

User Concepts

Contents

1	Trajectory Control	1
1.1	Trajectory Planning	1
1.2	Path Following	1
1.3	Programming the Planner	1
1.4	Planning Moves	2
2	G Code	3
2.1	Defaults	3
2.2	Feed Rate	3
2.3	Tool Radius Offset	3
3	Homing	3
4	Tool Changes	3
5	Coordinate Systems	3
5.1	G53 Machine Coordinate	3
5.2	G54-59.3 User Coordinates	4
5.3	When Your Lost	4

This chapter covers important user concepts that should be understood before attempting to run a CNC machine with g code.

1 Trajectory Control

1.1 Trajectory Planning

Trajectory planning, in general, is the means by which EMC follows the path specified by your G Code program, while still operating within the limits of your machinery.

A G Code program can never be fully obeyed. For example imagine you specify as a single-line program the following move:

```
G1 X1 F10 (G1 is linear move, X1 is the destination, F10 is the speed)
```

In reality, the whole move can't be made at F10, since the machine must accelerate from a stop, move toward X=1, and then decelerate to stop again. Sometimes part of the move is done at F10, but for many moves, especially short ones, the specified feed rate is never reached at all. Having short moves in your G Code can cause your machine to slow down and speed up for the longer moves if the naive cam detector is not employed with G64 Pn.

The basic acceleration and deceleration described above is not complex and there is no compromise to be made. In the INI file the specified machine constraints such as maximum axis velocity and axis acceleration must be obeyed by the trajectory planner.

1.2 Path Following

A less straightforward problem is that of path following. When you program a corner in G Code, the trajectory planner can do several things, all of which are right in some cases: it can decelerate to a stop exactly at the coordinates of the corner, and then accelerate in the new direction. It can also do what is called blending, which is to keep the feed rate up while going through the corner, making it necessary to round the corner off in order to obey machine constraints. You can see that there is a trade off here: you can slow down to get better path following, or keep the speed up and have worse path following. Depending on the particular cut, the material, the tooling, etc., the programmer may want to compromise differently.

Rapid moves also obey the current trajectory control. With moves long enough to reach maximum velocity on a machine with low acceleration and no path tolerance specified, you can get a fairly round corner.

1.3 Programming the Planner

The trajectory control commands are as follows:

G61

(Exact Path Mode) visits the programmed point exactly, even though that means it might temporarily come to a complete stop in order to change direction to the next programmed point.

G61.1

(Exact Stop Mode) tells the planner to come to an exact stop at every segment's end.

G64

(Blend Without Tolerance Mode) G64 is the default setting when you start EMC. G64 is just blending and the naive cam detector is not enabled. G64 and G64P0 tell the planner to sacrifice path following accuracy in order to keep the feed rate up. This is necessary for some types of material or tooling where exact stops are harmful, and can work great as long as the programmer is careful to keep in mind that the tool's path will be somewhat more curvy than the program specifies. When using G0 (rapid) moves with G64 use caution on clearance moves and allow enough distance to clear obstacles based on the acceleration capabilities of your machine.

G64 P- Q-

(Blend With Tolerance Mode) This enables the naive cam detector and enables blending with a tolerance. If you program G64 P0.05, you tell the planner that you want continuous feed, but at programmed corners you want it to slow down enough so that the tool path can stay within 0.05 user units of the programmed path. The exact amount of slowdown depends on the geometry of the programmed corner and the machine constraints, but the only thing the programmer needs to worry about is the tolerance. This gives the programmer complete control over the path following compromise. The blend tolerance can be changed throughout the program as necessary. Beware that a specification of G64 P0 has the same effect as G64 alone (above), which is necessary for backward compatibility for old G Code programs. See the G Code Chapter for more information on G64 P- Q-.

Blending without tolerance

The controlled point will touch each specified movement at at least one point. The machine will never move at such a speed that it cannot come to an exact stop at the end of the current movement (or next movement, if you pause when blending has already started). The distance from the end point of the move is as large as it needs to be to keep up the best contouring feed.

Naive Cam Detector

Successive G1 moves that involve only the XYZ axes that deviate less than Q- from a straight line are merged into a single straight line. This merged movement replaces the individual G1 movements for the purposes of blending with tolerance. Between successive movements, the controlled point will pass no more than P- from the actual endpoints of the movements. The controlled point will touch at least one point on each movement. The machine will never move at such a speed that it cannot come to an exact stop at the end of the current movement (or next movement, if you pause when blending has already started) On G2/3 moves in the G17 (XY) plane when the maximum deviation of an arc from a straight line is less than the G64 Q- tolerance the arc is broken into two lines (from start of arc to midpoint, and from midpoint to end). those lines are then subject to the naive cam algorithm for lines. Thus, line-arc, arc-arc, and arc-line cases as well as line-line benefit from the naive cam detector. This improves contouring performance by simplifying the path.

