Marlin Setup with New/Custom Printer

After jerry-rigging an old i3 to work with a spare e3 mainboard, I have a lot of very useful information on setting up a marlin printer from scratch. Firstly, you want to make sure all the hardware is configured correctly and wired properly. This is mainboard-dependent, so make sure you reference relevant documentation for your mainboard of choice.

Some relevant reading material:

- Marlin Configuration Docs
- Marlin Troubleshooting Guide

Setup Marlin for Build

1. Install vscode (ew) and make sure you have the following extensions installed:
    - Patformio
    - Auto Build Marlin
2. Clone the repo then open in vscode
3. Open Auto Build Marlin
4. Navigate to Marlin Firmware Configurations and pick a configuration that is similar/matches your hardware
5. Copy the relevant `*.h` files to `Marlin/` in the repository.
6. Build using auto build marlin.
7. Verify It Builds

Configuring Marlin

The following is a list of steps to take when setting up the configuration for your printer. Generally speaking if you have any calibrated values (i.e. extruder Esteps, hotend/bed PID parameters) you should get the values from the current firmware (ie `M503`) and then edit the ones in the new firmware build so that your calibrated values are the defaults. There also may be need for post-firmware configuration (such as home offsets and zprobe offsets). The following is a general list of stuff to setup and make sure are present (I would hope the printer you are targeting has most of these features).

At a baseline, identify the following characteristics of the printer and change the configs to match:

- Motion System
- Mainboard make/build
- Number of Extruders
- Motor Driver type
- Probes
- Termistors/Heaters
- LCD or display

Generally speaking, if you are running some commercially-available board, there will be a configuration in the examples on the marlin repository. If you are running a custom mainboard, why are you reading this?.

Configuration Checklist (reference marlin docs for additional prose):

1. Motion System:
    - Style (ie corexy or cartiesian?)
    - XYZ Steps/mm (use values derived from datasheets FIRST)
    - E Steps/mm (usually a post-flash calibration value, favor undershooting if you dont know)
    - Sensorless / Endstop Homing (sensorless usually requires setting jumpers on the mainboard)
    - Stepper Driver Type (ie TMC2209)
2. Print Volume / Travel
    - Bed Size
    - Home offsets
    - Min/Max XYZ motion
3. Z-Probe
    - Type (ie inductive, bltouch, etc)
    - Probe is endstop?
    - Probe XY Offset to nozzle (Z is usually configured later, keep it at 0 for now)
    - Z Safe Homing (this is basically a requirement if your Z endstop is also a probe)
    - Temperature Compensation?
    - Offset config wizard?
4. Thermal Performance / PID
    - Max Hotend & Bed Temps?
    - PID control enabled for both Hotend & Bed (i.e. not PID for hotend and bang-bang for bed)
    - Autotune PID menu option (otherwise use gcode)
    - Thermal Runaway tuning? (only really consider this if PID tuning still causes inaccurate thermal runaway and you have CONFIRMED it is not a hardware issue)
5. ABL
    - Bilinear vs UBL
    - min 3×3 grid, 2 probes per pt (2nd slowed, grid ideally is 5×5)
    - set `PROBING_MARGIN` to some value to offset the probe from bed edges
6. Miscellaneous Items
    - SD Card Reader Functionality
    - Printer Stats
    - Input shaping (though you should probably just use klipper for this)

Additionally, these are things you want to put in firmware after calibration:

- Probe z offset
- E steps/mm
- Hotend/Bed Kp, Kd, Ki PID values

Slicer Profile (OrcaSlicer)

This is for OrcaSlicer because I like it and it gives you the most granular settings control for the most general set of printers. When setting up a profile, I recommend basing your profile off of the generic printer profile in Orca, keeping in mind the firmware version (and this what gcode flavor) your printer is working in.

Base your default process template off a similar printer build. In the case of the i3, I could base it off any of the i3-style printers in Orca. I chose to base it off of what was working on my ender 3. Print

both the Voron Cube and Benchy for a robust cover of most print moves/orientations.

Start gcode suggestions

If you have a probe that has temperature sensitivity (ie pinda v1/v2) you may want to home before configuring z offest or get within the ballpark then adjust with babystepping with a test print.

Heating Sequence

```
G1 Y1.0 Z3 F1000                    ; move print head up
M104 S[first_layer_temperature]     ; set extruder temp
M140 S[first_layer_bed_temperature] ; set bed temp
G1 X0 Y0 F1000                      ; move to orgin
G0 Z100 F1000                       ; Move head up for heating
M190 S[first_layer_bed_temperature] ; wait for bed temp
M109 S[first_layer_temperature]     ; wait for extruder temp
G1 Z0.3 F1000                       ; move back down
G92 E0.0
G1 Y1.0 Z0.3 F1000                  ; move to neutral pos
```

Post-Setup Tuning / Firmware Adjustments

Access the printer over serial:

```
picocom -b 115200 /dev/ttyACM0 --imap lfcrlf --echo
```

- Printer frame skew: you can use calistar to compensate for frame skew if you demand high dimensional accuracy. Marlin will have options to add this information for compensation.

From:
https://memex.kyaruc.moe/ - **kyaruc memex**

Permanent link:
https://memex.kyaruc.moe/3dp:marlin

Last update: **2026-06-29 Mon 03:30**