2021

Marlin 2.0.7.2
for the Anycubic Chiron
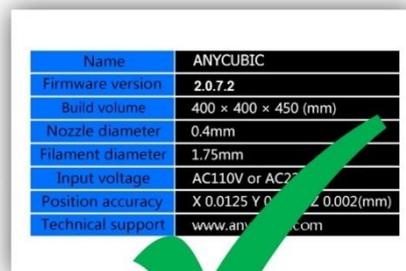
Nick Wells

3/28/2021

Contents

Please read this guide first!

There are many posts on the Chiron Facebook groups and I get many DMs from people having problems with the firmware. Nearly all of these problems could be resolved by reading this guide and following the instructions it contains.

The links to the firmware and source code are in this document.

So please do have a quick read first!

Disclaimer

I do not work for Anycubic and have no affiliation with Anycubic Ltd.

I have created this custom version of Marlin to support the factory build of the Anycubic Chiron, as I am a Chiron owner and have many years of experience coding with Marlin firmware.

You are installing this firmware because you want to try new features in Marlin and you know how to safely operate your machine.

This firmware has been tested on the standard factory build Chiron and the printer operates as expected. This is open-source firmware that comes with no support, warranty or guarantees.

You have chosen to install it this firmware and by doing so accept responsibility for any problems, failures and losses that may occur.

If you aren't sure about upgrading firmware, then please don't.
Get to know your machine before changing things!

The upgrade process is very simple and you can return to the Anycubic standard firmware at any time using the files in this package, or by downloading it from the Anycubic website.

Firmware Features

Here is a summary of the Marlin features enabled in the firmware

Machine Protection

- All Marlin thermal protection features are enabled.
- Thermal error detection has been retuned for the Chiron to prevent false errors.

TFT Panels

- Old and new style TFT panels are now supported!
- TFT panel type is automatically detected

Bed Levelling

- Mesh bed Levelling
- Sensor Bed Probing
- Manual level adjustment
- Enhanced single point adjustment
- Automatic mesh loading after homing
- Updated tolerances to resolve probing errors

Pre-set Calibration

- Stock Hotend PID tune profile
- Stock Heat-bed PID tune profile
- Stock extruder E-Step Calibration (405 Steps/mm)
- Increased speed and acceleration settings
- Automatic EEPROM initialisation

Error Recovery

- Filament out detection and recovery enabled
- Power loss detection and recovery enabled for SD Printing only
  (Class 10 SD card recommended)
- Auto park head on Pause and filament change
- Validated temperature error detection (reduces spurious temperature warnings)
- Last machine error is displayed on the firmware info page

Printing Features

- Marlin Linear Advance enabled (preset K=0.05)
- Marlin Junction Deviation (preset to 0.028)
- Live Z axis baby stepping movement from panel during print
- M600 filament change during print
- M125 job pause during print
- Auto head park during pause/filament change
- Host action notification enabled
- Meat packer data compression enabled
- Idle Hotend heater shutdown after 5 minutes

SD Card file listing

- Flat file folder navigation
- PC style folder navigation
- Patch applied https://github.com/MarlinFirmware/Marlin/pull/20035

Which Version Chiron do I have?

There are newer factory builds of the Chiron with an updated TFT panel that works slightly differently.

This firmware will auto-detect and support both TFT panels!

Original version supported ✓   **New version now supported** ✓

Use stock Anycubic firmware v1.3.0   **Use stock Anycubic Firmware v1.3.5**

When the printer powers up, it will run a quick test to identify the type of TFT panel connected to the machine and will enable the required settings automatically.

What's in the package?

There are various files and folders in the zip package, so do have a look around!

The package includes:
- Marlin 2.0.7.2 firmware as compiled .hex files.
- Original Anycubic compiled .hex files v1.3.0 & v1.3.5
- Marlin 2.0.7.2 source code for the Chiron
- Termite terminal program
- XLoader firmware uploading tool

Firmware Installation Quick Start

Please follow these quick instructions to upgrade the printer and get printing. These instructions work with any version of firmware.

Before You Start

If you have already PID tuned and E-Step calibrated your printer it may be worth making a record of your settings. To do this connect a terminal program to the printer and send M503 to display settings. Save this info somewhere useful!

NOTE: The 2.0.7.2 will automatically reset the EEPROM with the new firmware defaults. So, save your old values first!   You have been warned!!