In the following figure the blue line represents the actual machine velocity. The red lines are the acceleration capability of the machine. The horizontal lines below each plot is the planned move. The upper plot shows how the trajectory planner will slow the machine down when short moves are encountered to stay within the limits of the machines acceleration setting to be able to come to an exact stop at the end of the next move. The bottom plot shows the effect of the Naive Cam Detector to combine the moves and do a better job of keeping the velocity as planned.

common/images/naive-cam.*

Figure 1: Naive Cam Detector

1.4 Planning Moves

Make sure moves are long enough to suit your machine/material. Principally because of the rule that the machine will never move at such a speed that it cannot come to a complete stop at the end of the current movement, there is a minimum movement length that will allow the machine to keep up a requested feed rate with a given acceleration setting.

The acceleration and deceleration phase each use half the ini file MAX_ACCELERATION. In a blend that is an exact reversal, this causes the total axis acceleration to equal the ini file MAX_ACCELERATION. In other cases, the actual machine acceleration is somewhat less than the ini file acceleration

To keep up feed rate, the move must be longer than the distance it takes to accelerate from 0 to the desired feed rate and then stop again. Using A as 1/2 the ini file MAX_ACCELERATION and F as the feed rate {*} in units per second {*}, the acceleration time is $t_a = F/A$ and the acceleration distance is $da = (1/2) * F * t_a$ the deceleration time and distance are the same, making the critical distance $d = da + dd = 2 * da = F^2 / A$.

For example, for a feed rate of 1 inch per second and an acceleration of 10 inch/sec², the critical distance is $12 / 10 = .1$ inch. For a feed rate of .5 inch per second, the critical distance is $.52 / 10 = .025$ inch.

2 G Code

2.1 Defaults

When EMC first starts up many G and M codes are loaded by default. The current active G and M codes can be viewed on the MDI tab in the Active G-Codes:window in the AXIS interface. These G and M codes define the behavior of EMC and it is important that you understand what each one does before running EMC. The defaults can be changed when running a G-Code file and left in a different state than when you started your EMC session. The best practice is to set the defaults needed for the job in the preamble of your G-Code file and not assume that the defaults have not changed. Printing out the G-Code Quick Reference ([?]) page can help you remember what each one is.

2.2 Feed Rate

How the feed rate is applied depends on if an axis involved with the move is a rotary axis. Read and understand the Feed Rate section ([?]) if you have a rotary axis or a lathe.

2.3 Tool Radius Offset

Tool Radius Offset (G41/42) requires that the tool be able to touch somewhere along each programmed move without gouging the two adjacent moves. If that is not possible with the current tool diameter you will get an error. A smaller diameter tool may run without an error on the same path. This means you can program a cutter to pass down a path that is narrower than the cutter without any errors. See the Tool Compensation Section ([?]) for more information.

3 Homing

After starting EMC2 each axis must be homed prior to running a program or running a MDI command.

If you want to deviate from the default behavior, or want to use the Mini interface you will need to set the option NO_FORCE_HOMING = 1 in the [TRAJ] section of your ini file. More information on homing can be found in the Integrators Manual.

4 Tool Changes

There are several options when doing manual tool changes. See the [EMCIO] section of the Integrators Manual for information on configuration of these options. Also see the G28 and G30 section of the User Manual.

5 Coordinate Systems

The Coordinate Systems can be confusing at first. Before running a CNC machine you must understand the basics of the coordinate systems used by EMC. In depth information on the EMC Coordinate Systems is in the coordinate section [?] of this manual.

5.1 G53 Machine Coordinate

When you home EMC you set the G53 Machine Coordinate System to 0 for each axis homed.

- No other coordinate systems or tool offsets are changed by homing.

The only time you move in the G53 machine coordinate system is when you program a G53 on the same line as a move. Normally you are in the G54 coordinate system.

5.2 G54-59.3 User Coordinates

Normally you use the G54 Coordinate System. When an offset is applied to a current user coordinate system a small blue ball with lines will be at the machine origin when your DRO is displaying Position: Relative Actual in Axis. If your offsets are temporary use the Zero Coordinate System from the Machine menu or program G10 L2 P1 X0 Y0 Z0 at the end of your G Code file. Change the Pnumber to suit the coordinate system you wish to clear the offset in.

- Offsets stored in a user coordinate system are retained when EMC is shut down.
- Using the Touch Off button in Axis sets an offset for the chosen User Coordinate System.

5.3 When Your Lost

If you're having trouble getting 0,0,0 on the DRO when you think you should. You have some offsets programmed in and need to remove them.

- Move to the Machine origin with G53 G0 X0 Y0 Z0
- Clear any G92 offset with G92.1
- Use the G54 coordinate system with G54
- Set the G54 coordinate system to be the same as the machine coordinate system with G10 L2 P1 X0 Y0 Z0
- Turn off tool offsets with G49
- Turn on the Relative Coordinate Display from the menu

Now you should be at the machine origin X0 Y0 Z0 and the relative coordinate system should be the same as the machine coordinate system.
