This is the normal mode of operation. You get into this mode via G90.
In incremental distance mode (reached via G91), the axis values (X~ Y~ Z~ A~ B~ C~) in a movement command usually represent increments from the current position, but there are numerous exceptions. For example, the command sequence `G91 / G0 Z10 / G90` (where the slashes represent the New Line character) will move the Controlled Point up by 10 units from where ever it was. This can be very useful for small moves, like retracting the cutter from wherever it was last. That said, the incremental space can be a dangerous space to work in for any length of time.
A reminder: 'incremental' applies only to *moves*, not to other things.

IJK
Some commands allow you to use I, J & K variables in addition to or in place of X, Y & Z. The manual for G91 says that the I & J values are always incremental, regardless of the Distance Mode. The K value is also incremental except for G87 (a special potted boring command). Commands to machine circles and helices can use these IJK variables to some benefit. However, the manual for G90.1 and G91.1 gives a different mode of operation intended for use in G02 and G03 commands. (One can only imagine the intense in-fighting between different vendors supporting different definitions during the committee sessions which NIST used to create the Standard for g-code!) These two commands have been called 'Set IJK Mode' rather

than 'Set Distance Mode'. G90.1 countermands the above, so that IJK values are absolute positions rather than incremental distances. Why one would want to use IJK in this manner is a bit strange, but some people like to machine circles around a centre. Perhaps it allowed some vendors to claim compatibility. G91.1 accords with the above, such that IJK usually represent increments from the current position. As far as I can see, G91.1 is the same as G91 - but I may be wrong as I have never used it. The best explanation I have seen (and about the only one) is that it ‘returns I, J & K to their normal behaviour’.
Note: seeing large, strange and very unexpected circles in place of rounded corners in the Mach tool-path display usually means you may have the wrong IJK mode in the set-up. This happens. Some CAM post-processors are prone to this.

Machine Coordinates vs Controlled Point: Workspace Offsets

So how do these two spaces relate to each other? Basically, thus:

Machine Coord = Controlled Point + Workspace Offset + Other Offset

or

Controlled Point = Machine Coord - Workspace Offset - Other Offset

What is the point here? You might not want your job to have its origin at the Home position: you might want it offset a bit, for any of several good reasons. Any time you position the cutter where *you* want the origin to be and zero the displays, you are doing this.
So Mach3 (following the NIST Standard) has the idea of applying a Workspace Offset to the basic Machine Coord origin. Actually, Mach3 allows up to 254 workspace offsets (or 256 - the manual says different things in different places), with the first 6 being directly accessible by simple g-code commands. To go to Workspace #1, program G54; for Workspace #2 program G55, etc. Workspace #6 is reached by G59; to reach Workspace #7 you program `G59 P7`, and so on. The offsets are all stored in a Workspace Table, which can be saved. (Why not just `G54 Pn`? More committees?)
To set the XYZ offsets for Workspace #n you program `G10 L2 Pn X~ Y~ Z~ A~ B~ C~`; you can omit the axes you do not want to affect and they will not be changed. So the Workspaces are fully programmable.
It is worth noting that when you move the machine so you can zero the DROs where you want the origin to be, you are in fact changing the parameters in the Workspace offset.

'Other Offset'

There is also the 'Other' Offset. This has two components: the second one is dealt with later. For the first component, imagine you have a group of 6 parts to machine, all in a line at equal spacings. Instead of changing the Workspace values or Workspace number (which is indeed quite OK as a method), you can apply a temporary Offset to the base Workspace. This is done using `G52 X~ Y~ Z~`, where the XYZ values are the offset, the distance between the parts. What happens if you program `G52 X~` with the same value a second time? Nothing. You have defined the offset, and that is it. Going into incremental mode does not change this because incremental mode only applies to movements. So if you program `G0 X0 / G52 X10 / G0 X0`, the controlled point (ie the cutter) will actually move along +10 units at the second G0 command (not earlier). You don't have to start at X0 of course: this is just a simple example. If I have a row of parts to machine, all 20 mm apart in a line, then `G52 X20, G52 X40` etc, at the start of a single subroutine, are the way.
There is another way of doing this: use `G92 X~`. This is a highly deprecated command, although it has its uses. What it does is to jam the given X value into the current X Coordinate Space DRO by altering the 'Other Offset'. In effect, G92 simulates typing new values into the DROs. But it can really foul up things if you try to mix G52 and G92 in a program, as they use

the same special Offset variables! Total confusion is probable; some damage is likely.
A caution here: whereas the full sequence for G52 given above will move the X axis +10 units, the same sequence with G92 (`G0 X0 / G92 X10 / G0 X0`) will move the X axis -10 units (at the second G0 command). The X DRO was reading 0, the G92 command put +10 into it, so G0 X0 had to move 'left' to bring the DRO back to 0.
It can be most frustrating trying to set an axis value to something if you don't understand these offsets. Trust me, I know.

