

# Monotronics

From RepRapWiki

*Schematic is missing*

## General

The RepRap Monolithic Electronics set (or Monotronics) is based upon the Generation 3 electronics. It uses the same underlying Sanguino (ATmega644P) processor and the same stepper controller circuit, except everything is on a single board.

### Contents

- 1 General
- 2 Files
  - 2.1 Design Files
  - 2.2 Firmware
- 3 Installation
  - 3.1 Host Software Recommendations
  - 3.2 Adjust the Motors
  - 3.3 Further Bench Test(s)
  - 3.4 Connecting the Motor(s)
  - 3.5 Install the Mounting Board
  - 3.6 Mounting The Electronics
  - 3.7 Connect The Endstop(s)
- 4 Uploading the Firmware
- 5 Pinout Maps
  - 5.1 Mainboard
    - 5.1.1 Breakout Pins
    - 5.1.2 RJ45 Connectors for TipManager and BedManager
  - 5.2 Tip Manager and Bed Manager
- 6 Design considerations and thoughts
- 7 Troubleshooting
- 8 Alternatives



### Example Development

Release status: unknown



<b>Description</b>	A single-board electronics config designed by TechZone
<b>License</b>	GPL
<b>Author</b>	Kymberlyaandrus
<b>Contributors</b>	
<b>Based-on</b>	Generation 3 Electronics
<b>Categories</b>	Examples, Has Files, Files Missing, Needs Render, TechZone electronics, electronics, surface-mount electronics, TechZone Mondo
<b>CAD Models</b>	none
<b>External Link</b>	none

You can access this page and its peers (for the other parts of the Huxley assembly from the Main Huxley page or from the TechZone Huxley Page These electronics can also be used on a Mendel, a LaserCutMendel, the new Mondo or even on a Prusa

## Files

### Design Files

In addition to these source files, you may find the pinout maps (see section below) to be helpful.

#### Mainboard

FILE ID#	TYPE	DESCRIPTION	AVAILABLE FORMATS	CREATED/RESERVED BY
PDF files for PCB layout	MONOTRONICS BOARDS	This file contains a separate Pdf for each layer of the PCB file	.zip	--Tech Zone Communications Feb 18, 2011
Original PCB Drawing	MONOTRONICS BOARDS	This file contains the drawing as it was created in PCB Artist (A free, but proprietary PCB drawing tool)	.zip	--Tech Zone Communications Feb 18, 2011
Original PCB Drawing	MONOTRONICS BOARDS	This file contains the schematic as an svg file. We started with the PCB file, not with the schematic (Yes, I know that is backwards)	.zip	--Tech Zone Communications Feb 18, 2011

#### Tip Manager and Bed Manager

FILE ID#	TYPE	DESCRIPTION	AVAILABLE FORMATS	CREATED/RESERVED BY
PDF files for PCB layout	MONOTRONICS AUXILIARY BOARDS	This file contains a separate Pdf for each layer of the PCB file	.zip	--Tech Zone Communications Feb 18, 2011
Original PCB Drawing	MONOTRONICS AUXILIARY BOARDS	This file contains the drawing as it was created in PCB Artist (A free, but proprietary PCB drawing tool)	.zip	--Tech Zone Communications Feb 18, 2011
Original PCB Drawing	MONOTRONICS AUXILIARY BOARDS	This file contains the schematic as an svg file. We started with the PCB file, not with the schematic (Yes, I know that is backwards)	.zip	--Tech Zone Communications Feb 18, 2011

## Firmware

We now have a new firmware, we have added some backwards compatibility to it so that the windows version of RepSnapper would show the temperature correctly. Update Oct. 2012: see also the Alternatives section for other options.

FILE ID#	TYPE	DESCRIPTION	AVAILABLE FORMATS	CREATED/RESERVED BY
TechZone Monotonics Firmware	MONOTRONICS Firmware	This file is the firmware for the Monotonics	.zip for use with arduino	--Tech Zone Communications Jan 16, 2011
TechZone Monotonics Firmware Feb 18, 2011	MONOTRONICS Firmware	This file is the firmware for the Monotonics released Feb 18, 2011	.zip for use with arduino	--Tech Zone Communications Feb 18, 2011

**NOTE:** This firmware is developed from the RepRap firmware. We added some code for communications with the tip manager and re-published it. We use this same firmware as a basis for our Generation 3 electronics firmware. All the ancillary code that is not specifically used is left in place so that it can be compared to the original RepRap firmware (besides the fact that I am too busy to figure out all the pieces which are no longer used and remove them).

