G-Code Manuscript

Collin Safford

May 2019

Written in partial fulfillment of an independent study project with the aim of improving the 3D printing capability at Northern Vermont University-Johnson.

Table of Contents

Starting Code

M-codes tell the machine to do a task while G-codes deal more with the movement of the head. Many of the examples start out with something similar to:

```
M107
M83  ; extruder relative mode
M104 S210 ; set extruder temp
M140 S55 ; set bed temp
M190 S55 ; wait for bed temp
M109 S210 ; wait for extruder temp
G28 W ; home all without mesh bed level
G80 ; mesh bed leveling
G1 Y-3.0 F1000.0 ; go outside print area
G1 X60.0 E9.0  F1000.0 ; intro line
G1 X100.0 E12.5  F1000.0 ; intro line
G21 ; set units to millimeters
G90 ; use absolute coordinates
M83 ; use relative distances for extrusion
```

Explanation

It is usually a good idea to start with M107 to turn off the fan in case it was on from before. M106 is used to turn on the fan, but this is usually done later in the first portion of the code when it is needed. M83 sets the extruder to relative mode so that the amount needed is typed instead of the total amount used this far. The amount needed to be extruded depends on the material, the settings of the printer, and where in the project it is. It has been seen to be between 1/29.0 and 1/41.4 of the distance traveled. For example, G1 X108.467 Y93.351 E0.03799 is telling the machine to use 0.03799mm of material.

M104 and M109 set and wait for the temperature of the extruder while M140 and M190 set and wait for the bed temperature.

G28 moves the head to the home position. G80 helps compensate for the bed being tilted and not level.

G1 is the main command used with 3D printing as it tells the head to move in the X, Y, and Z plane as well as E which tells how much to extrude between the starting point and ending point, F which is the feed rate per minute, and H (or S) which checks to see if an end-stop was hit.

G21 sets the units to millimeters while G20 sets the units to inches. G90 sets the head to use absolute coordinates (a numbered grid on the bed) and G91 sets it to relative positioning (how much to move in a direction).

The example from before: G1 X108.467 Y93.351 E0.03799 is telling the head to move from its current point to the grid point (108.467, 93.351) with the Z axis remaining the same.

There are a few more M-codes commonly used in programs, as can be seen below, and a few more that are uncommon, not used by the Prusa examples, or not for this model of 3D printer.

There are also more G-codes than were used in the examples, but many listed might not work with this model as many are created by users for their machines and might be able to be programed in later.

M-codes and G-codes

List of M-codes

All M-codes found in Prusa-made examples and useful other codes and description:

M107: fan off
M104: set extruder temp
M140: set bed temp
M190: wait for bed temp
M109: wait for extruder temp
M106: fan on
M84: motors
M115: tell printer latest firmware version
M83: use relative distances for extrusion
M82: use absolute distances for extrusion
M204: set default acceleration
M201: set max acceleration

List of G-codes

All G-codes codes found in Prusa-made examples and useful other codes and description:

G1: linear movement
G20: set units to inches
G21: set units to millimeters
G28: move to home position
G80: mesh bed leveling
G90: use absolute distances for movement
G91: use relative distances for movement

Links for other possible G-codes and M-codes

(lists all RepRap G-code and M-code created; might not work with Prusa):
https://reprap.org/wiki/G-code

(list of G-code and M-code used in large program; many use as guide):
https://github.com/prusa3d/Prusa-Firmware/blob/MK2/Firmware/Marlin_main.cpp

(list of G-code and M-code supported by Prusa; differs from program that runs on Prusa):
https://github.com/prusa3d/Prusa-Firmware/wiki/Supported-G-codes

(lists most commonly used G-code and M-code; not Prusa specific):
https://all3dp.com/g-code-tutorial-3d-printer-gcode-commands/

(overview of g-code from Linux; basics should be the same with all G-code)
http://linuxcnc.org/docs/2.6/html/gcode/overview.html

(list of g-codes from Linux; program from above linked to this):
http://linuxcnc.org/docs/2.6/html/gcode/gcode.html

(list of m-codes from Linux; program from above linked to this):
http://linuxcnc.org/docs/2.6/html/gcode/m-code.html

G-code without other programs

Programming with G-code without any help from other programs would take a very long time as there are tens of thousands of lines of code that all depend on the previous line and have their own rules depending on how far in you are. The code tells the printer to print the bottom layer first and continue printing on top of that layer and each following layer as it goes. Occasionally some support struts need to be coded in that break off easily so that the entire structure does not collapse before it is done being printed. Due to this layering, the lines for a specific portion are often hundreds or thousands of lines away from each other. Without any other programs to help, it would be nearly impossible to cut out a selected portion in order to color it differently from the rest or to print only that portion. Looking at the G-code is almost never needed unless there was a problem with the initial few lines telling the printer how to start the project or there was a problem with the last few lines telling the printer how to end the project. Everything else can be done with other programs making it unnecessary to know what the G-code is saying. Most ways to get the G-code are not user friendly and have lines of code separated by a space instead of each line of code getting its own line. The most the average user would need to know is that G-code is what tells the printer what to print. A curious user would need to know the basic G-codes and M-codes to figure out what the code was saying.

Additional Programs

PrusaSlicer

PrusaSlicer (formerly known as Slic3r Prusa Edition or Slic3r PE) is the slicer software specifically for the Prusa printers. This allows someone to see what objects are being created without having to print them. It can convert STL files to G-code as well. By breaking up an object into multiple STL files, PrusaSlicer allows the user to print each section as a different color while keeping the object whole. PrusaSlicer can also slice objects up on the y-axis allowing taller objects to be printed in parts letting them fit on the printer.

Meshmixer

Meshmixer allows the user to cut portions off an object much easier than with PrusaSlicer and it also allows the user to create new objects such as pivot points that go into both cut off portions allowing them to be connected without the need of glue after they are printed.

Tinkercad

Tinkercad is an online software that allows the user to design an object using premade shapes. Other users on Tinkercad have also shared their creations allowing anyone to download them and print them out, look at how it was created, or change portions of it to make it their own. All objects made can be exported into files that PrusaSlicer recognizes allowing them to be turned into G-code or edited more.

Additional Resources

Prusa has a YouTube channel that has helpful videos that show how to use the printer better as well as how to use some software like PrusaSlicer and Meshmixer. The channel name is “Prusa 3D by Josef Prusa” and can be found here https://www.youtube.com/user/prusajr/featured.