

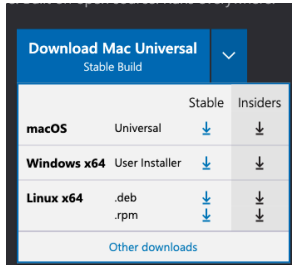
MUTANT

Marlin Tool Changing Feature Set Up Guide

Quick Start guide for Marlin 2.0.X

For this tutorial, we will use Visual studio and the PlatformIO extension within visual studio for editing the Marlin firmware code.

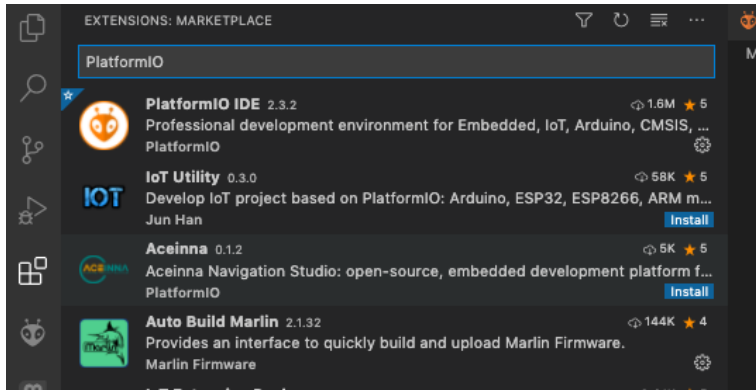
First, you need to download **Visual Studio**. Go to <https://code.visualstudio.com/> and click on the download button and install the application after it has been downloaded. (see example image)



Once you have it installed, open up **Visual Studio** and click on the extensions button. (see example image)



In the search bar, type in PlatformIO and **install PlatformIO IDE**. (see example image)



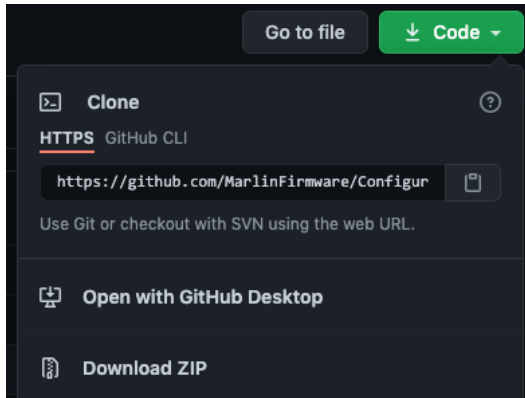
While that is installing, go to <https://marlinfw.org/meta/download/> and download a fresh copy of the latest release of **Marlin** and also example **Configurations** for that version. (see example image)

Description	Version	Download	Configurations
Latest release <i>Supports AVR and ARM Arduino and PlatformIO</i>	2.0.7.2	2.0.x.zip	View / Download

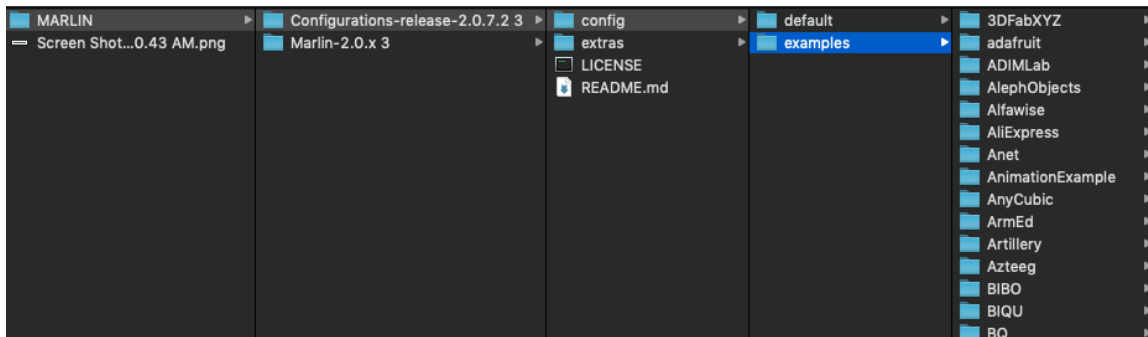
On the configurations page click on the green “code” button then click on download zip to download the example configs. (see example image)

MUTANT

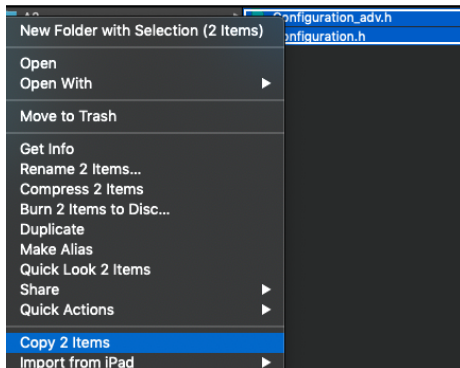
Marlin Tool Changing Feature Set Up Guide



Extract the files after they have been downloaded and keep note of the location where you have extracted them. Once you have extracted the files, you can open up the example **Configurations-release-2.0.X Folder** you just downloaded and extracted. (see example image)



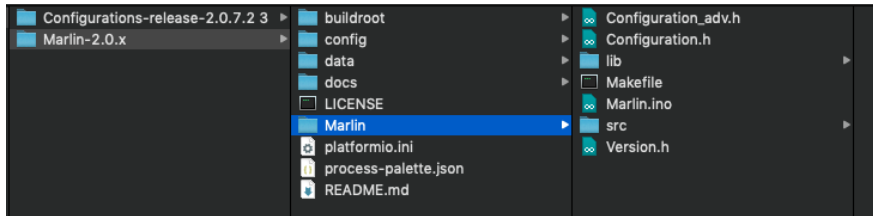
Within the **Configurations-release-2.0.X** folder open **config**, then open **examples** and find your printer model. Within the folder for your printer you will find files called **configuration.h** and **configuration_adv.h**. Copy these files by highlighting both of them and right-clicking on them and selecting copy.(see example image)



Then navigate to the **Marlin-2.0.x** and open the **Marlin** folder within that. (see example image)

MUTANT

Marlin Tool Changing Feature Set Up Guide

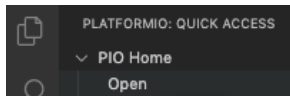


Replace the files already within that folder with the ones we copied earlier by right clicking inside the folder and clicking paste.

