



Lite6 Marlin Configuration


Set up your Marlin Firmware to support your new E3D HotEnd

Written By: Gabe S.



Step 1 — Download Marlin



- First things first: you're going to need a copy of Marlin.
 - If you are upgrading an existing 3D printer to use a V6 HotEnd, you should try to get a copy of your current firmware from your printer's manufacturer.
 - If you're building a new printer, or simply want to upgrade to the latest version of Marlin, download it at <http://marlinfw.org/meta/download/>
-  If you download a fresh version of Marlin you'll have to configure more settings than the ones mentioned in this guide so that it will work well with your printer.

Step 2 — Download Arduino



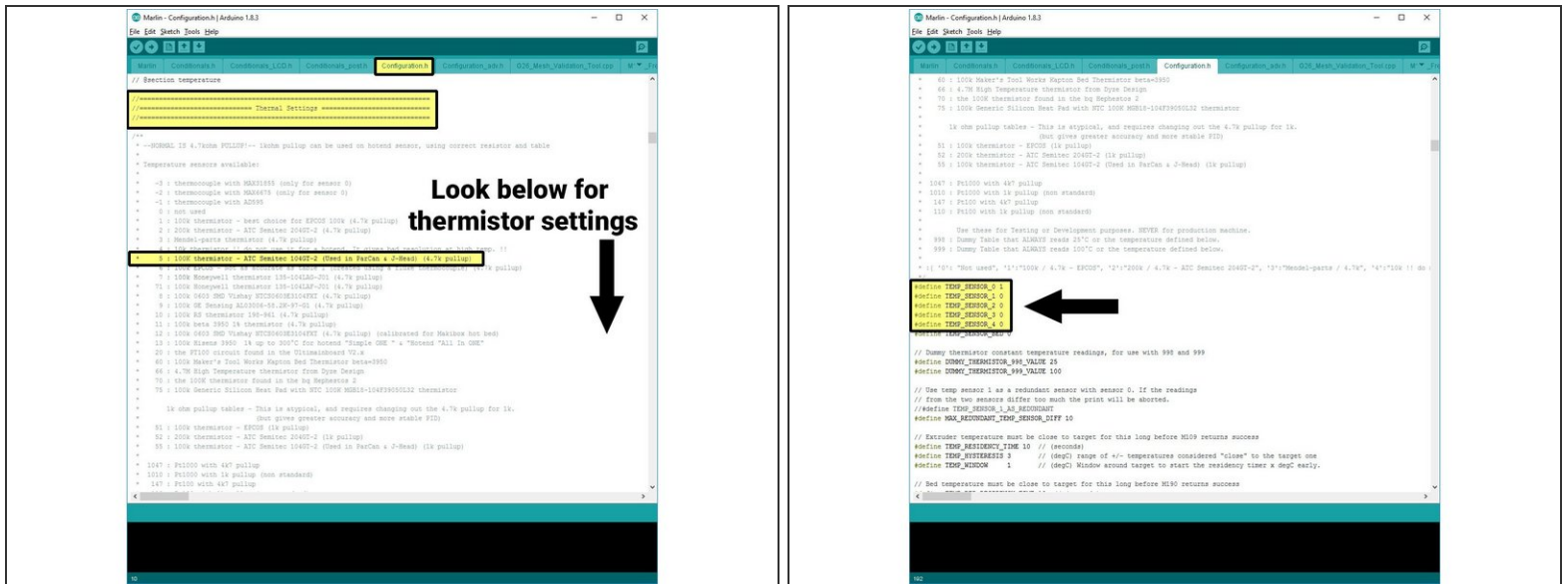
- Almost all printers use Arduino IDE to upload fresh firmware, so download it at <https://www.arduino.cc/en/Main/Software>

Step 3 — Open Marlin in Arduino



- Unzip Marlin from the zip file you downloaded and put the resulting folder anywhere on your computer for safe keeping.
- Inside this folder, navigate to the Marlin sub-folder, and open the Marlin.ino file. This should open every file in Marlin.
- Find the Configuration.h file