Tool Offsets

While Workplace Offsets relate to the coordinate system inside the CNC, Tool Offsets relate to the cutter itself. They were temporarily included in the above under 'Other Offsets' for convenience. However, Tool Offsets are for the length of the cutter, so they only affect the Z axis. (There is an X parameter in the command, but it has little to do with the X axis.) Again, this can be used in at least two different ways.
If you are using tooling mounted in holders such as BT30, then the Tool Offsets can be effectively the distance from the BT30 reference plane to the tool tips. As you change tooling you change which Tool Offset you use. In more practical terms, the Tool Offset is the difference in length between some nominal Tool #0 and other tools. You don’t have to go back to the BT30 reference plane. When using Tool Offsets, the rest of the Workspace stays unchanged. In particular, note that the Tool Offset value affects the Controlled Point; it does NOT affect the Machine Coordinates. (The latter relate after all to some real or virtual Home switches on each axis.)
The Tool Offset value is set by the command `G10 L1 Pn Z(offset)`, where 'n' is the tool number. The Offset is an entry in another table, also savable. You select which physical Tool you want to use with the command M6 ('change the tool') or with Tn M6 if you are using an ATC, but note that all this does is to change the physical tool: it does not change the Tool Offset. It is crucial to understand the difference. To change a Tool Offset you need to program `G43 Hn`, where n is the Tool number. (There is also a G44 command for use when the Tool offsets have the wrong sign - more committee in-fighting? I have not studied this G44 command.)
Can you load Tool 7 and Tool Offset 3? Yes. However, the results might not be exactly what you wanted (or wish to afford).
What this means is that the 'Other Offset' in the above section will also include the Tool Offset value, at least on the Z axis. Yes, that means there are no less than 3 different offsets to be applied to the Machine Coordinate to get the Controlled Point. Actually, the Tool Offsets Table also allows entries for tool wear in both Z and diameter, but Mach3 does not use this data afaik.
You can turn the Tool Offset system off by programming G49. Effectively that selects Tool 0, which has unalterable zero offset values. In many cases this will be perfectly OK, and may make life easier and safer. The Tool Offset table can also hold the Tool Diameter (X~), but that is not touched here.
And that leads to the other method of handling the concept of tool length compensation. You can turn the Tool Offset table off and do all offsetting via the Workspace. If you are sticking with just one or two manually-loaded cutters for the entire job, a direct calibration of a Workspace for each one is quite reasonable, especially for a fully manual operation. Position the tip of the cutter some fixed distance above the top of the part (feeler gauge, ground HSS rod, whatever) and type the thickness of the gauge into the Z DRO. Now Z=0 puts the tool tip just in contact. You can also do this using the Mach Mill Offsets screen.

Test it for Yourself

This is all very well, but you may need to see it for yourself to really understand it all. This is very easy to do. Fire up Mach Mill and go to the Program Run screen. Move the table and spindle to somewhere near the middle. If you have Home switches, you will have to edit the following a bit. I will assume you do not. I will refer to the on-screen DROs as 'user coords' for convenience.

Click on the Ref All Home button. The DROs will change. Click on the Machine Coords button and you should have all zeroes in the DROs. Click again back to get the 'user coords'.

Now zero each of the X, Y & Z axes via the individual buttons. Check that the Machine Coords are still zero.

Go to the Offsets screen. You should have Current Work Offset of 1 (from the G54 in the Init string). Active Work Offset should read G54. The Part Offsets should all be zero. The user coords DROs above the Machine Coords button (on the right) should all be zero. Click on the Machine Coords button and you should still have all zeros.