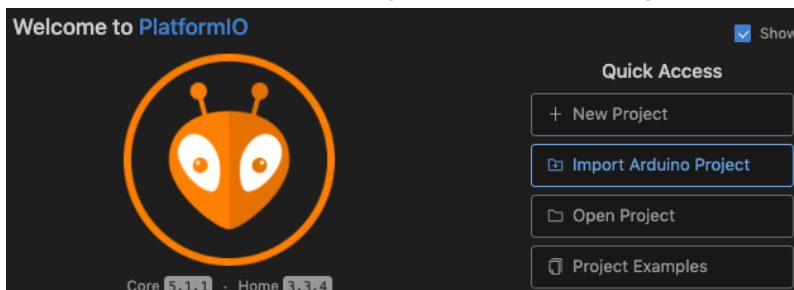
Next, open **Visual Studio** and click on the **Alien symbol** to the left. That will bring up **PlatformIO**. (see example image)



On the left pane, under Quick Access > **PIO Home**, click **Open**. (see example image)



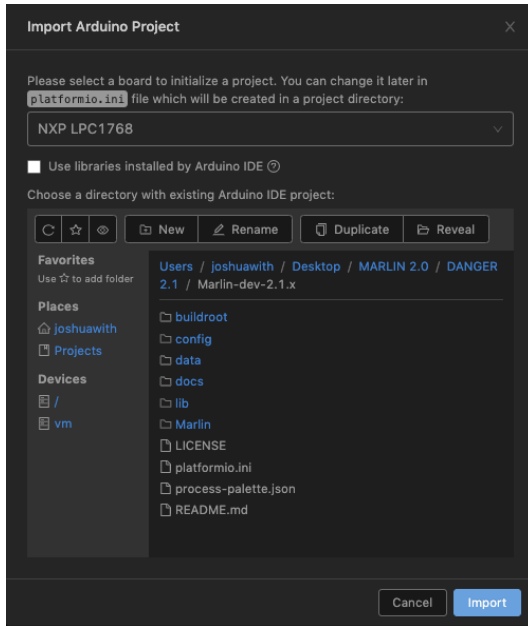
From here click **Import Arduino project**. (see example image)



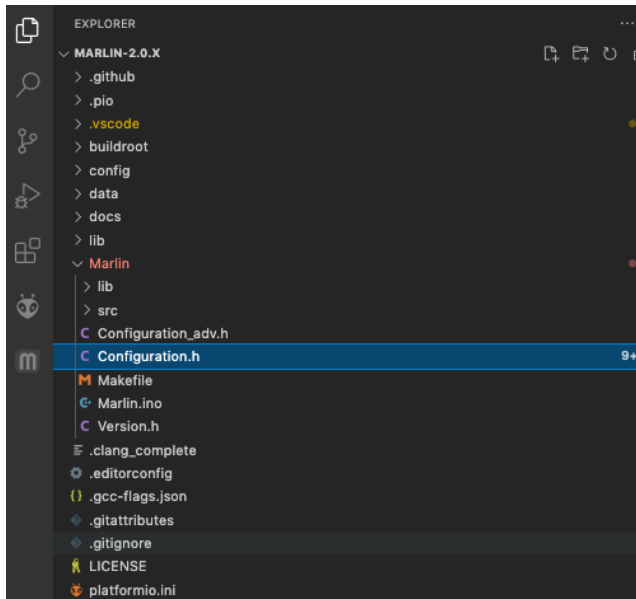
Select the board that you will be using (Refer to the **BOARD** section below for help with identification) and then select the location where you extracted the Marlin Firmware and select the underlying Marlin Folder and click Import. (see example image)

MUTANT

Marlin Tool Changing Feature Set Up Guide



Now that the project has been loaded, open up the **configuration.h** file within the **Marlin Folder**. (see example image)



Next we are going to build it by clicking the **Blue Check Mark** on the bottom of the window to make sure that there are no errors before you start configuring. (see example image)



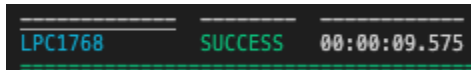
MUTANT

Marlin Tool Changing Feature Set Up Guide

If you don't get any errors, plug your board into the computer and then you can upload the firmware to the board by **clicking on the Right Arrow** next to the checkmark. (see example image)



If you see success at the bottom, your firmware has been uploaded and you should be set to test the board. (see example image)



Make sure to test the functions of the printer to see that everything is working as it should (ex:motors going in the correct direction). You can use a program like **PronterFace** to connect to the board to send commands like M503 to have the board send back machine info, or M119 to send back endstop status. You can get it from <https://www.pronterface.com/>. This software also lets you move each axis in small increments which also helps with calibration.

If everything works as it should, move on to adding the select tool menu item.

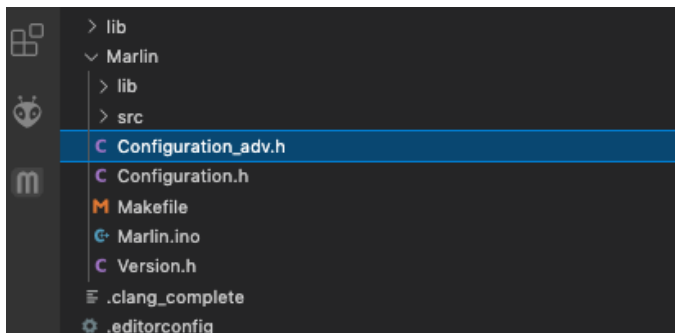
If something is not working as it should, edit the code (we will cover that later in this tutorial)

Adding the Select Tool Menu Item for the MUTANT

Here we will go over how to set up a menu item in Marlin to allow you to select different tool heads in the menu on the LCD, with each one having its own offsets, thermistor values, extruder calibration, and more.

These instructions assume that you have saved, compiled, uploaded, you have tested your firmware and have a fully working printer, and are continuing on from the instructions above with your project still open.

First you're going to want to open **Configuration_adv.h** (see example image)



Then use **ctrl+f** to open the find function and search for the phrase "**CUSTOM_USER_MENU**". (see example image)

MUTANT

Marlin Tool Changing Feature Set Up Guide

```
Marlin > C Configuration_adv.h > ...
2842 #define GCODE_MACROS_SLOT_SIZE 50
2843 #endif
2844
2845 /**
2846  * User-defined menu items that execute custom GCode
2847  */
2848 //#define CUSTOM_USER_MENUS
2849 #if ENABLED(CUSTOM_USER_MENUS)
2850   //#define CUSTOM_USER_MENU_TITLE "Custom Commands"
2851   #define USER_SCRIPT_DONE "M117 User Script Done"
2852   #define USER_SCRIPT_AUDIBLE_FEEDBACK
2853   //#define USER_SCRIPT_RETURN // Return to status screen after a script
2854
2855   #define USER_DESC_1 "Home & UBL Info"
2856   #define USER_GCODE_1 "G28\nG29 W"
2857
2858   #define USER_DESC_2 "Preheat for " PREHEAT_1_LABEL
2859   #define USER_GCODE_2 "M140 S" STRINGIFY(PREHEAT_1_TEMP_BED) "\nM104 S" STRINGIFY(PREHEAT_1_TEMP_
2860
2861   #define USER_DESC_3 "Preheat for " PREHEAT_2_LABEL
2862   #define USER_GCODE_3 "M140 S" STRINGIFY(PREHEAT_2_TEMP_BED) "\nM104 S" STRINGIFY(PREHEAT_2_TEMP_
2863
2864   #define USER_DESC_4 "Heat Bed/Home/Level"
2865   #define USER_GCODE_4 "M140 S" STRINGIFY(PREHEAT_2_TEMP_BED) "\nG28\nG29"
2866
2867   #define USER_DESC_5 "Home & Info"
2868   #define USER_GCODE_5 "G28\nM503"
2869 #endif
```