Step 4 — Thermistor Settings



- In the configuration.h file, find the Thermal Settings section. Below the comments you'll find the settings for the types of thermistors your printer uses. (Typically there will be one per hotend and one more if you have a heated bed.)
- If you're installing your Lite6 as your only HotEnd, change the first highlighted line to: `#define TEMP_SENSOR_0 5`
- If you're replacing an existing HotEnd or have multiple HotEnds, adjust whichever line corresponds to the tool number that you're changing (they start counting from 0)

Step 5 — Set Maximum Temperature

```

Marlin - Configuration.h [Arduino 1.8.3]
File Edit Sketch Tools Help
Main Configuration.h Conditionals_LCD.h Conditionals_post.h Configuration.h Configuration_adv.h GCode_Mesh_Valuation_Tools.h
// Bed temperature must be close to target for this long before M190 returns success
#define TEMP_RESIDENCY_TIME 10 // (seconds)
#define TEMP_HYSTERESIS 3 // (degC) range of +/- temperatures considered "close" to the target one
#define TEMP_WINDOW 1 // (degC) Window around target to start the residency timer x degC early.

// The minimal temperature defines the temperature below which the heater will not be enabled It is used
// to check that the wiring to the thermistor is not broken.
// Otherwise this would lead to the heater being powered on all the time.
#define HEATER_0_MINTEMP 5
#define HEATER_1_MINTEMP 5
#define HEATER_2_MINTEMP 5
#define HEATER_3_MINTEMP 5
#define BED_MINTEMP 5

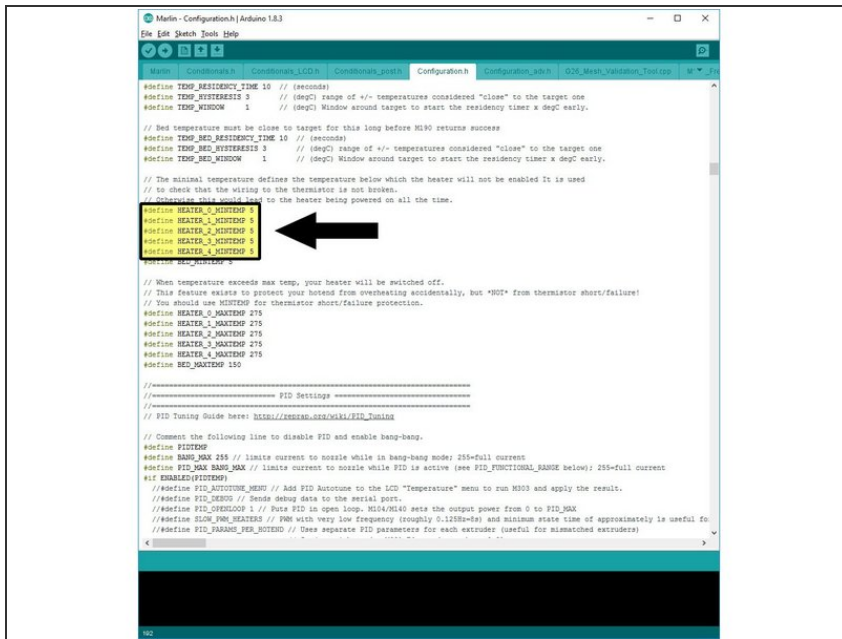
// When temperature exceeds max temp, your heater will be switched off.
// This feature works to protect your filament from overheating accidentally, but "M0" from thermistor short/failure!
// See https://www.repetier.com/forum/thermistor-short-failure-protection.
#define HEATER_0_MAXTEMP 275
#define HEATER_1_MAXTEMP 275
#define HEATER_2_MAXTEMP 275
#define HEATER_3_MAXTEMP 275
#define BED_MAXTEMP 150

//===== PID Settings =====
//===== PID Tuning Guide here: https://repetier.com/wiki/PID\_tuning =====
// Comment the following line to disable PID and enable bang-bang.
#define PIDTEMP
#define BANG_MAX 255 // limits current to max while in bang-bang mode; 255=full current
#define PID_MAX_BANG_MAX // limits current to max while PID is active (see PID_FUNCTIONAL_RANGE below); 255=full current
#define ENABLED(PIDTEMP)
//#define PID_AUTOTUNE_MENU // Add PID Autotune to the LCD "Temperature" menu to run M303 and apply the result.
//#define PID_DEBUG // Sends debug data to the serial port.
//#define PID_OPENLOOP 1 // Puts PID in open loop. M304/M340 sets the output power from 0 to PID_MAX
//#define SLOW_PWM_HEATERS // PWM with very low frequency (roughly 0.128Hz) and minimum state time of approximately 1ms is useful fo
//#define PID_PARAMS_PER_HOTEND // Uses separate PID parameters for each extruder (useful for mismatched extruders)