Upload the firmware

1) Make sure any programs that use the COM port are closed.
2) Connect your printer to a PC and launch Xloader
3) Browse to the Hex file you wish to upload
4) Set the Device to Mega(ATMEGA2560)
5) Select the correct COM port for your machine
6) Set the Baud Rate to 115200
7) Click the Upload button

Once the firmware upload has finished, your printer will reboot.

Why should I Reset the EEPROM?

All of the machine settings are stored in a chip called an EEPROM, which stay there when the power is turned off. (Electrically Erasable Programmable Read Only Memory) if you wanted to know!

Each version of firmware has its own way of storing information in the EEPROM and they are not compatible with each other.

If you change the version of firmware, **YOU MUST RESET THE EEPROM to factory settings**. I have had many questions from owners with problems caused by bad EEPROM settings.

So, this firmware will reset the EEPROM automatically when it needs to.
You only need to follow the instructions below if you are changing to a different firmware.

Either send the commands M502 M500 from your PC using a terminal tool,
or copy 'Reset EEPROM.gcode' onto an SD card and print it.

Error Reporting

The last reported machine error will be displayed the firmware info page.
When you display the firmware version on the TFT panel, the last printer error will be shown on the firmware version line ONLY ONCE.

The error will be cleared once it has been displayed and also if you reboot the printer.

The following errors can be displayed:

- **Abnormal_bed_temp**
  If the bed temperature is not within the normal operating range.
  This is usually due to a sensor or wiring failure.

- **Abnormal_hotend_temp**
  If the hot end temperature is not within the normal operating range.
  This is usually due to a sensor or wiring failure.

- **SD_card_error**
  If there has been a problem reading the SD card.

- **Filament_runout**
  The print has stopped because the filament sensor was triggered.

- **Power_failure**
  The print stopped because the power failed. (Onl shown is power loss recovery is operating)

- **EEPROM_ver_wrong**
  The EEPROM contains invalid information and default settings have been loaded.
  You should reset the EEPROM and relevel the printer.

Levelling the bed

Follow the instructions below to level the print bed. You only need to do this once, the levelling mesh is saved in the EEPROM and is automatically loaded each time you home the machine.

The bed levelling uses the Anycubic Probe which creates a 5 x 5 mesh that is used to adjust the Z axis position during print. The mesh is automatically saved after homing.
Despite the discouraging comments, this probe works really well!

Top Tip! The bed levelling will be more accurate if you probe the bed when it is hot!

Here are the bed levelling steps: **Please follow these steps carefully!**

1) Select Tools>More>Level>Advanced Setting and reset the ALL points value to -15.00
2) Home the printer
3) From the panel select Tools>More>Level>Probe make sure the probe is fitted then click OK. The printer will now probe the bed.
4) Once probing has finished, the head will park and you should remove the probe.
5) You now need to adjust ALL the mesh points to lower the nozzle onto the bed. Select Tools>More>Level>Advanced Setting
6) Tap one of the red dots twice and the head will move to that point and will be about 2mm above the bed.
7) **Check the 'ALL' box** and place a piece of paper under the head.
8) Use the +/- buttons to lower the head so that is grips the paper tightly.
9) Now click OK to save the new setting.
10) Double tap a couple of other points to make sure the paper is gripped in other locations.
11) The bed is now levelled and the settings will be saved once you click OK!

Single Mesh Point Adjustment

If you want to check and adjust individual points:
1) Home the printer
2) Go to the advanced level menu and click any of the red dots twice. The head will move to the probe point and you can adjust the level using the +/- buttons on the panel.

All Points Adjustment

If you want to lower the entire mesh:
1) Home the printer
2) Go to the advanced level menu and click any of the red dots twice.
3) Make sure the ALL box it ticked The head will move to the probe point and you can adjust the level using the +/- buttons on the panel. With the ALL box ticked, all of the mesh points will be altered together.

Pressing - will make the starting point lower (closer to the bed)
Pressing + will make the staring point higher (further away from the bed)

NOTE: When using the Advanced Setting screen
- **Click OK to save your changes**
- **Click back to cancel and revert to the last saved values.**

What is PID Tuning?

PID tuning calibrates the algorithms that control the heaters. When calibrated, the controller will keep the temperature stable, which is a good thing!

There are 2 heaters on the Chiron, one for the hot end (nozzle) and one for the heat bed. You need to calibrate these separately. You only need to do this once.