Uncomment by removing the two back slashes // in front of **#define** on any given code. That will change the color of the code giving you a visual that the uncommented code has been activated. (see example images)

```
38 */
39 //#define CONFIGURATION_H_VERSION 020005
40
41 //=====
42 //===== Getting Started =====
43 //=====
44
```

```
38 */
39 #define CONFIGURATION_H_VERSION 020005
40
41 //=====
42 //===== Getting Started =====
43 //=====
44
```

Uncomment **#define CUSTOM_USER_MENUS** (see example image)

```
2848 #define CUSTOM_USER_MENUS
```

Then uncomment **#define CUSTOM_USER_MENU_TITLE** and replace “Custom Commands” with “Select Tool”. (see example image)

```
2850 #define CUSTOM_USER_MENU_TITLE "Select Tool"
```

Then, after **#define user_script_done** you want to change the text in the parentheses to "M117 Tool Selected" (see example image)

```
2851 #define USER_SCRIPT_DONE "M117 Tool Selected"
```

MUTANT

Marlin Tool Changing Feature Set Up Guide

Then uncomment `#define USER_SCRIPT_RETURN` so that it will return to the info screen after you select a tool (after executing the script). (see example image)

```
2853 #define USER_SCRIPT_RETURN // Return to status screen after a script
```

The menu items are just below.

Make sure you keep them in the same format, and only change what is inside the parentheses.

Copy and paste this in the place of the first option.

```
#define USER_DESC_1 "3D print"
```

```
#define USER_GCODE_1 "M206 X0Y0Z0\nM92 E93\nM301 P21.3I1.5D76.55\nM204 P500 R500 T500\nM900 K0.0\nM305 R4700 T100000 B4092\nM200 D1.75\nM500; 3D printer"
```

(see example image)

```
2848 #define CUSTOM_USER_MENUS
2849 #if ENABLED(CUSTOM_USER_MENUS)
2850 #define CUSTOM_USER_MENU_TITLE "Select Tool"
2851 #define USER_SCRIPT_DONE "M117 Tool Selected"
2852 #define USER_SCRIPT_AUDIBLE_FEEDBACK
2853 #define USER_SCRIPT_RETURN // Return to status screen after a script
2854
2855 #define USER_DESC_1 "3D print"
2856 #define USER_GCODE_1 "M206 X0Y0Z0\nM92 E93\nM301 P21.3I1.5D76.55\nM204 P500 R500 T500\nM900 K0.0\nM305 R4700 T100000 B4092\nM200 D1.75\nM500; 3D printer"
2857
2858
2859 #define USER_DESC_2 "Preheat for " PREHEAT_1_LABEL
2860 #define USER_GCODE_2 "M140 S" STRINGIFY(PREHEAT_1_TEMP_BED) "\nM104 S" STRINGIFY(PREHEAT_1_TEMP_HOTEND)
2861
2862 #define USER_DESC_3 "Preheat for " PREHEAT_2_LABEL
2863 #define USER_GCODE_3 "M140 S" STRINGIFY(PREHEAT_2_TEMP_BED) "\nM104 S" STRINGIFY(PREHEAT_2_TEMP_HOTEND)
```

In the parentheses after `USER_DESC_X` there will be the name of the toolhead and this will be the name of the selection that will show in the menu under Select Tool on the LCD.

In the parentheses after `USER_GCODE_X` you will paste the gcode from above that you need to modify for each toolhead and this executes upon selecting the toolhead in the menu on the LCD.

Modify these numbers to fit your needs.

M206 - Toolhead offset relative To zero

M92 - E step calibration

M301 - PID calibration

M204 - Acceleration values

M900 - Linear advance settings

M305 - Thermistor parameters

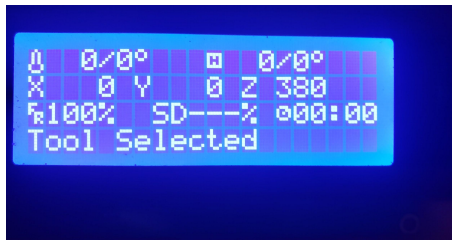
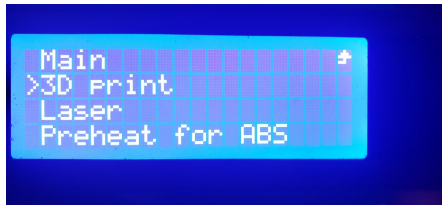
M200 - filament diameter

After that, you upload it to your board, and test it out to make sure all is working properly. (see example images)



MUTANT

Marlin Tool Changing Feature Set Up Guide



General Walkthrough of Marlin 2.0 firmware Configuration.

As indicated earlier, if the example configuration was not behaving properly we will cover the settings and solutions here.

If you are going through these instructions here as a troubleshooting step to make an example configuration work properly, every time you change a setting within the code, you are going to want to save it, compile it, and then upload it to your board and test to see if you have fixed your issue. Make sure that you are only working with one issue at a time to make things easier on yourself for troubleshooting. This also allows you to be able to revert settings easily if something didn't work the way you expect it to.

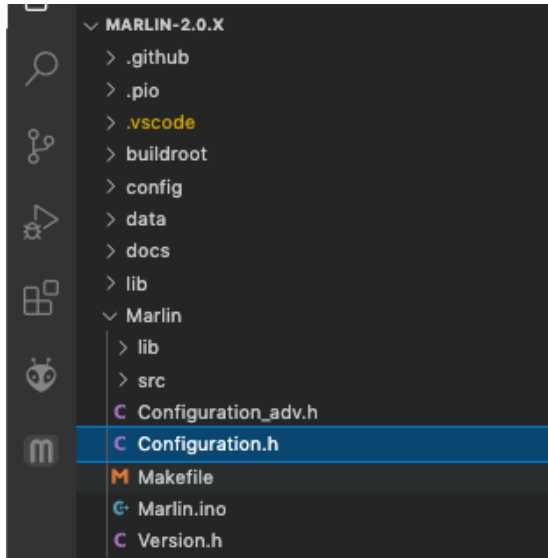
If you are configuring Marlin from scratch without one of the example configurations, just read through the whole thing, then go ahead and compile when you're done changing the settings that you need for your printer. While compiling, if it fails, it will spit out an error code telling you what needs to be fixed in order to be able to compile.

Let's move forward.

Open up your **Configuration.h** (see example image)

MUTANT

Marlin Tool Changing Feature Set Up Guide



We are going to go down the list and explain the most important functions.

There is a lot that does not need to be messed with in most cases, so we will skip over some stuff.

