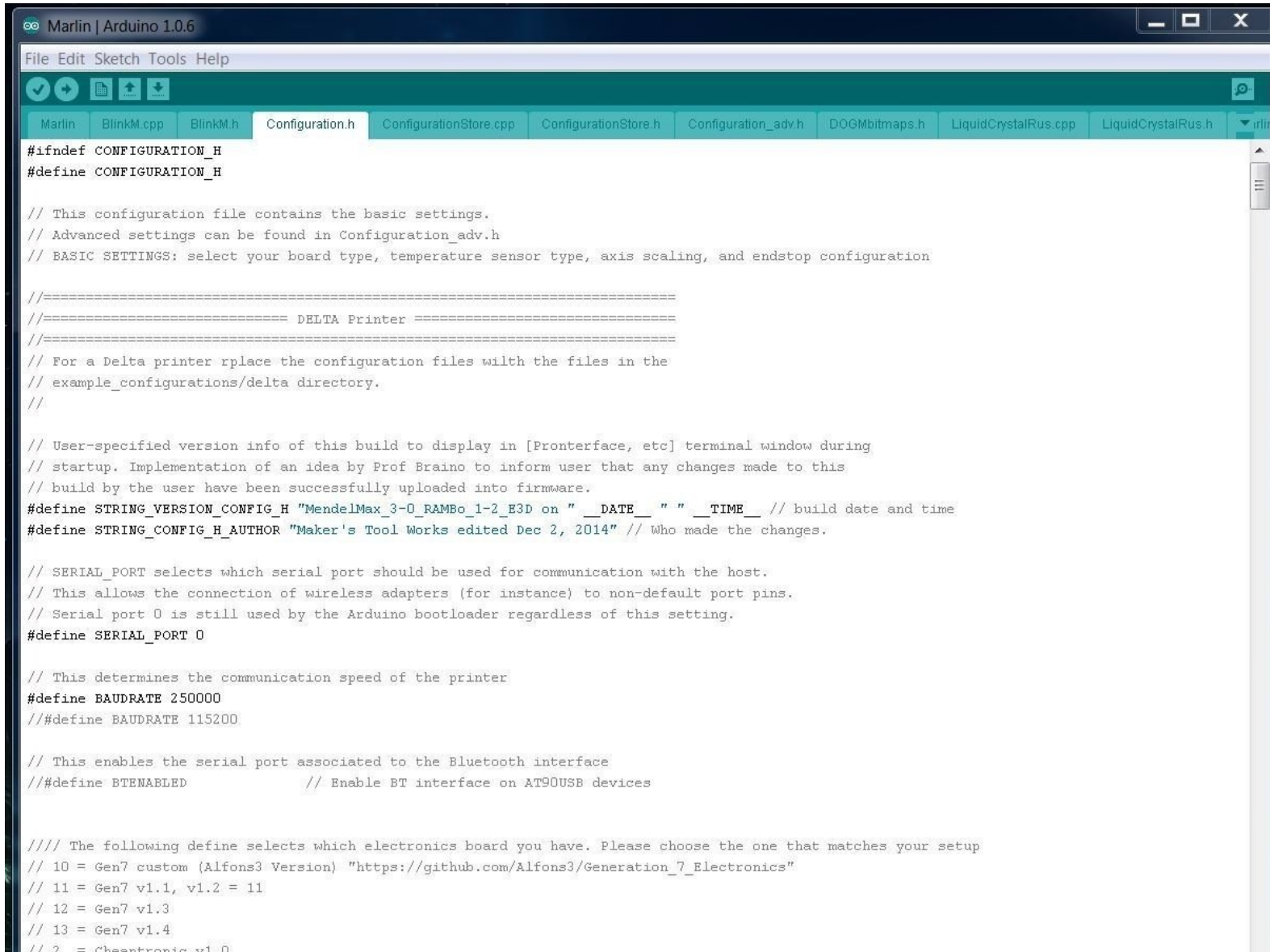


Maker's Tool Works

Firmware Programming

Tutorial on how to flash the firmware on the Rambo controller used in the MTW 3D printers.