When uploading firmware to the Monotonics, use an older version of the Arduino IDE. Version 0021, 0019 and 0018 seem to work really well. Newer versions may be able to communicate to the main board, but not the secondary board for the hot end.

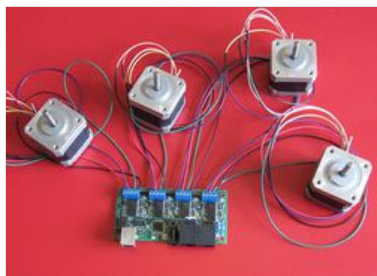
## Installation

### Host Software Recommendations

If you are using Windows, RepSnapper seems to work very well right out of the box. You can find RepSnapper here (<https://github.com/timschmidt/repknapper/downloads>) . An alternative is to use Printron. This is especially nice on OS X and Linux when combined with Slic3r.

RepSnapper includes a slicer, but if you wind up using something else, make sure you take a look at CAM ToolChains ([http://reprap.org/wiki/CAM\\_Toolchains](http://reprap.org/wiki/CAM_Toolchains)) .

### Adjust the Motors

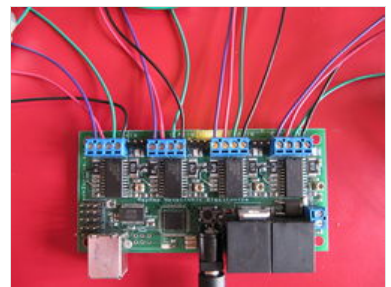


I start by connecting the motors to the mainboard (see below for wire order), on a workbench. The purpose of this is to adjust the trimpots so the stepper motor driver provides the correct current to the stepper motor. Too little current and your stepper motor will skip steps, too much and it will heat up and warp the plastic motor mounts.

I go ahead and hook up all four motors, but you could use one motor (especially if all the others are already attached to your machine) and move it from one motor controller to the next, adjusting one of them at a time. With all the motors hooked up, you can connect the 12v power (? amps). The X axis, the Y axis and the Extruder are the only motors which are always activated, and so they are the only ones you

can adjust by temperature. The Z axis only activates when it is being used, so after the other three are adjusted, I would adjust the Z to about the same position on the trimpot.

You don't need to connect the USB to your computer to get this first adjustment. After letting the electronics and motors sit for about 20 minutes, I check the temperature of the motors. I use an IR thermometer, but you can do it with your sense of touch (not as precise, but it should work fine). I want to get my motors to where they are running at 50 degrees celcius or less, most of mine are adjusted to be closer to 40 degrees. I have found that if they are hotter than that, then I am loosing torque due to the increased resistance of the hot windings... which creates a weaker magnetic feild. This is an example of a case where more is not better. If you are using your hand, 50 degrees feels uncomfortably hot still, but not unbearably hot, 40 degrees feels like a warmish/hot shower.



To turn it down, adjust the trimpot counter clockwise. To turn it up, adjust it clockwise. The trimpot is extremely sensitive, so very small adjustments may make a big difference. I would adjust about 1/8th of a turn at a time to start with, then fine tune it with even smaller adjustments. Below are images of each of the motors being adjusted. Click on the image to see it larger (true of most images here in the wiki)

With Monotonics 2.X Layout you have to turn the trimpot clockwiese to decrease the current. In my setup it starts with about 0,2 A.



IF you have the USB connected to your computer and host software connected to it, you should be able to tell the machine to turn your motors at this time and they should obey.

(--BenM 13:52, 17 February 2011 (UTC)): It is possible to reverse the direction of the motors by changing the sequence of the wiring. By replacing the Blue-Red-Green-Black order with Green-Black-Blue-Red the motor will turn clockwise when they previously did counter clockwise and vice versa. This could possibly be set in software too, but I have yet to discover that option.

(--BenM 13:52, 17 February 2011 (UTC)): A different way to adjust the motor current is to hook up all but one of the wires and then run a piece of scrap wire from the free terminal block to an Ammeter (set to A not mA) and then back to the unattached cable (using crocodile clamps). You can then monitor the approximate current to the motor as you adjust the pots. Note that it depends on whether a single coil or two coils are on so you will want to set it to a value and then step the motor in software a small distance to see if the current increases or decreases. The value for the STP-42D241 should be 0.4A/Phase for good operation.