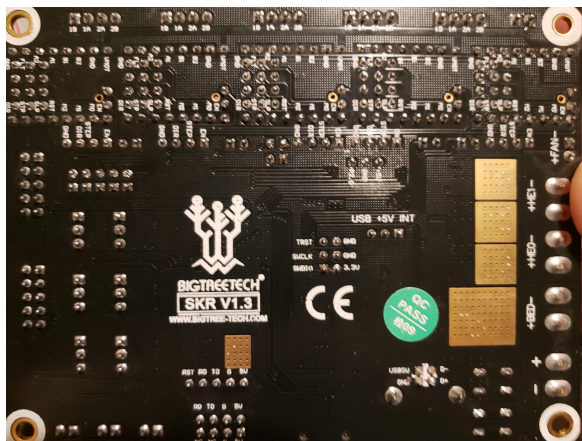
It's a lot easier than it looks at first glance. You are going to want to just read through it all as a lot is explained through comments throughout the code.

Before we start configuring the code, we need to make sure we are able to connect and upload to the board. If you have used an example configuration and have already uploaded it to your printer, then you can skip down to the **CONFIG section**.

If you could not get the firmware uploaded to your board, then continue on with the **BOARD section**.

BOARD

First things first, we need to make sure that the correct board is chosen. You are going to want to look up your printer and also look at the information screen printed onto the board itself to help identify which board you are using. We can see that the example board is a BIG TREE TECH SKR v1.3 (see example image)

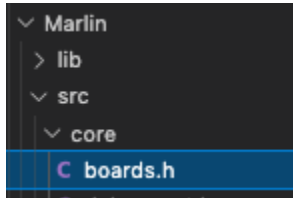


MUTANT

Marlin Tool Changing Feature Set Up Guide

Once you know what board you have, it is also a good idea to do a quick google search on the board and identify which processor is used on that board.

On the left panel open your Marlin folder > src > core > and then open **boards.h** (see example image)



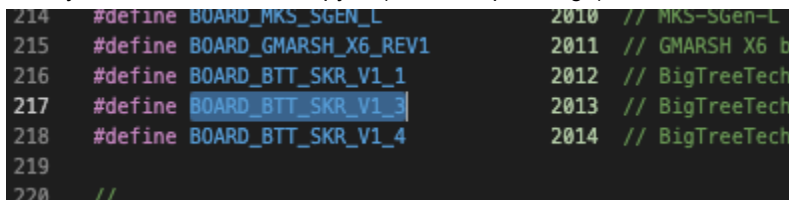
Within this file, find the names to all of the boards that are capable of being used with the Marlin firmware.

After research you may find that your board is wired exactly the same as another board, like one of the ramps boards for instance.

If you press Ctrl+f, a search bar will come up and you can type in part of the name of your board to search for it. (see example image)

```
94 #define BOARD_BRAINWAVE 1700 // Brainwave Pro (AT90USB1286)
95 #define BOARD_BRAINWAVE_PRO 1701 // Brainwave Pro (AT90USB1286)
96 #define BOARD_SAV_MKI 1705 // SAV Mk-I (AT90USB1286)
97 #define BOARD_TEENSY2 1706 // Teensy++2.0 (AT90USB1286)
98 #define BOARD_5DPRINT 1707 // 5DPrint D8 Driver Board
99
100 //
101 // LPC1768 ARM Cortex M3
102 //
103
104 #define BOARD_RAMPS_14_RE_ARM_EFB 2000 // Re-ARM with RAMPS 1.4 (Power outputs: Hotend, Fan, Bed)
105 #define BOARD_RAMPS_14_RE_ARM_EEB 2001 // Re-ARM with RAMPS 1.4 (Power outputs: Hotend0, Hotend1, Bed)
106 #define BOARD_RAMPS_14_RE_ARM_EFF 2002 // Re-ARM with RAMPS 1.4 (Power outputs: Hotend, Fan, Bed)
107 #define BOARD_RAMPS_14_RE_ARM_EEF 2003 // Re-ARM with RAMPS 1.4 (Power outputs: Hotend0, Hotend1, Bed)
108 #define BOARD_RAMPS_14_RE_ARM_SF 2004 // Re-ARM with RAMPS 1.4 (Power outputs: Spindle, Controller)
109 #define BOARD_MKS_SBASE 2005 // MKS-Sbase (Power outputs: Hotend0, Hotend1, Bed, Fan)
110 #define BOARD_AZSMZ_MINI 2006 // AZSMZ Mini
111 #define BOARD_BIQU_BQ111_A4 2007 // BIQU BQ111-A4 (Power outputs: Hotend, Fan, Bed)
112 #define BOARD_SELENA_COMPACT 2008 // Selena Compact (Power outputs: Hotend0, Hotend1, Bed)
113 #define BOARD_BIQU_B300_V1_0 2009 // BIQU B300_V1.0 (Power outputs: Hotend0, Fan, Bed)
114 #define BOARD_MKS_SGEN_L 2010 // MKS-SGen-L (Power outputs: Hotend0, Hotend1, Bed, Fan)
115 #define BOARD_GMARSH_X6_REV1 2011 // GMARSH X6 board, revision 1 prototype
116 #define BOARD_BTT_SKR_V1_1 2012 // BigTreeTech SKR v1.1 (Power outputs: Hotend0, Hotend1, Bed)
117 #define BOARD_BTT_SKR_V1_3 2013 // BigTreeTech SKR v1.3 (Power outputs: Hotend0, Hotend1, Bed)
118 #define BOARD_BTT_SKR_V1_4 2014 // BigTreeTech SKR v1.4 (Power outputs: Hotend0, Hotend1, Bed)
119
```

Once you find it, select and copy it. (see example image)



Then go back to your **Configuration.h** file.

Scroll down to where it says **#define MOTHERBOARD** and **paste the name of your board** over the one already there. (see example images)

MUTANT

Marlin Tool Changing Feature Set Up Guide

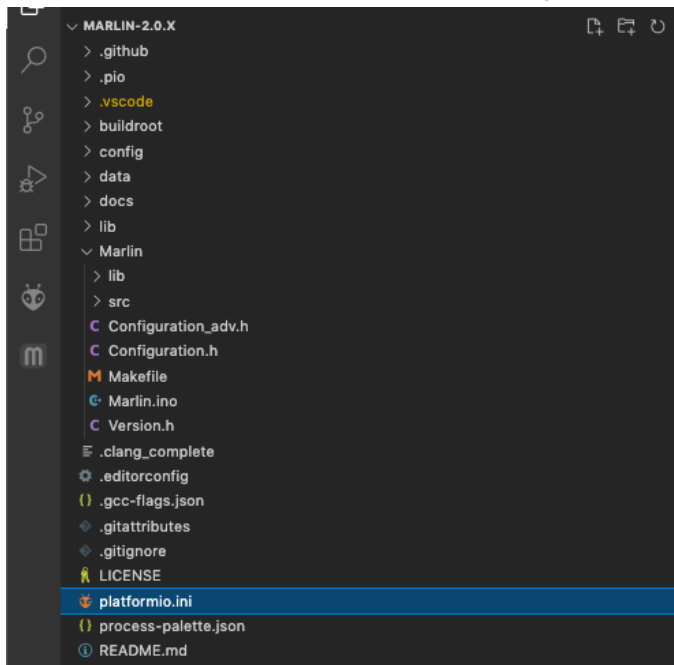
```
129 // Choose the name from boards.h that matches your setup
130 #ifndef MOTHERBOARD
131 | #define MOTHERBOARD BOARD_RAMPS_14_EFB
132 #endif
```

```
129 // Choose the name from boards.h that matches your setup
130 v #ifndef MOTHERBOARD
131 | #define MOTHERBOARD BOARD_BTT_SKR_V1_3|
132 #endif
```

Then close the **boards.h** file.