Written By: Micro

A screenshot of the Marlin | Arduino 1.0.6 IDE. The window title is "Marlin | Arduino 1.0.6". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". The toolbar shows icons for opening, saving, and running. The file explorer at the top lists several files: "Marlin", "BlinkM.cpp", "BlinkM.h", "Configuration.h" (selected), "ConfigurationStore.cpp", "ConfigurationStore.h", "Configuration_adv.h", "DOGMbitmaps.h", "LiquidCrystalRus.cpp", and "LiquidCrystalRus.h". The main editor displays the contents of "Configuration.h". The code is a C++ header file for Marlin firmware configuration. It starts with conditional compilation directives for "CONFIGURATION_H". It contains comments explaining the file's purpose and where to find advanced settings. It defines "DELTA Printer" and provides instructions for Delta printers. It includes version and author information. It defines "SERIAL_PORT" and "BAUDRATE". It also defines "BTENABLED" for Bluetooth. Finally, it lists several board options for the Gen7 printer, with comments indicating the correct one to choose.

```
#ifndef CONFIGURATION_H
#define CONFIGURATION_H

// This configuration file contains the basic settings.
// Advanced settings can be found in Configuration_adv.h
// BASIC SETTINGS: select your board type, temperature sensor type, axis scaling, and endstop configuration

//=====
//===== DELTA Printer =====
//=====
// For a Delta printer rplace the configuration files wltch the files in the
// example_configurations/delta directory.
//

// User-specified version info of this build to display in [Pronterface, etc] terminal window during
// startup. Implementation of an idea by Prof Braino to inform user that any changes made to this
// build by the user have been successfully uploaded into firmware.
#define STRING_VERSION_CONFIG_H "MendelMax_3-0_RAMBo_1-2_E3D on " __DATE__ " " __TIME__ // build date and time
#define STRING_CONFIG_H_AUTHOR "Maker's Tool Works edited Dec 2, 2014" // Who made the changes.

// SERIAL_PORT selects which serial port should be used for communication with the host.
// This allows the connection of wireless adapters (for instance) to non-default port pins.
// Serial port 0 is still used by the Arduino bootloader regardless of this setting.
#define SERIAL_PORT 0

// This determines the communication speed of the printer
#define BAUDRATE 250000
// #define BAUDRATE 115200

// This enables the serial port associated to the Bluetooth interface
// #define BTENABLED // Enable BT interface on AT90USB devices

//// The following define selects which electronics board you have. Please choose the one that matches your setup
// 10 = Gen7 custom (Alfons3 Version) "https://github.com/Alfons3/Generation_7_Electronics"
// 11 = Gen7 v1.1, v1.2 = 11
// 12 = Gen7 v1.3
// 13 = Gen7 v1.4
// 2 = Cheantronic v1.0
```


Step 1 — Downloading the necessary files



- Access the [Download Center](#) and download the platform-appropriate versions of the following files:
Software: Arduino, Printron, and Slic3r.
- Drivers (only necessary for Windows at this time) for RAMBo. Firmware, be sure to select the appropriate version for your setup.
- Unzip all files.
- Get acclimated with all of the different software:
 - Arduino is an application used to modify, compile, and upload the firmware to the print RAMBo controller.
 - Marlin is the firmware that is running on the print controller, interpreting instructions and powering the different parts of the printer
 - Repetier is the application for communicating with the printer. It takes manual instructions as well as the toolpaths that Slic3r makes and sends them to the printer
 - Slic3r is for taking 3D models and converting them into toolpaths (gcode) for the printer to understand. Included in Repetier.

Step 2 — Configuring the Arduino IDE to upload firmware



- If using Windows, it is necessary to first install the RAMBo driver that you previously downloaded.
 - Installing RAMBo Driver
 - Attach the RAMBo to the computer via the USB cable.
 - Under Other devices, look for the RAMBo.
 - Right click RAMBo and select Update Driver Software Click Browse my computer for driver software, then click the Browse... button and browse to the directory where you have unzipped the RAMBo drivers, click OK, and then click Next
-  Taking note of the COM port listed in the title of the window, click close

Step 3 — Uploading the Marlin firmware using Arduino to your print controller



- Provide 24v to all the inputs on the Print controller.
- Open the previously-downloaded Arduino application by clicking on arduino.exe.
- Select File: Open. Browse into the directory that you extracted your firmware, then in the Marlin folder, then selecting the file Marlin.pde or Marlin.ino (whichever exists)
- Select Tools: Board: Arduino Mega 2560 Select Tools: Serial Port: and then the port that you noted earlier when installing drivers
- Select the Configuration.h tab, here you can review and change all the common settings and tuning changes.
- ⓘ Note: If you are running linux you need to change the firmware setting in Configuration.h to `#define BAUDRATE 115200`
- Press the Upload button. Check to see if the upload succeeded; if not, troubleshoot the board type, com port, and firmware