#### NOTE About endstops and motors turning both directions

(--AlexRa 00:01, 27 January 2011 (UTC)): When the optostops are NOT connected, the board reads them as being in the home position. As the result, if you are trying to test the motors from the PC (using direct control commands in RepSnapper or other host program), the motors would go forward, but not backward (might just jerk a little, especially the Z motor). Do connect the optos and the motors will move both ways!



Adjusting the Extruder Motor



Adjusting the X Axis Motor



Adjusting the Y Axis Motor



Adjusting the Z Axis Motor

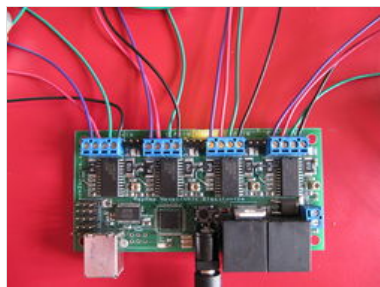
## Further Bench Test(s)

While we have the system out on the bench, I like to check and make sure that the mainboard and the tip manager are talking as they should. This is very simple to do. I unhook the 12v from the system, since it is not needed for this test. The first thing to look at is the temperature your host software is reporting to you. With the correct firmware (from Files above) installed and your host software communicating with the board, it should report a temperature of 2511 for both the tip manager and the Heated Bed Manager. If you are not seeing 2511 then one of two things is wrong (OK, maybe three things). The software is not connecting to the mainboard, or the mainboard has the wrong firmware; the third possibility is that the board is bad...

I would troubleshoot the above situation by re-flashing the mainboard. If it flashes correctly then it is 99% likely that it will report 2511 when your host software connects to it. I would then start checking the software to make sure it is working as it should... check the com port(s), check the computers OS, etc. I have seen a very high level of success running RepSnapper on 32 bit operating systems, the jury is still out on 64 bit operating systems - although it should work.

Next, take a tip manager and connect it to the mainboard with the RJ45 cable (a standard ethernet cable will work if you need a replacement). When you plug it in, the tip manager should immediately start reporting the room temperature to you. If you connect the same tip manager into the heated bed manager location, you should get room temperature for the heated bed temperature. Yes, it reports a temperature even without the thermocouple connected.

## Connecting the Motor(s)



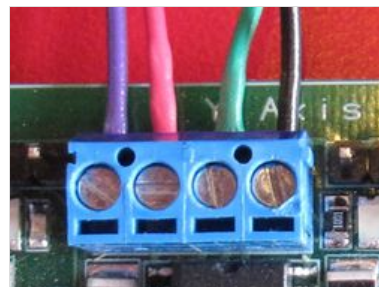
If you have read very much information about RepRap and the various electronics and motors... then you know that there are lots of motors out there that work... and many that don't. The first thing you should know is that your motor needs to be a bidirectional stepper motor, which has 6 ohms of resistance per phase or more. Although many people run motors with less resistance, it is not a good idea, and it has the possibility of drawing more current than the controller chip can handle, thereby frying the chip.

**A TECHNICAL NOTE ABOUT UNIDIRECTIONAL STEPPER MOTORS** - If you have a unidirectional motor, you can wire it up in bidirectional mode (most of the time). I am not going to go into

details here about how to do that, but the information is available on the web.. and someday I may create a wiki entry about doing this. I mention it because it may make finding a motor easier.. also because when you connect a unidirectional motor up in bidirectional mode, it combines two of the phases, so each separate phase from a unidirectional motor can be as low as 3 ohms, which will give you 6 ohms when you hook it up in bidirectional mode. See stepper wiring for details.

There are lots of different wire colors for stepper motors, but the most common are blue, red, green, and black wires. These motors are connected with the wire sequence of Blue-Red-Green-Black. If when you connect your motor, it causes the axis or extruder to move backwards from what you tell it, the simply mirror your wire connection (ie Black-Green-Red-Blue). On the right is a picture of this most common sequence. The pictures at the bottom of this gallery show some other wire colors as they should be connected to the stepper controller.

Most motors come with more wire than is needed, I recommend cutting the excess off, after the motor is mounted. Leave enough slack for the X axis motor to move up and down the Z axis and for the extruder which is not attached to the frame.

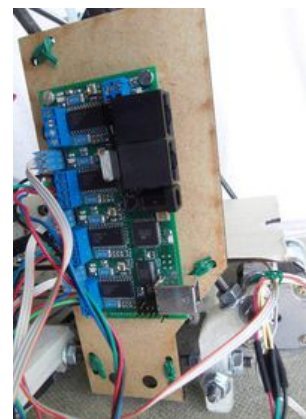


## Install the Mounting Board

The mounting board is not really part of the electronics set. It is, however required in some form. The mounting board we ship with the "TechZone Huxley Kit", or with the "LaserCut Mendel with Monotronics Kit" has a peculiar shape and is designed to be mounted to the side of the machine. I mount it to the side which is closest to the motors, to reduce the wire length needed.