Next you want to make sure that you have the correct PlatformIO environment chosen (essentially which processor your board uses).

Open up the **platformio.ini** file. (see example image)



Find the **default_envs**.

```
15 # Remove '-fmax-errors=5' from build_flags below to see all.
16 #
17
18 [platformio]
19 src_dir = Marlin
20 boards_dir = buildroot/share/PlatformIO/boards
21 default_envs = mega2560
22
23 [common]
24 default_src_filter = +<src/*> -<src/config> -<src/HAL> +<src/HAL/shared>
```

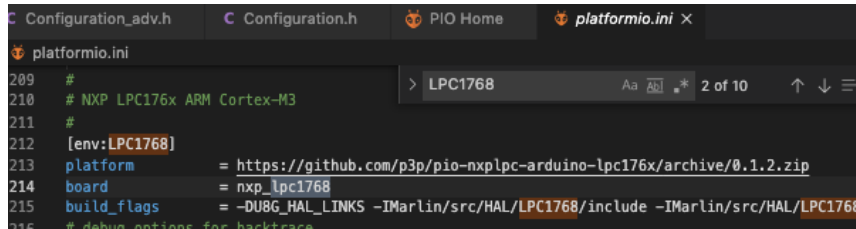
This is the environment you will be working in / the processor that your board uses.

Use **Ctrl+F** to open the find function and type the name of the processor to find it in this file.

My example board is a BigTreeTech SKR v1.3 and it uses an LPC1768 (see example image)

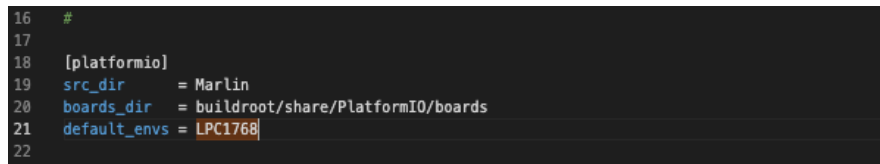
MUTANT

Marlin Tool Changing Feature Set Up Guide



```
platformio.ini
209 #
210 # NXP LPC176x ARM Cortex-M3
211 #
212 [env:LPC1768]
213 platform = https://github.com/p3p/pio-nxplpc-arduino-lpc176x/archive/0.1.2.zip
214 board = nxp_lpc1768
215 build_flags = -DUBG_HAL_LINKS -IMarlin/src/HAL/LPC1768/include -IMarlin/src/HAL/LPC1768
216 # debug options for backtrace
```

Copy the name after env: for the corresponding processor (in my case it is LPC1768) and paste it at the top of this page after **default_envs =** (see example image)



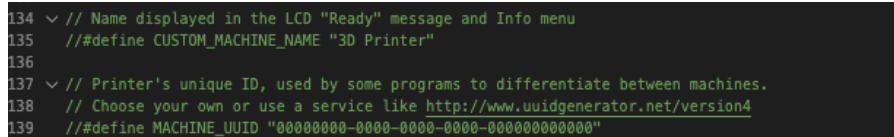
```
16 #
17
18 [platformio]
19 src_dir = Marlin
20 boards_dir = buildroot/share/PlatformIO/boards
21 default_envs = LPC1768
22
```

Then for good measure, save, compile, and upload to see if all went well. And now we can close the platformIO.ini file.

CONFIGURATION

Now we move back to the **Configuration.h** file.

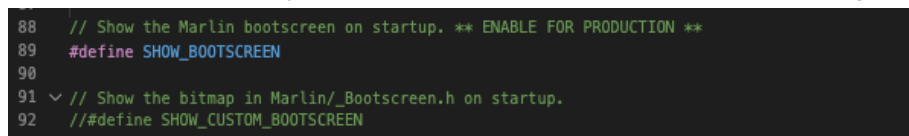
One of the first things at the top we are going to see is **CUSTOM_MACHINE_NAME**. (see example image)



```
134 // Name displayed in the LCD "Ready" message and Info menu
135 // #define CUSTOM_MACHINE_NAME "3D Printer"
136
137 // Printer's unique ID, used by some programs to differentiate between machines.
138 // Choose your own or use a service like http://www.uuidgenerator.net/version4
139 // #define MACHINE_UUID "00000000-0000-0000-0000-000000000000"
```

If you want to name the printer, you can uncomment this and change it to whatever you want within the parentheses. This will show up on the info screen when the printer boots.

A little bit down from there you will find **Boot Screen** options. (see example image)



```
88 // Show the Marlin bootscreen on startup. ** ENABLE FOR PRODUCTION **
89 #define SHOW_BOOTSCREEN
90
91 // Show the bitmap in Marlin/_Bootscreen.h on startup.
92 // #define SHOW_CUSTOM_BOOTSCREEN
```

Here you can turn the marlin boot screen off by commenting out **#define SHOW_BOOTSCREEN**. You can also add a custom boot screen and the configuration on that depends on which LCD you have. Absolutely worth a google, I have seen some pretty cool screens including animations.

After that, you should see **SERIAL_PORT**. (see example image)

MUTANT

Marlin Tool Changing Feature Set Up Guide

```
99  ✓ /**
100  * Select the serial port on the board to use for communication with the host.
101  * This allows the connection of wireless adapters (for instance) to non-default port pins.
102  * Serial port -1 is the USB emulated serial port, if available.
103  * Note: The first serial port (-1 or 0) will always be used by the Arduino bootloader.
104  *
105  * :[-1, 0, 1, 2, 3, 4, 5, 6, 7]
106  */
107  #define SERIAL_PORT 0
108
```

If for some reason you just cannot connect to the board, sometimes, changing this setting to 1 or -1 will fix that issue.

Then we are going to see the **BAUDRATE**. (see example image)

```
117  *
118  * 250000 works in most cases, but you might try a lower speed if
119  * you commonly experience drop-outs during host printing.
120  * You may try up to 1000000 to speed up SD file transfer.
121  *
122  * :[2400, 9600, 19200, 38400, 57600, 115200, 250000, 500000, 1000000]
123  */
124  #define BAUDRATE 250000
125
126  // Enable the Bluetooth serial interface on AT90USB devices
```

This is the speed at which the controller communicates over the port. Another setting to consider if the board is connecting but nothing seems to be happening or if it is not connecting at all.

We are going to skip **MOTHERBOARD** because we have already configured that.

Next, we will define how many extruders we are going to be using. (see example image)

```
143  ✓ // This defines the number of extruders
144  // :[1, 2, 3, 4, 5, 6, 7, 8]
145  #define EXTRUDERS 1
146
```

There are multiple ways to set up the MUTANT attachment, Make sure you set this number to the maximum number of extruder motors you will be using. If you are only going to be using one extruder motor per attachment, set this number to 1 for simplicity.