PID tune the hotend
1) Copy the file '`PID Tune Hotend.gcode`' to the SD card and print it.
2) The printer will beep when finished.
3) To manually tune send

PID tune the heat bed
1) Copy the file '`PID Tune Bed.gcode`' to the SD card and print it.
2) The printer will beep when finished.

What is STEP Calibration?

The motors on a 3D printer are called stepper motors. They are moved by a sequence of pulses sent to different windings or coils in the motor. The controller needs to know how far the parts of the printer moves with each step. The number is measured as steps per millimetre. (steps/mm). Most of this is already set in the firmware so you don't need to change anything. However, if you have fitted new motors, gears, or extruders then you should check the step calibration.

Calibrate the extruder E-STEPS
Please read the included guide 'Anycubic Chiron Step by Step Extruder flow calibration.pdf'

Special inserted GCode Commands

You can insert special Gcode commands into your print job if you want to do things such as pause the print or change the filament part way through a job.

The following commands are supported:

Job Pause from panel or M125 command in Gcode

When you hit pause, or the printer processes a 'M125' job pause command, the printer will pause the job and park the print head.

To resume the job, either click Resume on the TFT panel, or send a 'M108' command to the printer.

NOTE: **The extruder heater will shut down after 5 minutes.**
If this happens, you will need to resume twice to continue the print.

Filament Runout or M600 command in Gcode

When the filament runs out, or the printer processes a 'M600' filament eject command, a Marlin process is triggered that manages the unload and reload of the filament. As the front panel can't display instructions, you will hear a series of beeps when the print wants some help from you!

The filament change workflow looks like this:
1) The filament runs out or M600 is encountered

2) The printer beeps, parks the head and shows a warning on the panel

3) The printer ejects the remaining filament then beeps 5 times, waits and enables the continue button on the panel.

4) Insert the new filament into the extruder and click continue

5) The printer will slowly feed 50mm of filament

6) It will then accelerate and feed a further 500mm

7) Finally, it will slow down and purge the last 100mm

8) Once the filament starts to come out of the nozzle, click continue and the print will resume.

NOTE: **The extruder heater will shut down after 5 minutes. If this happens, once you reach step 4, the printer will heat up the extruder and will beep again. You need to click continue and the reload process will continue.**

Power loss Recovery

The printer can automatically recover a failed print during a power loss under the following conditions:

1. **You must be printing from the SD card**
2. **You must leave the front 10cm of the bed clear.**
   If you don't the hotend will crash into the print when the printer homes.

Note: **It is best to have just a few files on the SD and use a Class 10 SD card.**
If you have lots of files or folders on the SD card, the power loss recovery may fail to save the last state when the power goes out. If this happens the job will not auto recover.

You do not need to add and special commands to the print job, Power loss recovery is enabled by default.

How does power loss recovery work?

1) You are printing form SD card and the power goes out!
2) The printer saves the current printer state into a file called PLR in the root folder of the SD card.
3) When the power comes back, you will hear an SOS beep to tell you the printer is in recovery mode.
4) From the TFT panel, browse the SD card and select the file that was printing, then click 'resume'.
5) The printer will home X and Y then will home Z so it should be clear of the print.
6) The bed and hotend will be reheated.
7) The head will then raise to a height 5mm above the point where the print failed.
8) The head will lower onto the print and it will resume where it left off.
9) The PLR file will be automatically deleted.

If the power fails again, the print will be recovered from the new failed point.

If you wish to cancel recovery mode do any one of the following:
- Resume the print then stop it
- Print a different file
- Delete the PLR file from the SD card
- Send M1000 from the terminal

Live Baby Stepping during print

If you need to nudge the Z axis during a print you can now do this from the Advanced levelling menu.

Baby stepping works like this:

1) During a print Select Tools>More>Level>Advanced Setting
2) **Make sure ALL is selected**, the display will show 0.00
3) Each press of the +/- button will move the head by 0.05 mm
4) If you return to the menu during the print, that value will show how far the Z axis has been nudged.

At the end of the print the nudge value will be cleared. This is a temporary value that is not saved.

WARNING: This feature does not obey end stops so you can push the head into the bed with repeated movements, so use with caution!

Where can I get the release?

Thank you for reading this far all the good stuff is here!
You can download the full release pack here:

https://www.n-wells.co.uk/3DPrinting/Chiron/Marlin-2.0.7.2-for-the-Chiron.zip

I have built the firmware with some common options, so choose the build you want and get flashing!

All of the firmware here is for a stock build machine with stock A4988 stepper drivers, so you can use any of these files on a stock Chiron.