Now go back to the Program Run screen and move the X & Y axes a bit. Alternately, go to the MDI screen and enter G0 X10 Y-15 (assuming these values are reasonable). The screen DROs should update. Leave the Z axis alone for the moment: you will see why shortly.

Click on the Machine Coords button and the DROs should not change. You have moved the Controlled Point a bit after all, and both user cords and Machine Coords should reflect this.

Go to the Offsets screen and look at the DROs. Current Work Offsets will still be zero, since you have not changed anything there. The user coords on the right will have changed. So far, as expected.

Now zero the user coords on the right. They will go to zero, but the Current Work offsets will now show whatever the user coords were. Why? Because you have not moved the machine (or the Controlled Point) when you 'zeroed the user DROs', and you have not moved the Ref Home position. If you want the user DROs to read zero, then Mach3 will change the Work Offsets to accommodate you.

Type a zero into one of the Current Work Offsets DROs. The value which was there will move back to the user DROs. It has to: you still have not moved the machine.

Tool Offsets are a shade more complex to understand at first. The Tool Offsets area is at the bottom right hand corner. The Tool DRO should read 0 and the Z offset should read 0.000. You cannot change the offset value for Tool 0. Also, if necessary, click on the Zero Z button on the user DROs. This all says that the Controlled Point for Tool 0 is at Z=0. We will call that the surface of the part.

Now type 1 into the Tool DRO, type 5 into the DRO under Gage Block Height and click on Set Tool Offset. You should get -5 in the Z offset DRO and +5 in the Z DRO. This says that if you have a Tool 1 which is 5 units shorter than Tool 0 when mounted in the spindle but you have NOT moved the Z axis, then the tip of Tool 1 will be +5 units above whatever was the surface of the part. A 5 unit high gage block will just fit under Tool 1. While you are at it, check the current Machine Coords: Z will still read 0 because you still have not moved the machine.

To bring the tip of Tool 1 down to the surface of the part you will need to apply a Z Offset of -5 units. This is now obvious from looking at the picture of the gage block. Change Tool number to 0 and the user Z DRO will go back to 0.

You can also test G52 here. Let's assume Work Offset 1 has a Y value of 0 and user Y DRO has a value of 0 (so Machine Coord Y=0). Go to the MDI screen and type G52 Y5. The user Y DROs will now read -5. You have told Mach3 to move the user origin for the Y axis 5 units north. The Machine Coords will still show Y=0, because you have not moved the machine. But if the USER origin is now 5 units to the north, while the machine has not moved, then the current user Y coord must be -5 units.

You cannot see the G52 value on the Offsets screen, but you can see it on the Diagnostics screen under the G92 label. Curiously, while you can change other DROs on the Diagnostics screen, it seems you cannot change the G52/G92 DROs there. This may be a quirk of Mach3.

Parameters

Mach3 allows about 10,000 user parameters (like #10, #123, #1000 etc). However, there are also some special cases outside the 10,000 range which seem to be poorly documented. Obviously, these are Mach3 special bypasses, and may not be in the NIST spec. If you know of any more, please, let me know!
Parameters #500 - #600: these seem to be saved in the XML file and restored the next time you fire up Mach3. They are 'persistent'.
Parameters #5211 - #5216: these give you direct access to the offsets for the G52/G92 commands. Touching these without really, really knowing what you are doing could be exciting and expensive. Better by far to use G52 explicitly – although being able to read the current offsets may have some value.
Parameters #15001 - #15255: these are linked (to a certain extent) with User DROs 1001 – 1255. You would normally only need to use these if you are doing some fancy screen creation.
Parameters #99nnn: these are also special, at least for some values of nnn. Noting that the addresses for the X, Y, Z DROs etc for macro commands are 800, 801, 802 etc (ie for the get/setOEMDRO commands), it turns out that you can also access the on-screen DROs from g-code by using the parameters #9980n. I have not explored other values for nnn. So `#99802=0` is the direct equivalent of G92 Z0.

Practical Defaults

Not knowing which Workspace offset you are in can make life difficult, but do not despair. The default initialisation string for Mach3 includes a G54 command. This will select Workspace #1. Many people will never leave that workspace – and will never *need* to leave that Workspace. That's fine. Not knowing what Tool offset you have set up can be equally exciting, so the default initialisation string for Mach3 also includes a G49 command to turn the whole Tool Offset system off. You can in fact do everything you want to do in the G49/G54 space alone.