Next, you can set the default filament diameter.

```
146
147  // Generally expected filament diameter (1.75, 2.85, 3.0, ...). Used for Volumetric, Filament Width
148  #define DEFAULT_NOMINAL_FILAMENT_DIA 3.0
149
```

If you do not intend on setting up the **Select tool function** then you are going to want to set this parameter. Otherwise you can skip it as the **Select tool function** sets this parameter each time a tool is selected. (see example image)

Next comes the **thermal settings**. (see example image)

MUTANT

Marlin Tool Changing Feature Set Up Guide

```
345 //=====
346 //===== Thermal Settings =====
347 //=====
348
349 /**
350 * --NORMAL IS 4.7kohm PULLUP!-- 1kohm pullup can be used on hotend sensor, using correct resist
351 *
352 * Temperature sensors available:
353 *
354 * -5 : PT100 / PT1000 with MAX31865 (only for sensors 0-1)
355 * -3 : thermocouple with MAX31855 (only for sensors 0-1)
356 * -2 : thermocouple with MAX6675 (only for sensors 0-1)
357 * -4 : thermocouple with AD8495
358 * -1 : thermocouple with AD595
359 * 0 : not used
360 * 1 : 100k thermistor - best choice for EPCOS 100k (4.7k pullup)
361 * 331 : (3.3V scaled thermistor 1 table for MEGA)
362 * 332 : (3.3V scaled thermistor 1 table for DUE)
363 * 2 : 200k thermistor - ATC Semitec 204GT-2 (4.7k pullup)
364 * 202 : 200k thermistor - Copymaster 3D
365 * 3 : Mendel-parts thermistor (4.7k pullup)
```

CAUTION: Make sure that you do your research here and figure out what thermistor and heater cartridges you are using and make sure you read through this part very carefully as selecting the wrong setting has potential to cause a fire due to improper temp readings. Though that is almost impossible with all thermal runaway settings left on by default, it is still always a good idea to take caution when working with these numbers.

Next up is **PID settings** (see example image)

```
3 //=====
4 //===== PID Settings =====
5 //=====
6 // PID Tuning Guide here: http://reprap.org/wiki/PID\_Tuning
7
```

Though these settings are very important, we are not going to worry about them here because you can use a function in the firmware called autotune.

It will automatically set your PID settings for the heater cartridge and your heated bed.

To use the auto-tune function you are going to connect to your printer through a program like pronterface or octoprint and send the command M303 to the printer through its terminal.

It must be in the following format:

To autotune the hot end at 210°C for eight cycles you would send the code - **M303 E0 C8 S210**

To autotune the bed at 60°C for eight cycles you would send the code - **M303 E-1 C8 S60**

Next up is **thermal runaway protection settings**.(see example image)

```
570 //=====
571 //===== Thermal Runaway Protection =====
572 //=====
573
574 /**
575 * Thermal Protection provides additional protection to your printer from damage
576 * and fire. Marlin always includes safe min and max temperature ranges which
577 * protect against a broken or disconnected thermistor wire.
578 *
579 * The issue: If a thermistor falls out, it will report the much lower
580 * temperature of the air in the room, and the the firmware will keep
581 * the heater on.
582 *
583 * If you get "Thermal Runaway" or "Heating failed" errors the
584 * details can be tuned in Configuration_adv.h
585 */
586
587 #define THERMAL_PROTECTION_HOTENDS // Enable thermal protection for all extruders
588 #define THERMAL_PROTECTION_BED // Enable thermal protection for the heated bed
589 #define THERMAL_PROTECTION_CHAMBER // Enable thermal protection for the heated chamber
590
```

MUTANT

Marlin Tool Changing Feature Set Up Guide

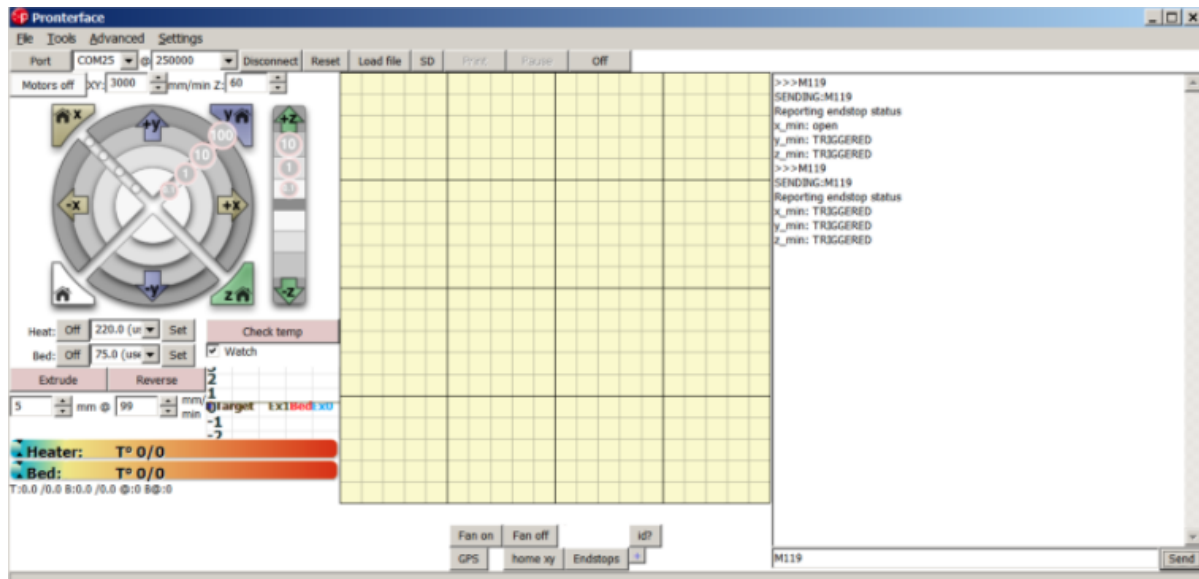
IMPORTANT: All of these are turned on by default. I suggest you leave it that way to be safe. These settings make sure that if something is not heating up the way that it should, the printer halts to try to prevent a fire.

Next is the endstop settings. (see example image)

```
606 //=====
607 //===== Endstop Settings =====
608 //=====
609
610 // @section homing
611
612 // Specify here all the endstop connectors that are connected to any endstop or probe.
613 // Almost all printers will be using one per axis. Probes will use one or more of the
614 // extra connectors. Leave undefined any used for non-endstop and non-probe purposes.
615 #define USE_XMIN_PLUG
616 #define USE_YMIN_PLUG
617 #define USE_ZMIN_PLUG
618 //#define USE_XMAX_PLUG
619 //#define USE_YMAX_PLUG
620 //#define USE_ZMAX_PLUG
621
```

Endstops are the switches or sensors that tell the printer that an axis has reached its limit.

You can use a program like pronterface or octoprint to send the **command M119** to the board. The board should spit back the status of the endstops. You can use that information to help you troubleshoot what needs to be fixed to get the endstops working properly. (see example image)



If it reads **OPEN** then the switch is not currently activated. When a switch has been activated it will say **TRIGGERED**. You must send the code each time you want to check the status of the endstops as the information that displays is not a live reading.