It simply fastens to the frame using some of the small zip ties that came in your kit.

NOTE: Pictures of various Mendels often show the thick sheet with the electronics mounted in front of the frame between the two threaded bar. **UNLIKE** that, in TechZone's design the board **MUST** be placed on the side of the machine, otherwise just too many things are getting in the way. Specifically, in this position the "network" cable between the main board and the tip control sub-board (fixed on the X caret) can move freely with the caret.



## Mounting The Electronics

If you have been following the Huxley build instructions (or most other RepRap instructions) you will have already mounted the opto-endstops.

I like to mount the Monotronics board to the Mounting board with 3mm bolts and screws. You can use plastic stand-off feet, or wire ties, or even hot glue if you prefer. If using the 3mm bolts, be careful to make sure that the bolts cannot short out any of the components on the board.

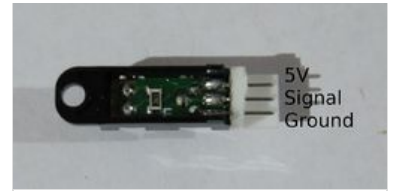
## Connect The Endstop(s)

One of the most common mistakes, is to hook all the endstops up at once, then power the electronics up and see what happens. IF you have them hooked up wrong, you just fried all three. I suggest that you hook up one and test it, then if that goes well, hook up the next one and test it... etc.. then they are all done and working. You will probably want to click on the images, to be able to see the details or read the text on them.

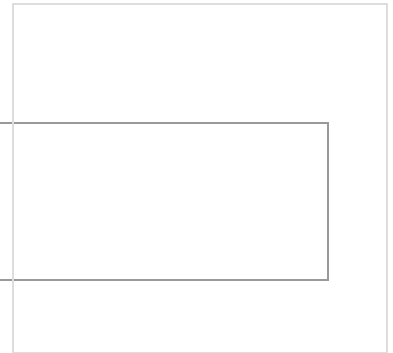
To the left is a photo of the Monotronics (click to enlarge), with the pin outs of the opto-endstops labeled. On the Right is the endstop itself, with the pin outs labeled.



If you want to be cautious and make sure you don't mix up the pin connections of your endstops, you can take a permanent marker and mark the 5v side of the endstop. This is where I will put the colored edge of the wire. (see photo below)



A three pin wire is used to connect the two together, as shown on the right, some close up details are shown below. For clarity, this wire connection is shown NOT installed in the machine, you will want to connect yours in the machine. The connector fits quite nicely on the main board, it only sort of fits on the endstop. I like to use a dab of hot glue on the endstop end so that the connector doesn't vibrate off while printing.



(--AlexRa): I found it more convenient to remove the white plastic tab, de-solder the pins from the tiny PCB on the opto, then just solder the cable directly to it. The benefits are: (a) avoid problems with the badly fitting connector; (b) 3 less connectors to crimp manually; (c) there is not much room on the Huxley to fix the optos, removing the tab + connector makes fitting them easier.

**NOTE** TechZone is looking for some pre-assembled cables with the correct ends on them, if you know of a good source, please let us know.

All three endstops are handled the same, the connector furthest to the left is the Z Axis, the one in the middle is the Y axis and the one on the right is the X axis.



## Uploading the Firmware

TechZone ships the electronics with the newest firmware which they have at the time of shipping. Your electronics should work when you get them, but to get added features you may want or need to upload a newer firmware. Follow these instructions and you will find that you have a good success rate. You may need an older version of the Arduino environment, as noted in the firmware file section of this document.

This quick tutorial assumes that you already have the Arduino/Sanguino software properly installed. If you do not yet have the software installed and configured you can find instructions here <http://arduino.cc/en/Guide/HomePage> in the wiki. The details for installing Sanguino support are on the Sanguino web page (<http://sanguino.cc/softwareforlinux>). The process below will work with other RepRap boards as well - just use the appropriate file for them. There are multiple ways to do this (variations of details), we have found that the following procedure works quite well. You may need to click on the images, to see where the mouse is on the screenshots.