The differences between these builds, is how you see files from the SD card.

Chiron-2.0.7.2-Folder-View.hex
- Stock Build Machine with folder style SD browsing.

Chiron-2.0.7.2-Flat-File-View.hex
- Stock Build Machine with standard file list style SD browsing.

Chiron-2.0.7.2-Anycubic-Startup.hex
- Stock Build Machine with folder style SD browsing.
- Anycubic start-up tune.

Source Code

All the custom code for the Chiron support has been added to the main Marlin distribution on GitHub. Support for the new panel is included in the current bugfix branch of Marlin.

The latest Marlin source code is here: [Marlin (github.com)](#)

The Stock v1.3.0 Anycubic Firmware source is here: [ANYCUBIC-3D (github.com)](#)

The source code in the Zipfile for this release is a customised version of the official Marlin 2.0.7.2 release. It includes some bug fixes and my code to support the new TFT panel.

This release is NOT THE SAME AS THE CURRENT Marlin 2.0.7.2 release on GitHub.

To build your own version of this firmware:

Download the source code you wish to use.

Start with the Example configuration files for the Chiron and make the changes you need.

Edit `configuration.h` and `configuration_adv.h` in the `Marlin\` folder to include your own setting changes.

Build away!

The compiled firmware.hex file will be located in the folder Marlin\.pio\build\mega2560 \

What settings can I change?

There are many things you can change in Marlin firmware. Learn what the settings are before you start making random changes! Some settings you can change by sending G-Code commands to the printer, these are known as 'soft changes'.
Other changes require you to edit firmware configuration files then rebuild and flash the printer. These are called 'hard changes'.

I will go through the useful ones here, for the rest you need to read up on Marlin.

Soft Changes (Changes without building firmware)

You can change some machine settings using Gcode commands to the printer.
 A full description for Gcode can be found here Gcode | Marlin Firmware (marlinfw.org)

Top Tip: **To view current settings send M503 to the printer.**
To restore default settings, send M502 to the printer.

This table shows the default firmware values that will be restored by an EEPROM reset (M502).

		Anycubic 1.3.0 Value	**2.0.7.2 Value**
Step Calibration	M92	M92 X80 Y100 Z400 E405	M92 X80 Y100 Z400 E405
Max Acceleration	M201	M201 X350 Y350 Z50 E30000	M201 X3000 Y3000 Z100 E10000
Max Feed Rate	M203	M203 X100 Y100 Z20 E80	M203 X200 Y200 Z25 E25
Default Acceleration	M204	M204 P350 R3000 T350	M203 P1500 R3000 T1500
Advanced Jerk Settings Junction Deviation	M205	M205 X4 Y4 Z0.4 E20 Junction dev not supported	Classic Jerk is disabled M205 B20000.00 S0.00 T0.00 J0.03
Filament Sensor	M412	M412 S1	M412 S1 D25
Power-loss Recovery	M413	Use 'G5' command in print	M413 S1
Filament Load/Unload	M603	Not Supported	M603 L500.00 U560.00
Linear Advance	M900	Not Supported	M900 K0.05

Remember! **You must send M500 to save your changes to EEPROM.**
If you don't save your changes, they will be lost when you reboot the printer.

Hard Changes (Changes in the source code)

There are many things in the Marlin source code that can be enabled and tweaked, far too many to cover here.
Have a read of the main Marlin documentation Configuring Marlin | Marlin Firmware (marlinfw.org)

These changes are relevant to the Chiron.