The next step is the **stepper drivers**.

This is where you are going to configure which stepper drivers that you have installed. (see example images)

MUTANT

Marlin Tool Changing Feature Set Up Guide

```
656 > stepper drivers Aa 1 of 5
657 /**
658  * Stepper Drivers
659  *
660  * These settings allow Marlin to tune stepper driver timing and enable advanced options for
661  * Stepper drivers that support them. You may also override timing options in Configuration_adv
662  *
663  * A4988 is assumed for unspecified drivers.
664  *
665  * Options: A4988, A5984, DRV8825, LV8729, L6470, L6474, POWERSTEP01,
666  *          TB6560, TB6600, TMC2100,
667  *          TMC2130, TMC2130_STANDALONE, TMC2160, TMC2160_STANDALONE,
668  *          TMC2208, TMC2208_STANDALONE, TMC2209, TMC2209_STANDALONE,
669  *          TMC26X, TMC26X_STANDALONE, TMC2660, TMC2660_STANDALONE,
670  *          TMC5130, TMC5130_STANDALONE, TMC5160, TMC5160_STANDALONE
671  * :['A4988', 'A5984', 'DRV8825', 'LV8729', 'L6470', 'L6474', 'POWERSTEP01', 'TB6560', 'TB6600'
672  */
673 //#define X_DRIVER_TYPE  A4988
674 //#define Y_DRIVER_TYPE  A4988
675 //#define Z_DRIVER_TYPE  A4988
676 //#define X2_DRIVER_TYPE A4988
677 //#define Y2_DRIVER_TYPE A4988
678 //#define Z2_DRIVER_TYPE A4988
679 //#define Z3_DRIVER_TYPE A4988
680 //#define Z4_DRIVER_TYPE A4988
681 //#define E0_DRIVER_TYPE A4988
682 //#define E1_DRIVER_TYPE A4988
```

Uncomment the drivers you will be using, and copy and paste the name of the ones you are using into where they belong unless you are using A4988 drivers which are set by default without uncommenting anything.

Next we are going to set the **default steps per unit**. (see example images)

```
728 /**
729  * Default Axis Steps Per Unit (steps/mm)
730  * Override with M92
731  * X, Y, Z, E0 [, E1[, E2...]]
732  */
733 #define DEFAULT_AXIS_STEPS_PER_UNIT { 80, 80, 4000, 500 }
734
```

This is the number of steps that the stepper driver needs to make to move the tool head 1mm. You will need to calibrate these numbers. There are a couple of different ways to figure out what you need to put here. There are calculators online such as the one found at https://blog.prusaprinters.org/calculator_3416/ (see example image)

Stepper Motors

Steps per millimeter - belt driven systems

The result is theoretically right, but you might still need to calibrate your machine to get finest detail. This is good start tho.

If you struggle how to use this calculator, try aksing in [i steps per mm forum](#).

Motor step angle	Driver microstepping				
1.8° (200 per revolution)	1/16 - uStep (mostly Pololu)				
Belt pitch (in mm)	Belt presets				
2	2mm Pitch (GT2 mainly)				
Pulley tooth count					
8					
Result	Resolution	Teeth	Step angle	Stepping	Belt
200.00 Click to Share!	5micron	8	1.8°	1/16th	2mm

And then you should also use a micrometer to fine tune and make sure that when you tell the machine to move a specific distance, it moves exactly that distance.

MUTANT

Marlin Tool Changing Feature Set Up Guide

Next, we are going to move on to the **Z Min Probe**. (see example images)

```
816
817  ✓ //=====
818  //===== Z Probe Options =====
819  //=====
820  // @section probes
821
822  ✓ //
823  // See http://marlinfw.org/docs/configuration/probes.html
824  //
825
826  ✓ /**
827   * Z_MIN_PROBE_USES_Z_MIN_ENDSTOP_PIN
828   *
829   * Enable this option for a probe connected to the Z Min endstop pin.
830   */
831  #define Z_MIN_PROBE_USES_Z_MIN_ENDSTOP_PIN
832
833  ✓ /**
834   * Z_MIN_PROBE_PIN
835   *
836   * Define this pin if the probe is not connected to Z_MIN_PIN.
837   * If not defined the default pin for the selected MOTHERBOARD
838   * will be used. Most of the time the default is what you want.
839   *
840   * - The simplest option is to use a free endstop connector
```

Here in this section you are going to define what kind of sensor you use for leveling the bed, such as BLtouch or a microswitch.

A little bit down from there you will see a spot to **Invert the stepper direction**. (see example image)

```
050
051  // @section extruder
052
053  #define DISABLE_E false           // For all extruders
054  #define DISABLE_INACTIVE_EXTRUDER // Keep only the active extruder enabled
055
056  // @section machine
057
058  // Invert the stepper direction. Change (or reverse the motor connector) if an axis goes the wrong way.
059  #define INVERT_X_DIR false
060  #define INVERT_Y_DIR true
061  #define INVERT_Z_DIR false
062
063  // @section extruder
064
065  // For direct drive extruder v9 set to true, for geared extruder set to false.
066  #define INVERT_E0_DIR false
067  #define INVERT_E1_DIR false
068  #define INVERT_E2_DIR false
069  #define INVERT_E3_DIR false
070  #define INVERT_E4_DIR false
071  #define INVERT_E5_DIR false
072  #define INVERT_E6_DIR false
073  #define INVERT_E7_DIR false
```

This is where you can change the direction of one of your stepper motors if it is going the wrong direction.

Just below that, you can change the direction that each axis homes. (see example image)

```
086  ✓ // Direction of endstops when homing; 1=MAX, -1=MIN
087  // :[-1,1]
088  #define X_HOME_DIR -1
089  #define Y_HOME_DIR -1
090  #define Z_HOME_DIR -1
091
```

This is the direction that the axis will move when you tell it to go home and the direction where the end stop will be.

MUTANT

Marlin Tool Changing Feature Set Up Guide

Next you will find the size of the print bed. The size is in millimeters. (see example image)

```
1093
1094 // The size of the print bed
1095 #define X_BED_SIZE 200
1096 #define Y_BED_SIZE 200
1097
```