### Windows 7 Users, this may help

(--AlexRa): when connecting the Monotonics to Windows 7 (I use 64-bit Windows, presumably the 32-bit version would behave the same), the USB driver as well as the serial-on-USB get installed automatically the first time you plug in the board. **Remember the serial port number** Windows reports during the installation! On my machine, it happens to be "COM3:", but that could vary. What you do have to check, is that both the **serial port** and the **connection speed** are correctly set in Arduino. If you are not very familiar with the environment (I'm not), here is how to find the speed setting: open the "Serial Monitor" window (using either a toolbar button or Ctrl-Shift-M) and look there. Set the speed to 19200.

### Dealing with Firmware Compilation Errors

(--KD0SKH): In the code for FiveD\_GCode\_Interpreter.pde, if you find that attempts to compile the code gives dozens of errors along the lines of: error: 'Serial' was not declared in this scope You need to place `#include "WProgram.h"` **above** `#include <HardwareSerial.h>` for anything to work. I was using the 32-bit Arduino IDE version 0019 when setting up my printer on a machine



running Windows 7 64-bit.

## Linux Users, this may help

(--KD0SKH): If you receive the error `HardwareSerial.cpp:132:15: error: attempt to use poisoned "SIG_USART1_RECV" when attempting to compile`, edit the sanguino's `HardwareSerial.cpp` file with the path listed in the error code. Comment out the following expressions `#elif defined(SIG_USART1_RECV)` and `#error SIG_USART1_RECV`. Afterwards you should be able to compile and upload your FiveD firmware! Works with Arduino IDE 0018 and 0022 on Arch Linux with kernel version 3.12.5. Other potentially relevant information about my environment: `avr-gcc 4.8.2`, `avrdude 6.0.1`, `libusb-compat 0.1.5`, and `java-rxtx 2.2pre2` were present on my system.

## Regarding the Reset button

(--AlexRa): at least in my setup (and I don't see how it can be any different on other computers connecting to the same board) I do NOT NEED to touch or hold Reset to upload the firmware from Arduino. I think much of the "things just don't work unless you press all the buttons you can think of" confusion may be caused by forgetting to set the connection speed (the default seems to be 9600 in Arduino and 19200 in the serial-to-USB driver).

(--AlexRa)Correction: the Arduino bootloader is supposed to wait for an upload "a few seconds" after a reset, then pass control to the currently loaded firmware. According to that, you have to click Reset (but not hold it), then immediately start the upload.

(--TechZone)The reason that we hold the reset in, is that it can take a while to build the firmware before it begins to transfer the file (depending upon the computer of course), if you hold it in until the right moment, then letting go will make sure that the board is in a ready state to receive the data from the arduino environment. The Idea of holding in the reset is simply a universal way of explaining to a wide variety of users how to get the timing of the reset to work well. Feel free to try it both ways.

(--Alromh87):Just take in mind that latest Sanguino expect a bootloader baud-rate of 57600, while some boards where shipped with 38400 baud-rate. So if your having trouble with the upload you might have your baud-rate wrong.

Download the firmware from the files section above and unzip it to a location you can find in the next few steps  
It should create several sub directories with files in them.

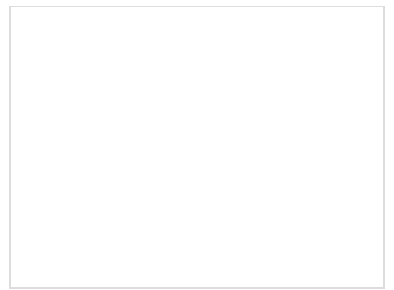
Open the Arduino environment, click on File - Open

Navigate to where you unzipped the file from above.

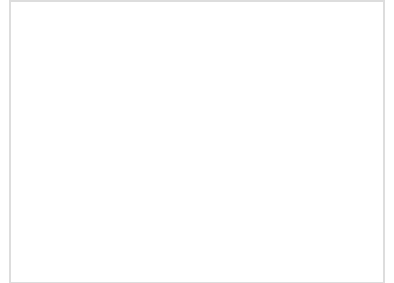
Open the FiveD\_GCode folder

To upload the firmware to the monolithic, select the `FiveD_GCodeInterpreter`, if you are updating firmware on Generation 3 electronics, you would use this selection for the mainboard and the Extruder file for the extruder (you would need a different firmware, obtained

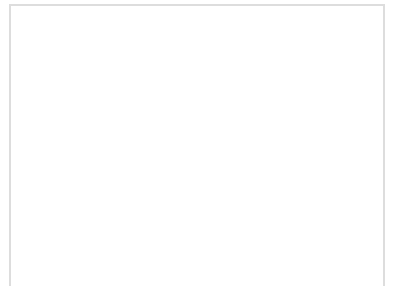
for the Generation 3 electronics).