```

- i While the metal of your new V6 hotend can withstand very high temperatures, the PTFE lining of your Lite6 can only go up to ~245°C for extended periods.
- Set that as your maximum temperature by changing the first highlighted line to: **#define HEATER_0_MAXTEMP 245** (or adjust the line corresponding to the hotend you're changing.)

Step 6 — Set Minimum Temperature



```

Marlin - Configuration.h [Arduino 1.8.3]
File Edit Sketch Tools Help
Main Configuration.h Conditionals_LCD.h Conditionals_pos.h Configuration.h Configuration_adv.h G29_Mesh_Visualization_Tools.h

#define TEMP_RESISTENCY_TIME 10 // (seconds)
#define TEMP_HYSTERESIS 0 // (degC) range of +/- temperatures considered "close" to the target one
#define TEMP_WINDOW 1 // (degC) Window around target to start the residency timer x degC early.

// Bed temperature must be close to target for this long before M190 returns success
#define TEMP_BED_RESIDENCY_TIME 10 // (seconds)
#define TEMP_BED_HYSTERESIS 0 // (degC) range of +/- temperatures considered "close" to the target one
#define TEMP_BED_WINDOW 1 // (degC) Window around target to start the residency timer x degC early.

// The minimal temperature defines the temperature below which the heater will not be enabled It is used
// to check that the wiring to the thermistor is not broken.
// Comment out this line to allow the heater to be powered on all the time.
#define HEATER_0_MINTEMP 5
#define HEATER_1_MINTEMP 5
#define HEATER_2_MINTEMP 5
#define HEATER_3_MINTEMP 5
#define BED_MINTEMP 5

// When temperature exceeds max temp, your heater will be switched off.
// This feature exists to protect your heater from overheating accidentally, but "M03" from thermistor short/failure!
// You should use M303 for thermistor short/failure protection.
#define HEATER_0_MAXTEMP 275
#define HEATER_1_MAXTEMP 275
#define HEATER_2_MAXTEMP 275
#define HEATER_3_MAXTEMP 275
#define BED_MAXTEMP 150

//===== PID Settings =====
//===== PID Tuning Guide here: https://reprap.org/wiki/PID_Tuning =====

// Comment the following line to disable PID and enable bang-bang.
#define PIDTEMP
#define PID_MAX_CURRENT 255 // limits current to max while in bang-bang mode; 255=full current
#define PID_MAX_BANG_MAX // limits current to max while PID is active (see PID_FUNCTIONAL_RANGE below); 255=full current
#define ENABLE_PIDTEMP
// #define PID_AUTOTUNE_MENU // Add PID Autotune to the LCD "Temperature" menu to run M303 and apply the result.
// #define PID_SEND // Send debug data to the serial port.
// #define PID_OPENLOOP 1 // Puts PID in open loop. M304/M340 sets the output power from 0 to PID_MAX
// #define SLOW_PWM_HEATERS // PWM with very low frequency (roughly 0.12Hz) and minimum state time of approximately 1s is useful fo
// #define PID_PARAMS_PER_EXTRUDER // Use separate PID parameters for each extruder (useful for mismatched extruders)

```

- For safety it is strongly recommended to make sure that your printer detects if the thermistor stops sending correct temperatures for any reason.
- Set the first highlighted line to: `#define HEATER_0_MINTEMP 5` (or adjust the line corresponding to the hotend you're changing.)
- In newer versions of Marlin there are other features such as Thermal Runaway Protection that might be useful as well, though they are typically on by default.

Step 7 — Upload Firmware



- Upload the new firmware to your electronics as you normally would. Typically this means plugging in your printer to your computer, selecting the correct COM port and board type, and pressing the upload button.
- If you're unsure of how to update your printer's firmware, check with its manufacturer.

Head back to the [Lite6 Assembly](#) page to finish the last few steps before you start printing.