Next we will find settings for a filament runout sensor. (see example image)

```
1135 /**
1136  * Filament Runout Sensors
1137  * Mechanical or opto endstops are used to check for the presence of filament.
1138  *
1139  * RAMPS-based boards use SERVO3_PIN for the first runout sensor.
1140  * For other boards you may need to define FIL_RUNOUT_PIN, FIL_RUNOUT2_PIN, etc.
1141  * By default the firmware assumes HIGH=FILAMENT PRESENT.
1142  */
1143 //#define FILAMENT_RUNOUT_SENSOR
1144 #if ENABLED(FILAMENT_RUNOUT_SENSOR)
1145   #define NUM_RUNOUT_SENSORS 1 // Number of sensors, up to one per extruder. Define a FIL
1146   #define FIL_RUNOUT_INVERTING false // Set to true to invert the logic of the sensor.
1147   #define FIL_RUNOUT_PULLUP // Use internal pullup for filament runout pins.
1148   //#define FIL_RUNOUT_PULLDOWN // Use internal pulldown for filament runout pins.
1149
1150   // Set one or more commands to execute on filament runout.
1151   // (After 'M412 H' Marlin will ask the host to handle the process.)
1152   #define FILAMENT_RUNOUT_SCRIPT "M600"
1153
1154   // After a runout is detected, continue printing this length of filament
1155   // before executing the runout script. Useful for a sensor at the end of
1156   // a feed tube. Requires 4 bytes SRAM per sensor, plus 4 bytes overhead.
1157   //#define FILAMENT_RUNOUT_DISTANCE_MM 25
1158
1159   #ifndef FILAMENT_RUNOUT_DISTANCE_MM
1160     // Enable this option to use an encoder disc that toggles the runout pin
1161     // as the filament moves. (Be sure to set FILAMENT_RUNOUT_DISTANCE_MM
1162     // large enough to avoid false positives.)
1163     //#define FILAMENT_MOTION_SENSOR
1164   #endif
1165 #endif
1166
```

This is for making sure that the machine pauses if you run out of filament.

A few additional settings are needed in order to get this function to work, but that all depends on the configuration that you have loaded and whether or not your printer already has this function to begin with.

After turning the setting on, compile it, and if there is something else that needs to be turned on, there will be a compile error giving you instructions on how to fix the issue.

Next will be the settings for **bed leveling**.

There is a description of each type of bed leveling in this section. (see example image)

```
167 //=====
168 //===== Bed Leveling =====
169 //=====
170 // @section calibrate
171
172 /**
173  * Choose one of the options below to enable G29 Bed Leveling. The parameters
174  * and behavior of G29 will change depending on your selection.
175  *
176  * If using a Probe for Z Homing, enable Z_SAFE_HOMING also!
177  *
178  * - AUTO_BED_LEVELING_3POINT
179  * Probe 3 arbitrary points on the bed (that aren't collinear)
180  * You specify the XY coordinates of all 3 points.
181  * The result is a single tilted plane. Best for a flat bed.
182  *
183  * - AUTO_BED_LEVELING_LINEAR
184  * Probe several points in a grid
```

Next, we're going to go to the additional features. First under that list is **EEPROM** settings. (see example image)

MUTANT

Marlin Tool Changing Feature Set Up Guide

```
30 //=====
31 //===== Additional Features =====
32 //=====
33
34 // @section extras
35
36 /**
37  * EEPROM
38  *
39  * Persistent storage to preserve configurable settings across reboots.
40  *
41  * M500 - Store settings to EEPROM.
42  * M501 - Read settings from EEPROM. (i.e., Throw away unsaved changes)
43  * M502 - Revert settings to "factory" defaults. (Follow with M500 to init the EEPROM.)
44  */
45 #define EEPROM_SETTINGS // Persistent storage with M500 and M501
46 // #define DISABLE_M502 // Saves ~2700 bytes of PROGMEM - Disable for release!
```

You need to **uncomment #define eeprom_settings**. This is necessary for you to be able to save settings into the EEPROM including the select tool menu item.

Next we are going to go down to **LCD and SD support**.

Uncomment #define SDSUPPORT if you intend on using a SD card in your LCD. (see example image)

```
1652 /**
1653  * SD CARD
1654  *
1655  * SD Card support is disabled by default. If your controller has an SD slot,
1656  * you must uncomment the following option or it won't work.
1657  *
1658  */
1659 #define SDSUPPORT
```

Encoder settings are going to be the settings you change if the knob on your LCD is either going the wrong direction or if it's acting sporadically. (see example image)

```
1688 // ENCODER SETTINGS
1689 //
1690 // This option overrides the default number of encoder pulses needed to
1691 // produce one step. Should be increased for high-resolution encoders.
1692 //
1693 // #define ENCODER_PULSES_PER_STEP 4
1694
```

Then, you are going to choose what kind of LCD you have. There are lots of different types. It does help to use ctrl+f to find the one you have by name. (see example image)

MUTANT

Marlin Tool Changing Feature Set Up Guide

```
1759 //=====
1760 //===== LCD / Controller Selection =====
1761 //===== (Character-based LCDs) =====
1762 //=====
1763
1764 //
1765 // RepRapDiscount Smart Controller.
1766 // http://reprap.org/wiki/RepRapDiscount\_Smart\_Controller
1767 //
1768 // Note: Usually sold with a white PCB.
1769 //
1770 #define REPRAP_DISCOUNT_SMART_CONTROLLER
1771
1772 //
1773 // Original RADDs LCD Display+Encoder+SDCardReader
1774 // http://doku.radds.org/dokumentation/lcd-display/
1775 //
1776 //#define RADDs_DISPLAY
```

In some cases, a screen may need a library installed to work properly. Read through the section where your LCD is, it should tell you what is needed right there, if anything is needed. If not, once again, just compile and check for errors.

You did it!

That about wraps it up for **the configuration.h**.

Just make sure you read through everything thoroughly, there are a lot of extra settings, and lots of experimentation to get some of them working.

Don't be afraid to test something out, but remember to save it, compile it, upload it and see how each function works as you enable it.

The console will tell you if something needs to be changed in order for it to properly compile.

Configuration_adv.h is set up in a very similar fashion as configuration.h.

The settings there are, as it sounds, more advanced.

Read through it thoroughly just like the configuration.h file.

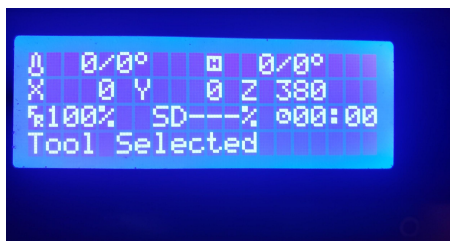
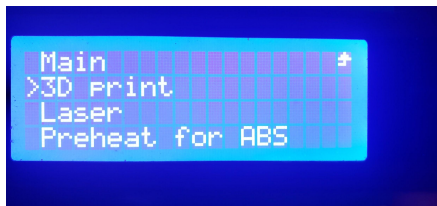
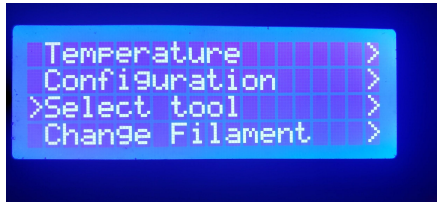
If there's anything in there that is unclear, you can always go to the Marlin website <https://marlinfw.org/> and search for any line within the code that you do not understand.

MUTANT

Marlin Tool Changing Feature Set Up Guide

Using your Machine Interface

It is easy to change between tools on your 3D printer. Just navigate to Control> Select Tool > (name of desired tool)> click to confirm



Written and created by
Joshua With
mrjoshuafixit@gmail.com
@MrJoshuaFixit

For video version please see Chris Riley's <https://youtu.be/JWUcnkjeNU8>