Changing Stepper Driver type

Make changes to 'Configuration.h'
This example shows the standard configuration for the A4988 drivers.
To use Trinamic 2209 drivers, replace A4988 with TMC2209_STANDALONE
on lines 745,746,747 & 750
Line 753 is for the extruder.

```
Marlin  >  C Configuration.h  >  ...
728       * Stepper Drivers
729       *
730       * These settings allow Marlin to tune stepper driver timing and enable advanced options for
731       * stepper drivers that support them. You may also override timing options in Configuration_adv.h.
732       *
733       * A4988 is assumed for unspecified drivers.
734       *
735       * Use TMC2208/TMC2208_STANDALONE for TMC2225 drivers and TMC2209/TMC2209_STANDALONE for TMC2226 drivers.
736       *
737       * Options: A4988, A5984, DRV8825, LV8729, L6470, L6474, POWERSTEP01,
738       *          TB6560, TB6600, TMC2100,
739       *          TMC2130, TMC2130_STANDALONE, TMC2160, TMC2160_STANDALONE,
740       *          TMC2208, TMC2208_STANDALONE, TMC2209, TMC2209_STANDALONE,
741       *          TMC26X,  TMC26X_STANDALONE,  TMC2660, TMC2660_STANDALONE,
742       *          TMC5130, TMC5130_STANDALONE, TMC5160, TMC5160_STANDALONE
743       * :['A4988', 'A5984', 'DRV8825', 'LV8729', 'L6470', 'L6474', 'POWERSTEP01', 'TB6560', 'TB6600', 'TMC2100',
744       */
745      #define X_DRIVER_TYPE  A4988
746      #define Y_DRIVER_TYPE  A4988
747      #define Z_DRIVER_TYPE  A4988
748    ∨ //#define X2_DRIVER_TYPE A4988
749      //#define Y2_DRIVER_TYPE A4988
750      #define Z2_DRIVER_TYPE A4988
751    ∨ //#define Z3_DRIVER_TYPE A4988
752      //#define Z4_DRIVER_TYPE A4988
753      #define E0_DRIVER_TYPE A4988
754    ∨ //#define E1_DRIVER_TYPE A4988
755      //#define E2_DRIVER_TYPE A4988
756      //#define E3_DRIVER_TYPE A4988
757      //#define E4_DRIVER_TYPE A4988
758      //#define E5_DRIVER_TYPE A4988
759      //#define E6_DRIVER_TYPE A4988
760      //#define E7_DRIVER_TYPE A4988
761
```

Changing Stepper Driver direction

Make changes to 'Configuration.h'
Change true to false or false to true to change the stepper direction.

```
Marlin  >  C Configuration.h  >  ...
1196      // @section machine
1197
1198      // Invert the stepper direction. Change (or reverse the motor connector) if an axis goes the wrong way.
1199      #define INVERT_X_DIR true
1200      #define INVERT_Y_DIR true
1201      #define INVERT_Z_DIR true
1202
1203      // @section extruder
1204
1205      // For direct drive extruder v9 set to true, for geared extruder set to false.
1206      #define INVERT_E0_DIR false
1207      #define INVERT_E1_DIR false
1208      #define INVERT_E2_DIR false
1209      #define INVERT_E3_DIR false
1210      #define INVERT_E4_DIR false
1211      #define INVERT_E5_DIR false
1212      #define INVERT_E6_DIR false
1213      #define INVERT_E7_DIR false
1214
```

TFT Panel Changes

Make changes to 'Configuration_adv.h'

If the panel detection fails, you can force the panel type by removing the //
at the start of line 1615 for the standard panel
or line 1616 for the new panel

To enable the Anycubic start-up song remove the // from line 1620

To use a flat list of files instead of a folder view for the SD card,
Add // to the start of line 1626

```
Marlin > C Configuration_adv.h > ...
1607
1608    //
1609    // Additional options for AnyCubic Chiron TFT displays
1610    //
1611    #if ENABLED(ANYCUBIC_LCD_CHIRON)
1612      // Panel type detection
1613      // The default behavior is to auto detect the panel type
1614      // If necessary you can override this by uncommenting one of the lines below.
1615      //#define CHIRON_TFT_STANDARD
1616      //#define CHIRON_TFT_NEW
1617
1618      // Powerup tune
1619      // Enable the standard Anycubic powerup tune instead of the short one
1620      //#define AC_DEFAULT_STARTUP_TUNE
1621
1622      // SD Card file display
1623      // The default behaviour is a flat list of all GCODE files on the card, including subfolders.
1624      // If you are not using folder view, it is advisable to enable SDCARD_SORT_ALPHA with folders shown after files.
1625      // Enable folder view to display and navigate folders
1626      #define AC_SD_FOLDER_VIEW
1627
1628      // NOTE: If used with the new panel, each line will have '.gcode' added at the end of each line.
1629      //       This is the only way to get the panel to show the folder navigation lines.
1630
1631    #endif // ANYCUBIC_CHIRON
1632
```

Thank you to:

Shane Dekart, Robert Filippo, Bob Modrow, Quentin McCain, Luca Marongiu, Marcin Nowak, JF Kerekes, for risking your machines testing the beta version of the firmware to help get it all working as it should!

A special thank you to Jay Sommerville and many others on the Chiron FB Group, for their efforts to provide me with a new TFT panel!

I would have been totally stumped without it!