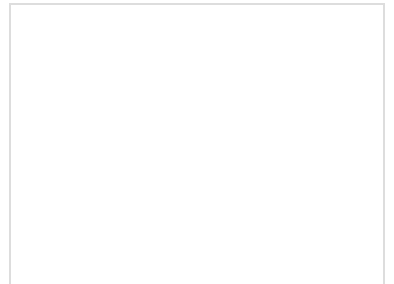
Select the file on the top right, as shown in the picture on the right



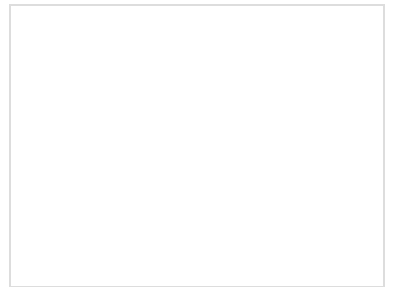
We need to make sure that the Arduino environment knows which type of board we are loading to: for the Monolithic, and the mainboard on the Gen 3 electronics select Sanguino, for the extruder controller in the Gen 3 electronics select Diecimila.



Make sure the correct serial port is selected, it should be a USB port that the machine found when you plugged the board in. You may need to unplug other USB devices for it to find the correct device

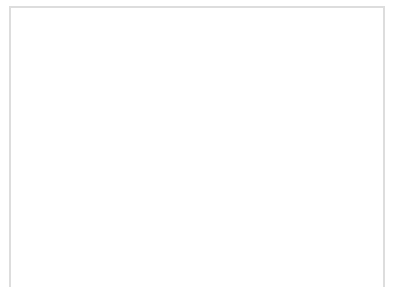


Click on the upload button in the Arduino program. As soon as you click this, hold the reset button down on the board.



As soon as the file size (circled text in the image to the right) is reported, let go of the reset button.

As it uploads, the debug LED will flash, when it is done the Arduino program will report that it is done.



## Pinout Maps

We tried to label most of the connections on the mainboard as clearly as we could. There is a block of pins on the left side of the board, near the USB connector, which are not labeled. These pins are intended for creative people to be able to add features or peripherals as they would like.

The pinout map created by TechZone has both the Arduino pin names (for programming purposes) and, where appropriate, a component or usage name for the pin. We have done this in hopes that it makes it easier to reference and modify the usage of this electronics set.

## Mainboard

### Breakout Pins

GND	+5V	D26/A5
GND	+5V	D1
GND	+5V	AREF
GND	+5V	D4
GND	D31/A0 (with pulldown and capacitor for thermistor)	

### RJ45 Connectors for TipManager and BedManager

**THIS IS NOT ETHERNET!**



Name	Function (Pin1/Pin2)
1	1_Wire_data (13/10)
2	+5V
3	+12V
4	+12V
5	GND
6	GND
7	GND
8	To_transistor for load (14/11)

## Tip Manager and Bed Manager

**NOTE** (--AlexRa): until Lambert provides a better info, here is how to connect the tip manager sub-board:

(--Alromh87: I am trying to enable tipManager on Marlin Firmware, I will be documenting pinouts and steps needed to use this electronics with Marlin, but cant figure out the use of a chip in the Daughter board (MFK01042500) does anyone knows whats this for?.

## Design considerations and thoughts

## Troubleshooting

- A motor vibrates but does not turn
  - adjust current setting resistor and/or decrease motor speed
- A motor turn forward but not backward
  - connect the optos end-stop and the motor will go both ways. When the optostops are NOT connected, the board reads them as being in the home position.

## Alternatives

You can use Sprinter firmware on your monotronics board. Using Sprinter, you can disregard the tip manager and connect a thermistor and a power transistor directly to the main board. Use this to interface other hotends to your printer, such as a J-Head. (If anyone designs a new part to hold the hot-end in place, please link to it from here.)

Retrieved from "<http://reprap.org/mediawiki/index.php?title=Monotronics&oldid=152444>"  
Categories: [Unknown status developments](#) | [Examples](#) | [Has Files](#) | [Files Missing](#) | [Needs Render](#) | [TechZone electronics](#) | [Electronics](#) | [Surface-mount electronics](#) | [TechZone Mondo](#)

- 
- This page was last modified on 4 August 2015, at 20:10.
  - Content is available under GNU Free Documentation License 1.2.